

SECURESPHERE®

Web Security

Web Application Firewall (WAF)
ThreatRadar

User Guide

Version 10.5
March 2014

www.imperva.com

Copyright Notice

© 2014 Imperva, Inc. All Rights Reserved.

Follow this link to see the SecureSphere copyright notices and certain open source license terms:

https://www.imperva.com/sign_in.asp?retURL=/articles/Reference/SecureSphere-License-and-Copyright-Information
This document is for informational purposes only. Imperva, Inc. makes no warranties, expressed or implied.

No part of this document may be used, disclosed, reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language in any form or by any means without the written permission of Imperva, Inc. To obtain this permission, write to the attention of the Imperva Legal Department at: 3400 Bridge Parkway, Suite 200, Redwood Shores, CA 94065.

Information in this document is subject to change without notice and does not represent a commitment on the part of Imperva, Inc. The software described in this document is furnished under a license agreement. The software may be used only in accordance with the terms of this agreement.

This document contains proprietary and confidential information of Imperva, Inc. This document is solely for the use of authorized Imperva customers. The information furnished in this document is believed to be accurate and reliable. However, no responsibility is assumed by Imperva, Inc. for the use of this material.

TRADEMARK ATTRIBUTIONS

Imperva and SecureSphere are trademarks of Imperva, Inc.

All other brand and product names are trademarks or registered trademarks of their respective owners.

PATENT INFORMATION

The software described by this document is covered by one or more of the following patents:

US Patent Nos. 7,752,662, 7,743,420, 7,640,235, 8,024,804, 8,051,484, 8,056,141, 8,135,498 and 8,181,246.

Imperva Inc.

3400 Bridge Parkway, Suite 200

Redwood Shores, CA 94065

United States

Tel: +1 (650) 345-9000
Fax: +1 (650) 345-9004

- **Website**: http://www.imperva.com
- **General Information**: info@imperva.com
- **Sales**: sales@imperva.com
- **Professional Services**: consulting@imperva.com

Technical Support: support@imperva.com

Imperva-SecureSphere-v10.5-Web-Security-User-Guide-v1

End User License and Services Agreement

BY CLICKING ON THE "ACCEPT" BUTTON, TAKING AN ACTION TO INDICATE ACCEPTANCE, OR USING THE PRODUCTS (AS DEFINED BELOW) END USER AGREES TO THE TERMS OF THIS END USER LICENSE AND SERVICES AGREEMENT ("AGREEMENT") WITH IMPERVA, INC. ("IMPERVA"). IN THE EVENT THE INDIVIDUAL IS ENTERING INTO THIS AGREEMENT ON BEHALF OF A CORPORATE OR OTHER PUBLIC OR PRIVATE ENTITY, END USER REFERS TO THAT ENTITY, AND SUCH INDIVIDUAL CERTIFIES THAT HE/SHE IS AN AUTHORIZED REPRESENTATIVE OF THE END USER. IF END USER DOES NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, CLICK THE "CANCEL" BUTTON, DISCONTINUE THE SET-UP AND INSTALLATION OR DISCONTINUE USE OF THE PRODUCT. IF THE TERMS OF THE AGREEMENT ARE CONSIDERED AN OFFER, ACCEPTANCE IS EXPRESSLY LIMITED TO THESE TERMS.

1. **Definitions**. The following capitalized terms shall have the meanings set forth below:
   a) "**Appliance**" means the Imperva branded computer hardware on which the Software operates.
   b) "**Delivery**" shall mean, (i) in the case of Software, when the Software is made available by Imperva for End User to electronically download; (ii) in the case of Subscription Services, when the Subscription Service has been provisioned and made available to End User to access; and (iii) in the case of an Appliance, when the Appliance has been tendered by Imperva for shipment.
   c) "**Documentation**" means Imperva's technical specifications that accompany and describe the installation, use and operation of a Product.
   d) "**End User**" means the party that has purchased the Products for its own use, either directly from Imperva or through an authorized third party.
   e) "**Licensed Volume**" means the volume or other measurement of permitted use for the Products as agreed to by Imperva.
   f) "**Open Source Software**" means third party software that Imperva distributes with the Software pursuant to a license that requires, as a condition of use, modification and/or distribution of such software, that the software or other software combined and/or distributed with it be (i) disclosed or distributed in source code form; (ii) licensed for the purpose of making derivative works; or (iii) redistributable at no charge.
   g) "**Products**" mean Appliances, Software or Subscription Services, as the case may be.
   h) "**Services**" mean Professional Services or Support, as the case may be.
   i) "**Software**" means Imperva's or its licensors' software (in object code format) or content, any updates or upgrades thereto provided to End User by Imperva and any Documentation pertaining thereto. Software may be delivered to End User on Appliances or on a standalone basis. The term "Software" does not include Open Source Software.
   j) "**Subscription Services**" mean the subscription services, including content, updates and upgrades thereto, that may be made available to End User by Imperva directly or through its partners and suppliers. Subscription Services include, without limitation, the ThreatRadar service and Imperva offered services that are "powered by Incapsula."
   k) "**Support**" means the technical support and maintenance services for the Product and periodic bug fixes and updates to the Software that Imperva may make generally available at an annual subscription cost to end users.
   l) "**Professional Services**" mean the installation, configuration, and training services that Imperva may provide to an End User.
2. **Licenses and Restrictions.**
   a) **Software**. Conditioned on the terms and conditions of this Agreement, Imperva grants End User a perpetual (subject to Section 11 or any term license restrictions), nonexclusive, nontransferable, nonsublicensable license to use the Software in accordance with its Documentation only for End User's internal business purposes on the Appliances or in the Licensed Volume licensed by End User. If End User purchases Software on a standalone basis, the license granted herein shall include the right to copy the Software up to the Licensed Volume.
   b) **Subscription Services**. Conditioned on the terms and conditions of this Agreement, Imperva grants End User a nonexclusive, nontransferable, nonsublicensable, revocable right to use and access the Subscription Services in accordance with its Documentation only for End User's internal business purposes.

c) **Restrictions**. End User may not (and may not permit any third party to): (i) modify, incorporate or use in any other works, translate, reverse engineer (except to the limited extent applicable statutory law expressly prohibits reverse engineering restrictions), decompile, disassemble, otherwise attempt to derive source code from or create derivative works based on the Products; (ii) make unauthorized copies of the Products; (iii) disclose, distribute, transfer or market the Products to third parties; (iv) remove or modify any proprietary notices, labels or marks on or in any copy of the Products; (v) distribute, sell, sublicense, rent, lease or use the Products (or any portion thereof) for time sharing, hosting, service provider or other computer services to third parties or otherwise make the functionality of the Software available to third parties; (vi) publicly disseminate performance information or analysis (including, without limitation, benchmarks and performance tests) from any source relating to the Products; (vii) access the database or any other third party product that is embedded in the Software with applications other than the Software; or (viii) use the Products other than as permitted herein.

d) **Appliance**. End User acknowledges that the Software included with the Appliance is licensed and not sold. Such Software is licensed solely in conjunction with such Appliance (and not separately or apart from such Appliance). If End User sells, leases, lends, rents, distributes or otherwise transfers any Appliance to any third party or if Imperva terminates this Agreement under Section 11.b or for a breach of Section 2.c, then End User will erase all Software from such Appliance.

e) **Open Source Software**. Open Source Software is copyrighted and licensed under the GPL/LGPL and other licenses. Copies of or references to those licenses are included with Software in the "Help" section. End User may obtain the complete corresponding Open Source Software source code for a period of three years after Imperva's last shipment of the Software, by sending a money order or check for $10 to: Legal Department – Open Source Software Request, Imperva, Inc., 3400 Bridge Parkway, Suite 200, Redwood Shores, CA 94065, United States.

f) **US Government End User**. For purposes of this Agreement, "commercial computer software" is defined at FAR 2.101. If acquired by or on behalf of a civilian agency, the U.S. Government acquires this commercial computer software and/or commercial computer software documentation and other technical data subject to the terms of the Agreement as specified in 48 C.F.R. 12.212 (Computer Software) and 12.211 (Technical Data) of the Federal Acquisition Regulation ("FAR") and its successors. If acquired by or on behalf of any agency within the Department of Defense ("DOD"), the U.S. Government acquires this commercial computer software and/or commercial computer software documentation subject to the terms of the Agreement as specified in 48 C.F.R. 227.7202-3 of the DOD FAR Supplement ("DFARS") and its successors. This U.S. Government End User Section 2(f) is in lieu of, and supersedes, any other FAR, DFARS, or other clause or provision that addresses government rights in computer software or technical data.

3. **Support and Subscription Services**. Provided End User has an active and fully paid contract for Support, Imperva will provide Support in accordance with its standard Support terms then in effect. Provided End User has an active and fully paid contract for Subscription Services, Imperva will provide such services in accordance with its standard Service Level Agreement (SLA) then in effect.

4. **Additional Terms for Subscription Services**.

a) **Accessing and Use of Subscription Services**. Except as explicitly set forth herein, End User is solely responsible for acquiring and maintaining all of the equipment, software, services and items necessary to access and make use of the Subscription Services, including without limitation paying all charges, taxes, and other costs and fees related to internet access. End User may access the Subscription Services only through the interfaces and protocols provided or authorized by Imperva and its partners, and agrees to set up, maintain and use the Subscription Services in strict compliance with Imperva's and its partners' instructions. End User is solely responsible for maintaining the confidentiality of any passwords and account information required to access Subscription Services, for all acts that occur in connection with End User's account and to immediately notify Imperva of any unauthorized use of End User's account. In the event of expiration or termination of any Subscription Services that require DNS routing, End User will be solely responsible for rerouting its DNS traffic and Imperva, its partners and suppliers shall have no liability for End User's failure to do so.

b) **Authorization**. Certain Subscription Services are offered to cache, monitor and optimize web pages and web sites. As such, End User hereby grants Imperva and its partners a nonexclusive, worldwide, fully paid-up, royalty-free license to use, transfer, display, minimize and compress the content and material on End User web sites ("**End User Content**"), in any media formats, solely as necessary for the performance of the Subscription Services. Imperva and its partners do not provide backup services for End User Content, and if End User's use of the Subscription Services terminates for any reason, Imperva and its partners may, without notice, delete or deny End User access to any of content or meta data that may remain in its/their possession or control. In addition, End User agrees that if, at Imperva's and its partners' sole determination, End User is using the Subscription Services in a manner that violates laws, rules or regulations or creates an excessive burden or potential adverse impact on Imperva's, its partners' or its suppliers' systems, business or customers, Imperva, its partners or its suppliers may suspend or terminate End User's access to the Subscription Services without notice to or liability to End User.

5. **Professional Services**. Professional Services, if any, to be provided by Imperva to an End User will be subject to a separate statement of work ("**SOW**") agreed to by Imperva and Imperva's standard Professional Services terms then in effect.

6. **Fees, Payment Terms, Shipment and Delivery**.

a) For orders accepted directly by Imperva, End User shall pay Imperva the applicable fees designated by Imperva. Overage fees may apply if End User exceeds its allowance for any Subscription Services at Imperva's then-current overage rates. Any fees payable to Imperva are non-refundable and payable in US Dollars. End User shall also pay all sales, use, value-added and other taxes, tariffs and duties of any type assessed against End User, except for taxes based on Imperva's income. Should End User be required under any law or regulation of any governmental entity or authority outside of the United States, to withhold or deduct any portion of the payments due to Imperva, then End User shall increase the sum payable to Imperva by the amount necessary to yield to Imperva an amount equal to the sum it would have received had no withholdings or deductions been made. Fees shall be invoiced as follows: (a) fees for all Subscription Services and Support shall be invoiced in advance; (b) fees for Software licenses and Appliance purchases will be invoiced upon Delivery. All payments from End User to Imperva are due net thirty (30) days after the date of invoice. If End User's account for Subscription Services or Support is thirty (30) days or more overdue, in addition to any of its other rights or remedies, Imperva reserves the right to suspend such services to End User, without liability to End User, until such amounts are paid in full. Imperva shall have the right to conduct and/or direct an independent accounting firm to conduct, during normal business hours, an audit of End User's facilities, computers and records to confirm End User's use of Products is in compliance with this Agreement. End User shall provide reasonable cooperation with any such audit.

b) Imperva will use commercially reasonable efforts to ship the Appliances and the Software license keys at the times requested in Orders (in partial or full shipments); provided, however, that Imperva shall in no event be liable for any delay in Delivery or for failure to give notice of delay. Without liability to any person and without prejudice to any other remedy, Imperva may withhold or delay shipment of any Order if End User is late in payment or is otherwise in default under this Agreement. Title to Appliances and risk of loss shall pass to End User upon Delivery, and Products shall be deemed accepted by End User upon Delivery. Appliances shall be delivered Ex Works (Incoterms 2000) Imperva's designated manufacturing facility. End User may specify shipping instructions with the Order. In the absence of specific shipping instructions from End User, Imperva will ship by the method it deems most advantageous. End User shall pay and be exclusively liable for all costs associated with shipping and delivery including without limitation, freight, shipping, customs charges and expenses, cost of special packaging or handling and insurance premiums incurred by Imperva in connection with the shipment of Appliances to End User. In its discretion, Imperva may advance shipping charges on behalf of End User on Appliances shipped to End User, and End User agrees to reimburse Imperva for any such advanced charges and expenses.

7. **Confidentiality**.

a) As used herein, "**Confidential Information**" means all confidential and proprietary information of a party ("**Disclosing Party**") disclosed to the other party ("**Receiving Party**"), whether orally or in writing, that is designated as confidential or that reasonably should be understood to be confidential given the nature of the information and the circumstances of disclosure. Confidential Information includes, without limitation, the Products, their performance (including any benchmarking information) and Imperva's pricing of the Products and Services. Confidential Information shall not include any information that: (i) is or becomes generally known to the public without breach of any obligation owed to the Disclosing Party; (ii) was known to the Receiving Party prior to its disclosure by the Disclosing Party without breach of any obligation owed to the Disclosing Party; (iii) was independently developed by the Receiving Party without breach of any obligation owed to the Disclosing Party; or (iv) is received from a third party without breach of any obligation owed to the Disclosing Party.

b) The Receiving Party agrees that it will (i) use Confidential Information for the sole purpose of exercising its rights and performing its obligations under this Agreement, (ii) divulge Confidential Information only to those of its employees, directors, independent consultants or agents who have a need to know such Confidential Information and who are bound by professional duty or in writing (in advance) to confidentiality and non-use obligations at least as protective of such information as this Agreement, and (iii) not disclose any Confidential Information to any third party. The Receiving Party shall notify and cooperate with the Disclosing Party immediately upon discovery of any unauthorized use or disclosure of Confidential Information of the Disclosing Party. The Receiving Party may disclose Confidential Information to comply with an order from a court of competent jurisdiction or with a mandatory requirement of a governing regulatory body, provided such party, to the extent permitted by law and as soon as reasonably practicable under the circumstances, informs the Disclosing Party and allows the Disclosing Party the opportunity to object to the disclosure order or to take action to preserve the confidentiality of the information. The Receiving Party shall cooperate with the Disclosing Party in such party's reasonable efforts to limit the disclosure of the Information.. End User acknowledges, understands and agrees that Imperva may, as part of its provision of the Product and/or Services to End User, collect, store and use information obtained from End User, including, but not limited to, information about End User's users and customers ("**Information**") for the purposes of the provision of the Product, Services and other services to End User and may use such information for analysis and improvement of Imperva's products and services. End User represents and warrants that it has all rights and permissions necessary to grant Imperva access to such Information as contemplated herein.

c) Upon termination of this Agreement for any or no reason, the Receiving Party shall (i) immediately cease all use of the Disclosing Party's Confidential Information, (ii) at the instruction of the Disclosing Party, either promptly destroy all Confidential Information of the Disclosing Party or return all Confidential Information of the Disclosing Party.

d) If the Receiving Party discloses or uses (or threatens to disclose or use) any Confidential Information of the Disclosing Party in breach of this Section 7, the Disclosing Party shall have the right, in addition to any other remedies available to it, to seek injunctive relief to enjoin such acts, it being specifically acknowledged by the parties that any other available remedies are inadequate.

8. **Proprietary Rights; Indemnity**.

a) All title and intellectual property rights in and to the Products are owned exclusively by Imperva and its partners and suppliers. Other than as expressly set forth in this Agreement, no license or other rights in or to the Products and intellectual property rights thereto are granted to End User, and all such licenses and rights are hereby expressly reserved. Any ideas, suggestions, modifications and the like made by End User with respect to a Product will be the property of Imperva regardless of whether Imperva chooses to exercise its rights to incorporate such ideas, suggestions or modifications into the Product.

b) Subject to the remainder of this Section 8.b, Imperva will indemnify End User from any Liability (as defined below) to a third party resulting from infringement of a U.S. patent or any copyright, or misappropriation of any third party trade secrets by the Product as delivered ("**Infringement Claim**"); provided that End User (1) promptly notifies Imperva of any and all threats, claims and proceedings of such Infringement Claim, (2) gives reasonable assistance in response to Imperva's request for assistance, and (3) grants Imperva sole control over defense and settlement thereof. For purposes of this section "Liability" means the resulting costs (including reasonable attorneys' fees) and damages awarded against End User to the third party making such Infringement Claim by a court of competent jurisdiction or agreed in settlement. The foregoing obligations do not apply with respect to Products or portions or components thereof, (i) that are modified after delivery by Imperva, (ii) combined with other products, processes or materials, where the alleged infringement relates to such combination, (iii) where End User continues allegedly infringing activity after being notified thereof or modifications that would have avoided the alleged infringement have been made available to End User, or (iv) where End User's use of such Product is not strictly in accordance with this Agreement. In the event that Products are held to or believed by Imperva to infringe, Imperva at its discretion, will have the option to (A) modify the allegedly infringing Products to be non-infringing, (B) obtain for End User a license to continue using the Products, or (C) request the return of the Product and upon receipt thereof terminate this Agreement as to the infringing Product and refund to End User the unused portion of the fees paid under this Agreement for such infringing Product, depreciated on a straight-line basis over a three (3) year period. End User will defend, indemnify and hold Imperva harmless against any claims, damages settlements and expenses (including attorneys' fees) excluded from Imperva's indemnity obligations in (i) – (iv) above. THIS SECTION SETS FORTH IMPERVA'S SOLE OBLIGATION AND END USER'S SOLE AND EXCLUSIVE REMEDY IN THE EVENT OF VIOLATION OF THIRD PARTY RIGHTS.

9. **Warranty and Disclaimer**.

a) Imperva warrants that during the sixty (60) day period commencing on the date of first Delivery, the Software and the Appliances will perform substantially in accordance with their Documentation. In the event of a breach of the foregoing warranty with respect to the Software, as End User's sole and exclusive remedy, Imperva shall, at its sole expense, use reasonable efforts to modify the Software, so that it performs substantially in accordance with its Documentation. In the event of a breach of the foregoing warranty with respect to an Appliance, as End User's sole and exclusive remedy, Imperva shall, at its sole expense and at its option, repair the Appliance or replace the Appliance with a new or reconditioned Appliance that performs materially in accordance with its Documentation. The foregoing warranty extends only to the original purchaser and will not apply to the misuse of or damage to the Products. The rights and remedies granted End User under this Section state Imperva's entire liability, and End User's exclusive remedy, with respect to any breach of the warranty set forth in this Section 9(a).

b) EXCEPT AS EXPRESSLY SET FORTH IN SECTION 9(a), THE PRODUCTS OR SERVICES ARE PROVIDED "AS-IS' AND IMPERVA MAKES NO WARRANTY OF ANY KIND, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE. IMPERVA, ITS PARTNERS AND SUPPLIERS MAKE NO WARRANTY THAT USE OF THE PRODUCTS OR SERVICES WILL BE UNINTERRUPTED, ERROR-FREE OR DEFECT-FREE, OR AVAILABLE AT ALL TIMES. IMPERVA HEREBY SPECIFICALLY DISCLAIMS, ON BEHALF OF ITSELF AND ITS PARTNERS AND SUPPLIERS, ALL IMPLIED WARRANTIES, INCLUDING ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW.

10. **Limitations of Liability**. IN NO EVENT WILL END USER'S OR IMPERVA'S (AND ITS PARTNERS' OR SUPPLIERS') LIABILITY FOR DIRECT DAMAGES HEREUNDER EXCEED THE TOTAL VALUE OF AMOUNTS TO BE PAID BY END USER FOR THE PRODUCT OR SERVICE AT ISSUE. IN NO EVENT SHALL END USER OR IMPERVA (OR ITS PARTNERS' OR SUPPLIERS') HAVE ANY LIABILITY TO THE OTHER FOR ANY LOST PROFITS OR REVENUES, LOSS OF DATA OR USE, INTERRUPTION OF THE SERVICES, COSTS OF PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, OR FOR ANY INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES HOWEVER CAUSED AND, WHETHER IN CONTRACT, TORT OR UNDER ANY OTHER THEORY OF LIABILITY, WHETHER OR NOT THE PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. THE FOREGOING LIMITATIONS SHALL NOT APPLY TO END USER WITH RESPECT TO ANY CLAIMS ARISING UNDER SECTION 2 (LICENSES AND RESTRICTIONS), OR TO EITHER PARTY UNDER SECTION 7 (CONFIDENTIALITY).

11. **Term and Termination**.

a) The term of this Agreement will commence upon Delivery of the Products to End User and will continue in effect for such time as End User continues to have the right to access the Products. Support for Software and/or Appliances will automatically renew at the end of the applicable Support term unless either party gives the other at least thirty (30) days' notice of non-renewal prior to the end of the then current term.

b) Either party may terminate this Agreement due to a material breach of this Agreement by the other party if such material breach remains uncured for a period of thirty (30) days following receipt of written notice by the breaching party; provided that Imperva may terminate this Agreement and/or all licenses granted to End User hereunder immediately upon written notice to End User if End User breaches any provision of Section 2 (License & Restrictions), Section 4 (Additional Terms for Subscription Services) or Section 7 (Confidentiality).

c) Upon the earlier of expiration of End User's rights or termination of the Agreement, Imperva will cease providing Subscription Services, Support and Professional Services, and each party shall promptly return or destroy the other party's Confidential Information. Termination shall not relieve End User of the obligation to pay any fees accrued or payable to Imperva prior to the effective date of expiration or termination. The following sections shall survive termination of this Agreement: Sections 2(c), 2(d), 2(f), 4, 6 -12, and 16.

12. **Compliance with Laws; Export**. End User acknowledges that the Software contains encryption technology that is subject to export restrictions by the U.S. government and import restrictions by certain other governments. End User will not and will not allow any third-party to remove or export from the U.S. or allow the export or re-export of any part of the Software or any direct product thereof: (i) into (or to a national or resident of) Cuba, Iran, North Korea, Sudan or Syria (to the extent the U.S. government or any agency thereof restricts export or re-export to such countries); (ii) to anyone on the U.S. Commerce Department's Table of Denial Orders or U.S. Treasury Department's list of Specially Designated Nationals; (iii) to any country to which such export or re-export is restricted or prohibited, or as to which the U.S. government or any agency thereof requires an export license or other governmental approval at the time of export or re-export without first obtaining such license or approval; or (iv) otherwise in violation of any export or import restrictions, laws or regulations of any U.S. or foreign agency or authority. End User agrees to the foregoing and warrants that it is not located in, under the control of, or a national or resident of any such prohibited country or on any such prohibited party list. The Software is restricted from being used for the design or development of nuclear, chemical, or biological weapons or missile technology without the prior permission of the U.S. government. End User agrees to indemnify and hold Imperva, its partners and suppliers harmless against any claims, losses or expenses arising out of End User's breach of this Section 12.

13. **End User Mention**. End User consents to Imperva using its name and logo to identify End User as a customer of Imperva, such as use on Imperva's web site. Any use shall be subject to Imperva complying with any guidelines that End User may deliver to Imperva from time-to-time regarding the use of its name and logo. This consent terminates upon termination of this Agreement.

14. **Force Majeure**. Neither party will be liable to the other for any delay or failure to perform any obligation under this Agreement (except for a failure to pay fees) if the delay or failure is due to unforeseen events, which occur after the signing of this Agreement and which are beyond the reasonable control of the parties, such as strikes, blockade, war, terrorism, riots, natural disasters, refusal of license by the government or other governmental agencies, in so far as such an event prevents or delays the affected party from fulfilling its obligations and such party is not able to prevent or remove the force majeure at reasonable cost.

15. **Evaluation**.

a) **Evaluation Product**. If the order is for End User to evaluate Product and its related Documentation on a temporary basis for non-commercial use in a non-production environment ("Evaluation Product") and Imperva agrees to such evaluation, conditioned on End User's compliance with the terms and conditions of this Agreement, Imperva grants to End User during the Evaluation Period (as defined below), a cost-free, nonsublicensable, nontransferable, nonassignable and nonexclusive, revocable license to use the Evaluation Product, solely at the location identified in writing by End User and solely for End User's internal evaluation of the Evaluation Product. End User may only grant access to the Evaluation Product to employees, contractors, agents or consultants who are bound to confidentiality and non-use obligations no less protective of Imperva's proprietary rights than this Agreement. Notwithstanding anything to the contrary as stated in this Agreement, all worldwide right, title and interest to the Evaluation Product, and all intellectual property rights in and to them, are and will remain the exclusive property of Imperva.

b) **Evaluation Period**. Unless otherwise agreed to by the parties in writing or terminated earlier in accordance with this Agreement, an evaluation shall commence upon delivery of the Evaluation Product and continue for thirty (30) days thereafter ("Evaluation Period"). Upon the expiration or termination of the Evaluation Period, (i) all licenses granted under this Section 15 for such evaluation will cease, and (ii) End User will immediately return the Evaluation Product to Imperva and destroy or erase any intangible copies of the Evaluation Product, and certify in a writing signed by an officer of End User and delivered to Imperva that all such copies of have been returned, destroyed or erased.

c) **Additional Evaluation Terms**. Notwithstanding anything to the contrary as contained in this Agreement, End User acknowledges and agrees that the Evaluation Product is provided for evaluation "AS-IS" and Imperva and its suppliers make no representations or warranties of any kind, express or implied, with respect to the Evaluation Product, including, without limitation, any implied warranties of merchantability, title, fitness for a particular purpose, informational content, system integration, enjoyment, non-infringement or any other warranties arising out of course of dealing, usage or trade, and no obligation under Section 8.b (Indemnity) shall arise with respect to an Evaluation Product.

16. **Miscellaneous Provisions**. The parties are independent contractors under this Agreement and nothing in this Agreement authorizes a party to act as an agent of the other or bind the other to any transaction or agreement. This Agreement will bind and inure to the benefit of each party's permitted successors and assigns. Neither party may assign or transfer this Agreement in whole or in part by operation of law or otherwise, without the other party's prior written consent. Any attempt to transfer or assign this Agreement without such written consent will be null and void. Notwithstanding the foregoing, however, Imperva may assign this Agreement without consent to the acquiring or surviving entity in a merger or acquisition in which Imperva is the acquired entity (whether by merger, reorganization, acquisition or sale of stock) or to the purchaser of all or substantially all of Imperva's assets. Imperva's licensors are intended third party beneficiaries of this Agreement. In the event any provision of this Agreement shall be determined to be invalid or unenforceable under law, all other provisions of this Agreement shall continue in full force and effect. Except as specifically provided in this Agreement, the exercise by either party of any rights and remedies under this Agreement will be without prejudice to its other remedies under this Agreement or otherwise. This Agreement contains the entire agreement of the parties with respect to the subject matter of this Agreement and supersedes all previous communications, representations, understandings and agreements, either oral or written between the parties with respect to said subject matter. This Agreement may be modified or waived only in a written instrument signed by both parties. A waiver of any breach under this Agreement shall not constitute a waiver or any other breach or future breaches. Notwithstanding the foregoing, if a separate, written and mutually signed agreement for the acquisition of the Products and/or Services exists between End User and Imperva, the terms of that written agreement (excluding any pre-printed terms of any purchase order, confirmation or similar document, all of which will have no effect and will not be considered agreed to by Imperva) shall take precedence over this Agreement. All notices, requests, demands and other communications hereunder shall be in writing to the address set forth below for Imperva and on the applicable order for End User and shall be deemed to have been duly given: (i) upon receipt if by personal delivery; (ii) upon receipt if sent by certified or registered mail (return receipt requested); or (iii) two (2) days after it is sent if by overnight delivery by a major commercial delivery service. This Agreement will be interpreted and construed in accordance with the laws of the State of California and the United States of America, without regard to conflict of law principles. The parties hereby consent to the exclusive jurisdiction and venue of the state and federal courts located in Santa Clara County, California for resolution of any disputes arising out or relating to this Agreement. The provisions of the United Nations Convention on Contracts for the International Sale of Goods and the Uniform Computer Information Transactions Act will not apply to this Agreement in any manner whatsoever.

Imperva, Inc.

3400 Bridge Parkway, Suite 200

Redwood Shores, CA 94065 USA

Table of Contents

Copyright Notice 2
End User License and Services Agreement 3
Overview 22
Introducing SecureSphere Web Application Security 22
Available Licenses for SecureSphere Web Security Products 23
Web Application Firewall 23
ThreatRadar 23
Accurate Protection 23
Transparent Inspection 24
Multi Layer Protection 24
The SecureSphere WAF System 25
SecureSphere Architecture 26
SecureSphere Main Features 26
ADC 27
Dynamic Profiling 27
Discovery 27
Assessment 27
Vulnerability 28
Risk Management 28
Policies 28
Monitoring 28
Reporting 28
Discovering and Classifying Network Assets 30
Introduction to SecureSphere Discovery and Classification 30
Understanding the Discovery and Classification Window 31
Major Discovery and Classification Tasks 31
Using Service Discovery to Populate a SecureSphere Site 32
Working with SecureSphere Scans 32
Creating a Scan 33
Configuring a Service Discovery Scan 33
Running a Scan 37
Managing Discovered Servers 37
Analyzing Discovered Servers 38
Working with Discovered Servers 41
Editing Discovered Servers 41
Manually Accepting and Rejecting Discovered Servers 42
Reinstating Rejected Discovered Servers 42
Basic SecureSphere Configuration 43
Basic Configuration Overview 44
Browser Support 44
Exporting and Importing Individual Objects 44
Basic Configuration Guidelines for Gateways, Management Servers and Agents 45
IPv6 Support 45

IPV6 Address Formats 46
CIDR Notation 47
IPv6 Support in Hostnames 47
IPv4 / IPv6 Dual Stack 47
IPv6 Support in SecureSphere MX and Gateway 47
IPv6 Support in WAF 47
IPv6 Support in Server Groups 48
IPv6 Support in IP Groups 48
IPv6 Support in Active Modules 48
IPv6 Support in Lookup Data Sets 48
IPv6 Support in ThreatRadar Reputation Services 48
IPv6 Support in External Systems 48
Basic Site Configuration 49
Site Best Practices 49
Creating a New Site 50
Renaming Sites 50
Basic Server Group Configuration 50
Server Group Best Practices 51
Basic Server Group Configuration Tasks 51
Understanding Server Group Parameters 54
Server Group Definition Parameters 55
Server Group Services and Ports 57
Server Group Server Information 57
Server Group Applied Policies Information 59
Basic Web Services Configuration 59
Basic Web Service Configuration Tasks 59
Creating New Web Services 60
Understanding Web Services Parameters 64
Managing Web Applications 69
Adding a New Web Application 70
Understanding Web Application Parameters 70
General Basic Configuration Tasks 72
Modifying Service Name and Port 72
Modifying the Character Set 73
Adding SSL Keys 73
Adding Session Identifiers Used for Session Tracking 75
Configuring Action Sets and Followed Actions 75
Activating Settings 75
Monitoring 77
Monitoring Models in SecureSphere 78
Sample SecureSphere User Types 78
Use Case: Small to Medium Business Implementation 79
Use Case: Security Officer Management 79
Use Case: ISP Implementation 79
Understanding Alerts, Events and Violations 80
Monitoring Architecture 83

Monitoring Checklist **83**
Working with the Dashboard **85**
Alert and Event Notification Capacity 86
Dashboard Management Workflow 86
Monitoring Gateway Status 87
Managing Alerts **88**
Viewing Alerts in the Alerts Window 89
Geographic Alert Map 89
Alert Management Workflow 90
Understanding the Alerts Table 91
Analyzing Alerts in the Details Pane 93
Flagging Alerts 97
Creating a Multiparty Review Workflow 98
Taking Action on Alerts and Violations 98
Managing Violations **99**
Viewing Violations in the Violations Window 100
Understanding Violations 100
Analyzing Violations in the Details Pane 102
Managing Blocked Sources **105**
Viewing Blocked Sources 105
Releasing Blocked Sources 106
Web Error Pages and Response Pages in Alerts **106**
Viewing Web Error Pages and Responses 106
Managing System Events **107**
Viewing System Events 107
Understanding System Events 108
Filtering Alerts and Violations **109**
Using the Quick Filter 109
Using the Basic Filter 110
Using the Advanced Filter 112
ThreatRadar Services **113**
Introducing ThreatRadar Reputation Services **114**
Working with ThreatRadar Reputation Services **114**
Disabling and Enabling ThreatRadar Reputation Services 116
All Services 117
Licensing Information **117**
Phishing URLs Service **117**
Phishing URLs Summary 118
Managing the Phishing URLs Service 118
Anonymous Proxy Service **119**
Anonymous Proxy Summary 120
Managing the Anonymous Proxy Service 120
Malicious IPs Service **121**
Malicious IPs Summary 121
Managing the Malicious IPs Service 122
IP Forensics Service **122**

TOR IPs Service **122**
TOR IPs Summary 123
Managing the TOR IPs Service 123
Introducing Community Defense **124**
Community Defense Back Feed **124**
How Security Policies Work with Community Defense 125
Defining Cloud Communication Parameters 126
Managing Violation Data Sharing 126
Understanding Community Defense Summary 129
Community Defense License **130**
Remote File Inclusion **130**
Remote File Inclusion Summary 130
Managing the Remote File Inclusion Service 131
SQL Injection **131**
SQL Injection Summary 131
Managing the SQL Injection Service 131
Risk Management **132**
Introduction to SecureSphere Risk Management **132**
Risk Management Process Overview 133
Understanding the Risk Management Window 134
Scanning and Assessing Web Services **135**
Web Service Vulnerabilities Overview 136
Integrating Web Scanners Task Overview 136
Conducting a Web Scan and Preparing Results 137
Creating and Configuring a Web Scanner Integration Policy 138
Running a Web Scanner Assessment Job 139
Configuring SecureSphere to Work with WhiteHat Sentinel 140
Configuring SecureSphere to Work with QualysGuard 142
Analyzing Risk **143**
CVSS Risk Scoring 143
Analyzing Risk Workflow 144
Examining Risk with Risk Views 144
Analyzing Risk Details 148
Managing Vulnerabilities **150**
Analyzing Discovered Vulnerabilities 150
Vulnerability Management Workflow 151
Introduction to Common Vulnerabilities and Exposures 152
Viewing Discovered Vulnerabilities in Tabular View 153
Viewing Discovered Vulnerabilities in Graphical Views 154
Analyzing Vulnerability Details 156
Managing and Mitigating Vulnerabilities 159
Unmitigating a Blocked Vulnerability 164
Security Policies **166**
Introducing Security Policies **166**
Providing Multi Level Protection Using Security Policies 167
Security Policy Types 167

Correlated Attack Validation (CAV) Used in Correlation Policies ... 170
Intrusion Prevention Used in Signature Policies ... 170
Security Policy Structure ... 171
Directly Configuring Policies from the Security Policies Window ... 171
Using Default Security Policies ... 172
Security Policy Exceptions ... 173
Enabling Response Page in Violations ... 175
Security Policy Main Tasks ... 175
Modifying Policies in the Security Window ... 176
Adding New Security Policies ... 179
Tuning Security Policies in the Setup Window ... 180
Modifying Security Policies in the Monitoring Window ... 180
Filtering Policies ... 181
Legacy Policies ... 182
Adding Comments to Security Policies ... 182
Network Security Policies ... 184
Firewall Policy ... 184
Network Protocol Validation ... 186
Stream Signature ... 188
Securing Web Transport ... 189
HTTP Protocol Validation Policy ... 189
Cookie Signing Validation Policy ... 195
HTTP Protocol Signature Policies ... 197
Web Service Custom Policies ... 198
Web Service Correlated Validation Policy ... 203
Securing Web Applications ... 206
ThreatRadar Fraud Prevention Services ... 206
Anti-Automation Policies ... 206
Web Profile and Web Profile Policy ... 208
Web Worm Policy ... 209
Web Application Signatures Policy ... 209
Web Application Custom Policies ... 210
OCSP Protocol Validation Policies ... 212
Reporting ... 213
Reports Introduction ... 213
Report Terminology ... 214
Understanding the Reports Window ... 214
Working with Reports ... 215
Understanding Report Types ... 216
Working with Alert Reports ... 218
Working with System Event Reports ... 219
Working with Task Reports ... 219
Working with Configuration Reports ... 220
Working with Vulnerability Reports ... 221
Working with Pre-Defined Vulnerability Reports ... 221
Working with Discovery Reports ... 222

Working with Profile Reports 223
Creating and Managing Reports 226
Report Management Main Tasks 226
Creating a Report 227
Configuring Reports 228
Scheduling Reports 235
Configuring Reports With Followed Actions 236
Running a Report 237
Viewing Generated Reports 237
Filtering Reports 238
Basic Filter Criteria 238
Web Application Profiles 239
Understanding Web Profiles in SecureSphere 240
Overview of SecureSphere Web Profiles 240
Web Profile Components 240
Web Profile Structure 241
Profile Modes 241
Web Profile Tasks 242
Viewing Profiles 244
Web Application Profiles 245
URL Patterns 252
Cookie Protection in Web Profiles 252
Displaying Information About a Profile's URLs 256
Tuning Web Profiles 257
Viewing and Editing URLs 258
Deleting All URLs and Folders in a Profile 259
Editing a URL's Methods 260
Editing a URL Parameter Value Type 260
Switching Between the URL Modes 261
Switching the Mode of Multiple URLs 261
Locking a URL Directory 262
Locking a Web Profile 262
Manually Adding a URL 263
Deleting a URL 264
Viewing a URL Pattern 264
Creating a URL Pattern 264
Creating a URL Pattern from an Existing URL 264
Manually Adding a Learned Host 265
Deleting All URLs and Folders in a Profile 266
Web Profile Graphs 266
Overall Graphs 266
Access Patterns 267
Web Profile Optimization 267
Web Profile Optimization Overview 267
Web Profile Optimization Issues 268
Web Profile Policy 275

Tuning a Web Profile Policy 275
Web APU (Automatic Profile Update) 279
Understanding Web APU Profiling Dynamics 279
Preparing for Change 281
Tuning a Web Profile APU Policy 281
Web Profile System Definitions 283
Universal User Tracking 284
Understanding Universal User Tracking 285
Understanding UUT Components 285
Universal User Tracking Architecture 286
DB Profiling and UUT 286
Web Application User Tracking 287
Form-Based User Authentication 287
Certificate-Based User Authentication 288
Configuring Web Application User Tracking 288
Web to Database User Tracking 295
Overview 296
Web to Database User Tracking Configuration 297
Topology Examples 299
SQL Connection User Tracking 300
SQL Connection User Tracking Architecture 301
SQL Connection User Tracking Overview 301
Configuring SQL Connection User Tracking 302
Direct User Tracking 305
Direct User Tracking Architecture 306
Configuring Direct User Tracking 306
Signatures 307
Working with Signatures 307
Signatures Workflow 308
Viewing and Filtering Signatures 309
Viewing a Signature's Details 309
Editing a Signature 309
Enabling and Disabling a Signature 310
Working with Dictionaries 310
Creating a New Dictionary 311
Adding a Signature to a Dictionary 313
Applying a Dictionary to a Policy 314
Standard Dictionaries 314
Advanced SecureSphere Configuration 317
Advanced Configuration Overview 318
Identifying Advanced Configuration Requirements 318
Applying Policies 318
Applying Security Policies to the Objects in the Sites Tree 319
Applying Security Policies to the Objects in the Applications Tree 320
Managing Archive and Key Settings 324
Managing Keys 324

Defining Archive Settings 325
Configuring and Applying Archive Settings 325
Working with Action Sets and Followed Actions 326
Action Sets and Followed Actions Overview 327
Action Set Main Tasks 328
Working with Default Action Sets 329
Creating Custom Action Sets 330
Action Interface Types 330
Configuring Action Interface Parameters 332
Assigning Followed Actions 341
Working with Global Objects 342
Summary of Global Objects 343
Creating New Global Objects 346
Working with Global Ports Groups 347
Configuring IP Groups 347
Configuring Sensitive Data Protection 350
Using Sensitive Data Dictionary Groups 350
Masking Data in Elements 356
Working with Data Types 357
Creating a New Data Type 358
Configuring Windows Domains 358
Configuring the Default Domain Name 359
Configuring Default Domain Credentials 360
Configuring Active Directory Integration for Data Enrichment 360
Generic Dictionary Groups 361
Protecting and Auditing Custom Information with Lookup Data 363
Predefined Lookup Data Sets 366
Creating a Lookup Data Set 366
Configuring a Lookup Data Set 369
Enriching Data 372
Creating an Enrichment Policy 374
Configuring Data Enrichment Policies 375
Configuring Data Enrichment Policy Rules 375
Configuring Data Enrichment Web Policy Rules 376
Introducing Active Modules 383
Change Tracking - External Objects 383
Overview of Change Tracking - External Objects 384
Before Running a Change Tracking - External Objects Policy 384
Creating a New Change Tracking - External Objects Policy 385
Viewing a Change Tracking - External Objects Policy 386
Editing a Change Tracking - External Objects Policy 386
Application Structure Modeling 387
Working with Tasks 387
Tasks Overview 387
Task Types 388
Understanding the Tasks List 388

Task Followed Action Configuration Guidelines ... 389
Creating Review Tasks ... 390
Creating Assignment Tasks ... 390
Working with Open Tasks ... 391
Configuring Preferences ... 392
Configuring User Details ... 392
Changing Passwords ... 393
Configuring User Preferences ... 393
Configuring Integration with Third Party Management Systems ... 394
Configuring Integration with ArcSight ... 394
Configuring Integration with BMC Remedy Service Desk ... 395
Integrating with RSA enVision ... 398
Configuring Integration with Active Directory for User Management ... 403
System Event Policies ... 411
Working with System Event Policies ... 411
Creating a System Events Policy ... 411
Configuring Scheduling ... 413
Understanding the Execution Plan ... 415
Customizing How SecureSphere Handles Web Traffic ... 415
Configuring Web Service Restrictions ... 416
Configuring Restrictions ... 418
Configuring Web Error Page Policies ... 421
Configuring Advanced Web Service Settings ... 424
Configuring OCSP Support ... 424
Configuring Gateway HTTP Proxies ... 426
Configuring Webworm Protection ... 426
Configuring Session Tracking ... 427
URL Rewrite ... 428
Supporting Web Load Balancers ... 432
Validating SSL Client Certificates ... 436
Malware Detection Support ... 439
Configuring Malware Detection Support ... 440
Understanding the List of Local IP Addresses Identified as Infected by Malware ... 448
Modifying Advanced Settings for Malware Detection Support ... 450
Applying Malware Detection Policies ... 454
Reverse Proxies ... 459
Kernel (Non-Transparent) Reverse Proxy ... 459
Configuring a Kernel (Non-Transparent) Reverse Proxy ... 460
Transparent Reverse Proxy ... 462
Configuring a Transparent Reverse Proxy ... 463
MXs Managed by SOM ... 465
Working with SOM ... 465
Registering MXs ... 466
Setting Up SSL with Certificate Authentication ... 466
Setting Up the Other Authentication Options ... 471
Removing MXs from SOM ... 472

Writing Signatures and Regular Expressions **473**
Writing Signatures **473**
Single Part Signatures 474
Multi Part Signatures 474
Absolute Modifiers 474
Relative Modifiers 474
Regular Expressions **475**
Regular Expressions in SecureSphere 475
Using Regular Expressions in Signatures 477
Plugins **478**
Plugins Overview **478**
Order of Plugin Execution **479**
Configuring Plugins **479**
Custom Plugins 479
Regular Expressions 480
Associating a Plugin with a Service and Configuring the Plugin 480
Custom Plugins Reference **487**
Custom Web Plugins 487
Custom Database Plugins 492
Placeholders **494**
Working with Placeholders **494**
Working with Standard Placeholders **495**
Working with Collection Placeholders **495**
Understanding Single Collection Placeholders 496
Understanding the Nested Collection Placeholder 496
List of Standard Placeholders 498
List of Collection Placeholders 521
Rule.dn Associated IDs **525**
About Rule.dns **525**
List of Rule.Dn Associated IDs **525**
ArcSight Placeholders and Syntax **535**
Macros for use with ArcSight Placeholders **535**
Examples of Syslog Events for ArcSight **536**
Regular Security Events 536
Custom Security Events 537
Firewall Security Events 537
System Events 537
CEF Fields **537**
Standard Fields 538
Extended Fields 540
Event Types **547**
About Event Types **547**
List of SecureSphere Events **547**
Filter Criteria and Report Data Fields **574**
Filter Criteria and Report Data Fields Overview **575**
Dataset: Lookup Attribute Usage Example **575**

Alert and Violation Criteria **575**
Assessment Criteria **577**
System Event Criteria **579**
Vulnerability Criteria **579**
Task Report Criteria **581**
Configuration Report Criteria **583**
Configuration - Sites Report Criteria 584
Configuration Security Policy Report Criteria 585
Risk Management Filter Criteria **586**
Vulnerability Management Filter Criteria 586
Assessment Results Filter Criteria 586
Discovery Report Match Criteria **588**
Service Discovery Detailed Report 589
Service Discovery Execution Report 590
Discovery Filter Criteria **590**
Active Module Criteria **592**
Active Module - Change Tracking Details 592
Active Module Pre-configured Objects 593
Active Module Track Value Changes 594
Active Module Track Value Changes Configuration 594
Profile Criteria **595**
Web Profile URL Report 595
Web Profile Cookies Report 597
Web Profile - Learned Hosts Report 598
Web Profile - Susceptible Directories Report 599
Web Profile - Web Application User Tracking Report 599
Custom Policies Match Criteria **600**
Web Custom Policies Match Criteria 601
Enrichment Policy Match Criteria **607**
Web Service Enrichment Policy Match Criteria 607
Report Columns **611**
Columns Available in Alert Reports **611**
Alert Report Available Columns 612
Alert Event Available Columns 613
Columns Available in Assessment Reports **614**
Columns Available in System Event Reports **616**
Columns Available in Vulnerability Reports **616**
Columns Available in Task Reports **619**
Columns Available in Configuration Reports **620**
Configuration Reports - Sites 621
Configuration Reports - Security Policies 622
Columns Available in Active Module Reports **622**
Active Module - Change Tracking Details Report Columns 623
Active Module Preconfigured Objects Report Columns 623
Active Module Track Value Changes Report Columns 624
Active Module Track Value Changes Configuration Report Columns 625

Columns Available in Discovery Reports **626**
Service Discovery Detailed Report Columns 627
Service Discovery Execution Summary Report Columns 629
Columns Available in Profile Reports **630**
Web Profile URL Report Columns 630
Web Profile Cookies Report Columns 632
Web Profile - Learned Hosts Report Columns 633
Web Profile - Susceptible Directories Report Columns 633
Web Profile - Web Application User Tracking Report Columns 634
Pre-Defined Reports **636**
Alert Reports **637**
Active Module Reports **641**
System Events Reports **645**
Alert Aggregation Descriptions **646**
Alert Aggregation Descriptions **646**
Troubleshooting Hashed Users **679**
Glossary **681**
Index 696

CHAPTER 1

Overview

SecureSphere Web Application Firewall secures web applications, providing protection and control over critical data. This chapter introduces SecureSphere Web Application Security and provides a short description of its main capabilities, including:

Introducing SecureSphere Web Application Security 22
Available Licenses for SecureSphere Web Security Products 23
Accurate Protection 23
Transparent Inspection 24
Multi Layer Protection 24
The SecureSphere WAF System 25
SecureSphere Architecture 26
SecureSphere Main Features 26

Introducing SecureSphere Web Application Security

SecureSphere Web Application Security solutions protect Web applications from cyber attacks. SecureSphere continuously adapts to evolving threats and enables security professionals, network managers, and application developers to mitigate the risk of a data breach and address key compliance requirements such as PCI 6.6.



Note: .This user guide includes configuration information for all Web Application security related features. Those features you are not licensed to use will not available in your installation of SecureSphere.

SecureSphere will:

- Model legitimate Web application usage

- Alert or block access requests that:
  - Deviate from normal application and data usage
  - Attempt to exploit known and unknown vulnerabilities
  - Originate from malicious sources
  - Violate corporate policies
  - Are part of a sophisticated multi-stage attack
- Update Web defenses with research-driven intelligence on current threats
- Virtually patch application vulnerabilities through integration with Web application vulnerability scanners, reducing the window of exposure and impact of ad-hoc application fixes.

Available Licenses for SecureSphere Web Security Products

SecureSphere Web Security products come in the following licenses:

Web Application Firewall 23
ThreatRadar 23

Web Application Firewall

The market-leading Web Application Firewall delivers automated protection against current application attacks, including SQL injection, XSS, and CSRF. SecureSphere combines automated application learning with up-to-date protection policies and signatures from the Imperva Application Defense Center to accurately identify and stop attacks. Granular correlation rules, reputation-based security, and a powerful reporting framework complete SecureSphere's superior multi-layer protection. With multi-gigabit inline and non-inline configuration options, SecureSphere offers drop-in deployment and ultra high performance meeting the most demanding data center requirements.

ThreatRadar

As an add-on security service for the Web Application Firewall, ThreatRadar bolsters defense against large-scale automated attacks. ThreatRadar enables timely, real-world protection from known attack sources, such as malicious IP addresses and phishing URLs, as well as identifies source reputation and geographic location for forensics. By transmitting attack source feeds in near real time to SecureSphere WAFs, ThreatRadar can quickly and accurately stop malicious users before an attack can be launched.

Accurate Protection

The accuracy of the web security and monitoring provided by WAF is the key component of the protection of business critical systems. SecureSphere Web Application Firewall protects the data and the transactions passing through web applications and has the ability to understand the application, to adapt to the changes and provide accurate protection against sophisticated attacks.

Understanding the application means learning about its elements: the URLs, the directories, the parameters, the methods used to call these different URLs, and even the cookies that are typically used for session tracking. On top of that WAF learns the expected user behavior in relation to all those elements. Therefore, to accurately protect an application, SecureSphere WAF provides a solution that automatically builds and updates the model of that application over time.

Transparent Inspection

Imperva's Transparent Inspection technology delivers multi-gigabit performance, sub-millisecond latency, and options for high availability that meet the requirements of the most demanding application and database environments. Transparent Inspection makes it possible for SecureSphere to be deployed in minutes with no changes to the web server or any other aspect of the infrastructure.

Multi Layer Protection

The SecureSphere system's protection operates in layers that correspond to the OSI 7-layer model. The firewall corresponds to OSI layers 2 through 4. Protocol Validation and Application Layer Signatures correspond to OSI layer 7, as shown below. Several of SecureSphere's advanced protection processes, such as Profile Evaluation, Web/DB Correlation, and Correlated Attack Detection operate at the level of the application and thus provide protection at what is effectively layer 8 — a layer not defined in the OSI model.

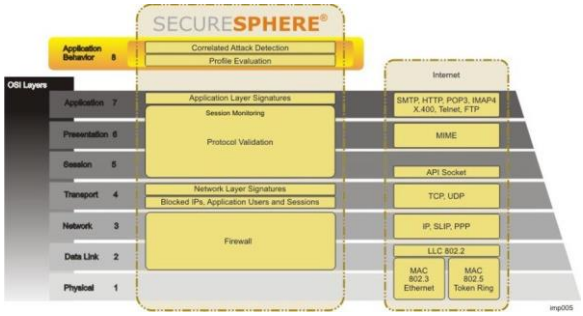
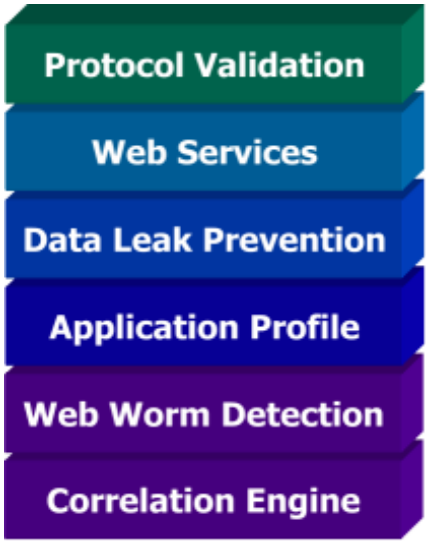


Figure 1: SecureSphere Multi Layer Protection

Another way to present WAF protection layers is illustrated in the following figure.

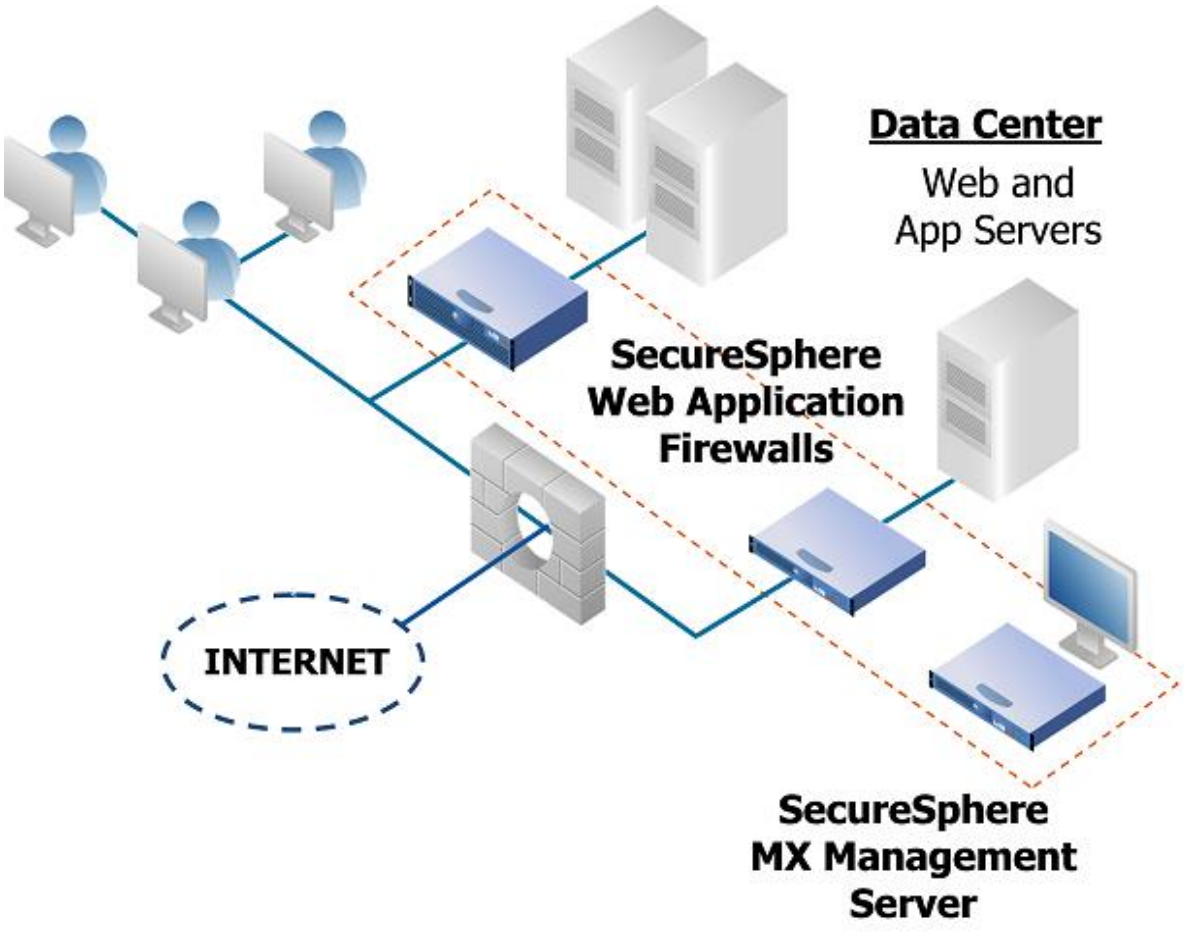


The defense model shown in Figure 2 contains the following protection layers:

- **Protocol Validation**: Filters HTTP protocol violations and attacks that take advantage of the HTTP protocol. For example, an attempt to make a buffer overflow with an abnormally large header of an HTTP request.
- **Attack Signatures**: Identifies known application, platform and network attacks. SecureSphere has a database of over 6500 signatures that's updated weekly by the experts at the Application Defense Center.
- **Data Leak Prevention**: Detects sensitive data such as credit card data, or personally identifiable information, as it is the exiting Web site. Often this is legitimate, but sometimes it represents data leakage. SecureSphere is able to see as sensitive data is exiting the application and administrators can ensure that that's legitimate usage of that data.
- **Application Profile**: SecureSphere compares the actual usage of the application to the expected usage in the model to identify suspicious occurrences or suspicious user behavior.
- **Web Worm Detection**: SecureSphere uses the advanced algorithms to stop Zero-day Web application attacks.
- **Correlation Engine**: event correlation and automated baselining provides SecureSphere WAF with a strong vertical integration and data analysis using the Correlated Attack Verification engine.

The SecureSphere WAF System

The figure below presents the SecureSphereWAF System.



SecureSphere Architecture

The SecureSphere architecture enables you to configure sites which reflect your network architecture, profiles the behavior of traffic running on these sites, then enables you to configure policies to define expected behavior for traffic on these sites. The following figure shows how the concept of application data security is applied by SecureSphere:

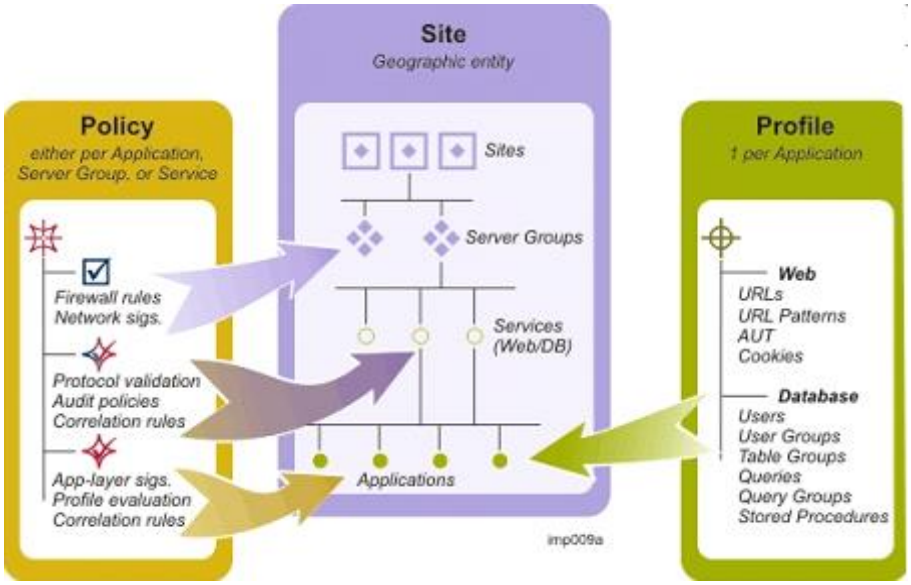


Figure 2: Multi-Level Application Data Protection

The domain protected by SecureSphere contains the following basic elements:

- **Sites**: A physical site where server groups are installed (for example, data centers).
- **Server Groups**: Contain physical servers, services and actual applications.
- **Services**: The HTTP service. A service is characterized by the port used, collection of relevant plug-ins, character sets, encryption and includes a collection of applications.
- **Applications**: Web applications.

Besides the physical representation of the protected domain (using sites, server groups, services, applications), SecureSphere provides a logical representation of the protected domain. The logical representation is focused on the application level, and enables unlimited hierarchical grouping of applications.

SecureSphere Main Features

SecureSphere offers a number of main features that touch every aspect of data center security. Available features include the following:

ADC 27
Dynamic Profiling 27
Discovery 27
Assessment 27
Vulnerability 28
Risk Management 28
Policies 28
Monitoring 28
Reporting 28

ADC

The security and compliance experts at Imperva's Application Defense Center (ADC) ensure that SecureSphere is always up-to-date with the latest defenses against new threats, as well as the most recent regulatory compliance best practices.

ADC research is based on multiple data sources and focuses on the following issues:

- Analyzing the application behavior, whether those are packaged applications like SAP or Oracle E-Business Suite, or custom applications such as PHP or AJAX.
- Collecting information on industry discovered security vulnerabilities and patches released by vendors.
- Providing SecureSphere with the most up-to-date protections, including signature updates, alert profiles, audit profiles and reports. Signatures identify known attacks. ADC maintains signatures of several thousand attacks.

Dynamic Profiling

SecureSphere provides a unique layer of protection based on dynamic profiling. This layer allows the SecureSphere system to detect and protect against threats which are specific to the custom code of the web application. These types of attacks cannot be detected by signature or firewall mechanisms. They require a learning phase in which the product understands the structure of each protected URL. SecureSphere automatically builds these profiles and uses them to detect deviations and various anomalies (or violations) and block attacks on the custom code of the application.

As soon as the system is installed, it starts learning the web usage. After a suitable learning period, the system switches to the protect mode, it starts generating events on usage that does not conform to the rules that it created during the learning period. All this is done in a manner that is completely transparent to users, thereby automating most of the tedious initial deployment tasks and preventing a flood of false positives that is typical when installing security devices. In addition, dynamic profiles can be manually changed, and information can be added and removed to fine tune the profiles and to ensure the system is configured properly.

SecureSphere enables you to define web profiles. These profiles may be built initially based on an application granularity, where an application is defined based on a host name or a list of hosts.

Discovery

The SecureSphere **Discovery Module** identifies services and makes you aware of valuable assets that exist in your network. These steps help you build a map of web services in your network. Furthermore Discovery can be configured to automatically create SecureSphere configuration objects including Server Groups, Services and Table Groups based on the items discovered, or enable you to review and manually approve suggestions.

Assessment

The SecureSphere **Assessment Server** enables you to import scans from third party vendors including IBM AppScan, HP WebInspect, NTObjectives, Cenzic, and WhiteHat, then list vulnerabilities in the SecureSphere vulnerability workbench. SecureSphere integrates Common Vulnerabilities Scoring System (CVSS), maintained by the National Institute of Standards and Technology, which scores each vulnerability on a scale of 0 to 10 based on the effect the vulnerability has and the effort required to exploit it.

Vulnerability

To help manage vulnerabilities discover using assessment, SecureSphere is built with a **Vulnerability Workbench** which lists all vulnerabilities that were discovered during an assessment scan. It rates vulnerabilities by severity, displays various details regarding detected vulnerabilities, and enables you to set properties regarding these vulnerabilities such as status, priority due date, assign them an owner, and much more. Vulnerabilities can also be mitigated. The mitigative action for some vulnerabilities may be to block their communication, while other vulnerabilities you may need to read their description and then take manual action.

Risk Management

The SecureSphere **Risk Explorer** gives you an overall view of the components in your network and the current risk they are exposed to. Information that is displayed in the Risk Explorer is the result of scanning and assessment of services and data discovered in the network, factored together using the Common Vulnerability Scoring System (CVSS) to analyze risk.

This is accomplished by viewing information regarding vulnerabilities that have not been mitigated as they relate to SecureSphere components such as Sites, Server Groups and Servers. The Risk Explorer uses color coding to provide a visual queue as to risk status, and enables you to zoom in on specific types of data and network components to gauge the risk to these components.

Policies

SecureSphere policies provide protection against the identified risks and control of your data.

The protection is provided using security policies. The predefined policies protect against the majority of known attacks and threats. You can use the default policies, modify them or create user-defined policies.

In addition to the security and audit policies, SecureSphere provides **action sets and followed actions**, which define the actions taken by SecureSphere when specific conditions are met. Action sets are configured, then attached as followed actions to various SecureSphere items such as security policies, system events, audit policies, reports, archiving, active modules, tasks and so on.

Monitoring

SecureSphere monitoring keeps you informed of all the events taking place in your system and enables you to understand the risk posed by suspicious activity.

It clearly displays generated information in a central location. Real-time information that is generated includes system events, alerts, violations, blocked sources, gateway status, system warnings, and archiving information.

SecureSphere automatically aggregates related security events, correlating them in intuitive alerts that categorizes activity associating it with known attacks, and informing you of this information. Additionally, the SecureSphere monitor updates you regarding system events such as logging into and out of the system, and system related errors or warnings (for example, predefined thresholds being exceeded).

Reporting

SecureSphere comes with a robust reporting mechanism that enables you to produce pre-defined or user-defined reports regarding data accumulated by SecureSphere. These reports that can be generated either automatically or on-the-fly, or scheduled to run at regular intervals, then distributed as required.

You can use automatic reporting capabilities to implement a workflow that assists in reviewing the most recent and immediate threats on a regular basis. For example, you can schedule SecureSphere to automatically generate a report at the beginning of every week which lists the top ten IP addresses from which attacks are being launched, then automatically e-mails this report to the Security Officer, while at the same time creating a SecureSphere review task that is assigned to the Security Officer. The Security Officer examines that report and can determine whether or not to block any of those IP addresses, then marks the task as closed. Furthermore, the task can be configured to automatically update the manager when the status of the task has changed. This results in a comprehensive workflow that guarantees that specific security threats are addressed on a regular basis.

CHAPTER 2

Discovering and Classifying Network Assets

Identifying the various assets in your network is a crucial step in proactively protecting them.

This section reviews the various aspects of working with SecureSphere Discovery and includes the following topics:

Introduction to SecureSphere Discovery and Classification 30
Understanding the Discovery and Classification Window 31
Major Discovery and Classification Tasks 31
Working with SecureSphere Scans 32
Managing Discovered Servers 37
Working with Discovered Servers 41

Introduction to SecureSphere Discovery and Classification

SecureSphere Discovery and Classification provides a complete set of tools to help you discover web services. It then allows you to use this information to create security policies to monitor them and alert you to suspicious activity.

The following types of discovery and classification are available:

- **Service Discovery**: Service discovery scans your network for open ports and determines the services listening on these ports. For more information on configuring a service discovery scan, see Configuring a Service Discovery Scan on page 33.

Additionally, with service discovery you can configure SecureSphere to automatically create configuration objects based on the items discovered, or enable you to review and manually approve suggestions.

Understanding the Discovery and Classification Window

The Discovery and Classification window provides a wide selection of options that enable you to navigate between the available features to configure scans and display discovered server. The Discovery and Classification window offers a number of main views, as represented by links in the Discovery and Classification navigation bar. These include:

- **Scans**: Lists scans used to discover services and user rights, and classify database data and files. Enables you to create and configure new and existing scans.
- **Discovered Servers**: Displays services discovered by a service discovery scan operating in your network. Enables you to manage these services and add them to your network's SecureSphere architecture.

To view the Discovery and Classification window:

- In the **Main** workspace, click **Discovery & Classification > Scans**. The **Scans** window appears.

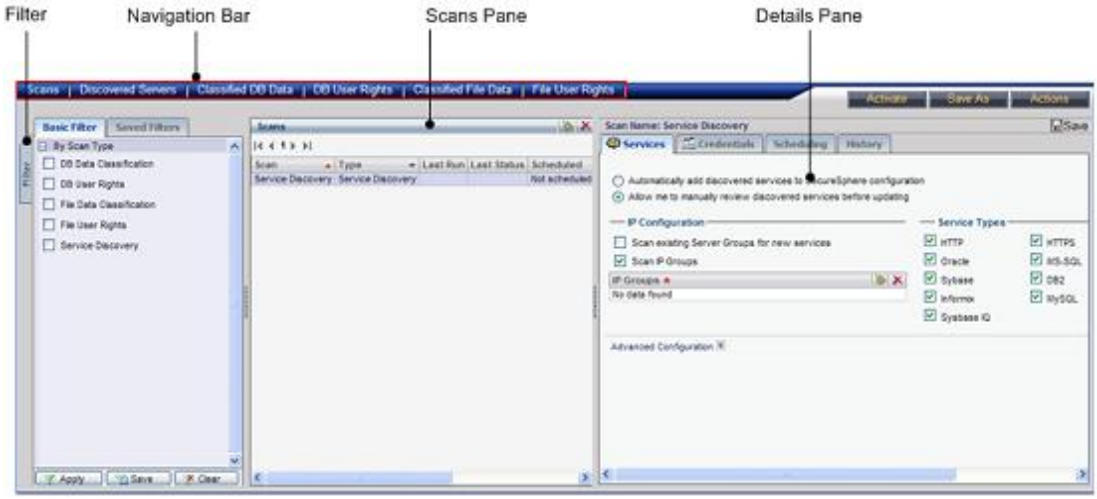


Figure 3: Discovery and Classification Window - Scans

The Discovery and Classification window consists of the following items:

- **Filter**: Enables you to filter scans using various criteria. For a list of available filter criteria, see Discovery Filter Criteria on page 590.
- **Navigation Bar**: Enables you to move between the different parts of the Discovery and Classification windows.
- **Scans Pane**: Displays the scans that have been configured to discover services and classify data in your network.
- **Details Pane**: Enables you to configure scans. For more information see Working with SecureSphere Scans on page 32.

Major Discovery and Classification Tasks

The following table lists the primary tasks involved in working with SecureSphere Discovery.

Major Tasks Overview

	Action	Description	For more information, see...
1	**Creating a Scan**	Create a discovery or classification scan.	Creating a Scan on page 33
2	**Configuring a Scan**	Configure the scan with settings for the specific type of scan you want to conduct.	Configuring a Service Discovery Scan on page 33
3	**Running a Scan**	Run the scan to discover and classify related components in your network.	Running a Scan on page 37
4	**Editing and Adding Components to SecureSphere**	Review discovered components, edit parameters if required and add to SecureSphere architecture.	Working with Discovered Servers on page 41

Using Service Discovery to Populate a SecureSphere Site

SecureSphere service discovery can be used as an alternative method to build a SecureSphere site. Once a site has been manually created, a service discovery scan is configured while selecting the site. It is then run, and as a result SecureSphere automatically creates Sites and Server Groups based on the New Entities configuration that is part of the service discovery scan. If you use SecureSphere discovery to both discover your network assets, and automatically create Sites and Server Groups using the **Automatically Accept New Configuration** option, you can later modify these automatic configurations in the SecureSphere Setup > Sites window. For information on Automatically Accepting New Configuration:

- For **Services**, see Configuring a Service Discovery Scan on page 33

Working with SecureSphere Scans

Discovery and classification scans are the means by which SecureSphere determines what to look for in your network. SecureSphere enables you to create customized default scans to match your requirements and your network.



Note: IPv6 DHCP is not supported for discovery and classification scans.

This section reviews the following subjects:

Creating a Scan 33
Configuring a Service Discovery Scan 33
Running a Scan 37

Creating a Scan

This procedure describes how to create a Service Discovery scan.

To create a Service Discovery Scan:

1. In the **Main** workspace, choose **Discovery & Classification > Scans**. The **Scans** window appears.
2. In the **Scans** pane in the middle of the **Scans** window, click **New**, then choose a scan type, as follows:
   - **Service Discovery**: Configures a service discovery scan to identify the services running in your network. For details on configuring its settings, see Configuring a Service Discovery Scan on page 33.

   The **Create New Scan** dialog appears for the selected scan type.
3. Type a **Name** for the scan.
4. Choose to create the scan from scratch or select an existing scan on which to base the new scan. If creating the scan from scratch select a site for the scan.
5. Click **Create**. The new scan appears in the central selection table.
6. Configure the scan as described in the following: Configuring a Service Discovery Scan on page 33

Configuring a Service Discovery Scan

Configuring a Service discovery scan involves configuring the various options available in the tabs located in the Discovery Scan Details pane.



Note: IPv6 addresses will be scanned, even if they are members of an IP group.

To configure a Service Discovery scan:

1. In the **Main** workspace, select **Discovery & Classification > Scans**. The **Scans** window appears displaying existing scans.
2. Select an existing **Service Discovery Scan** or click **New** and create a new Service Discovery Scan. Options for the selected Service Discovery Scan appear in the pane on the right-hand side of the window.
3. In the **Details** pane, click the **Services** tab. Service discovery options are displayed.
4. Configure settings in accordance with the table below.
5. Click the **Credentials** tab and configure Service credentials options as described in Configuring Service Credential Options on page 35.
6. Configure **Scheduling** if you'd like service discovery to run on a regular basis. For more information on scheduling see Configuring Scheduling on page 413.
7. Click **Save** in the upper right of the screen. Your settings are saved.



Note: This procedure describes how to configure a basic scan. For details regarding available advanced configuration options, see Customizing Service Discovery Advanced Options on page 34.

8. Run the scan. Right-click the scan in the scan window, then click **Run the policy now**. For more information on running a scan, see Running a Scan on page 37.

   For information on reviewing discovered servers, see Analyzing Discovered Servers on page 38.

Service Discovery Scan Options

Option	Description
Automatically add discovered servers to SecureSphere configuration	Automatically adds discovered server groups and services into SecureSphere's configuration. Once new services have been discovered by a Service Discovery Scan, they appear in the Sites > Server Groups information on the Servers tab. **Note:** Discovered IP addresses are not protected automatically. To add discovered IP addresses for protection click the **Add to Protected IPs** button on the Servers tab.
Allow me to manually review discovered servers before updating	Enables you to manually accept or reject discovered server groups and services in the Discovered Servers window.
IP Configuration	Determines how SecureSphere scans your network and resolves host names, operating systems and port version. Options include the following: • Scan existing Server Groups for new services: Enables discovery on server IP addresses already configured in SecureSphere. • Scan IP Ranges: Enables discovery on IP addresses as defined in IP groups. Click New, then select a group of pre-defined IP Groups. For more information on IP groups, see Configuring IP Groups on page 347. **Note:** • IPv6 addresses will be scanned, even if they are members of an IP group. • To create or edit an IP Group, click Edit.
Service Types	Determines the types of services to discover. SecureSphere does not test for types that are not selected.
Advanced Configuration	Offers a number of advanced options for configuring Service Discovery scans including additional ports to scan, how to name discovered services that are added to SecureSphere, and more. For more information, see Customizing Service Discovery Advanced Options on page 34.

Customizing Service Discovery Advanced Options

Service Discovery scans are preconfigured with the basic items that are required to discover services in your network. You can optionally configure advanced options that enable you to resolve hostnames, operating systems and port versions, determine what ports to scan, configure naming conventions for new services and have them automatically added to the SecureSphere site tree, and more. This section reviews these advanced options.

To customize service discovery scans:

1. In the **Main** workspace, select **Discovery & Classification > Scans**. The **Scans** window appears displaying existing scans.
2. In the **Scans** pane, click the scan you want to customize. The scan's configuration options appears in the **Details** pane. The table below details these options.
3. If you have made any changes, click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Service Discovery Scan Advanced Options

Option	Description
Advanced Configuration	• **Resolve Hostnames**: Looks up hostnames during discovery. **Note:** The **Resolve Hostnames** function requires that a PTR record be defined for the host in the DNS server. • **Resolve operating system and ports versions**: Includes full details of the operating system and services in the discovery results. • **Use enhanced scanning**: Additionally scans non-default ports for services like Oracle and MySQL. • **Discovery Timeout**: The time period, in milliseconds, after which the discovery attempt will be considered to have timed out.
Port Configuration	Determines how services are identified according to their associated ports, and the range of ports that are scanned. It is recommended use a port list unless used ports are unknown, in which case it would be better to use a limited range of ports. • **Global Ports list**: Select an existing Global Ports group global object. For more information, see Working with Global Ports Groups on page 347. **Note:** To create or edit the Global Ports list, click Edit. To configure additional ports, add them as entries in port table below the global port list.
New Entries Configuration	SecureSphere creates new configuration objects for discovered servers. This option enables you to determine how these objects are created, as follows: • **Server Group naming template**: Determines naming conventions for creating new server groups from the results of service discovery. Can utilize the site name ($SITE), service IP ($IP), Class A ($CLASS_A_SUBNET), Class B ($CLASS_B_SUBNET), and Class C ($CLASS_C_SUBNET), as placeholders. • **Service naming template**: Determines naming convention for creating new services from the results of service discovery. The template can utilize the site name ($SITE), service IP ($IP), service hostname ($HOST_NAME) and service type ($SERVICE_TYPE) as placeholders.

Configuring Credentials

When working with SecureSphere scans you need to configure credentials in order to access the relevant items that need to be scanned. Types of credentials that need to be configured include the following:

• Configuring Service Credential Options on page 35

Configuring Service Credential Options

Rules can be configured to test OS credentials during the service discovery stage. If successful, the credentials are saved in the service's direct access information and used for assessment tests. Alternatively when running on services that exist in SecureSphere, credentials from the Direct Access Information tab are used. Rules can be configured for individual IP addresses, IP address ranges, or subnets. In addition rules can be based on operating system type and hostname by using the Host field. Each set of credentials are attempted only once to prevent a system from locking up due to unsuccessful attempts.

Notes:

- In order to discover new credentials on existing services, you must enable the Scan Existing option discussed in Configuring a Service Discovery Scan on page 33.
- Credentials cannot be defined for IPv6 addresses.

To configure credential options:

1. In the **Main** workspace, choose **Discovery & Classification > Scans**. The **Scans** window appears.
2. In the **Scans** pane, select the Service Discovery scan you want to configure. That scan's options are displayed in the **Details** pane.
3. In the **Details** pane, click the  **Credentials** tab. Credentials options are displayed.
4. Select the **Run Credential Discovery** checkbox. This tries the credentials when running a discovery scan, and if credentials are valid automatically configures the resulting object with these credentials.
5. In the **OS Credentials** table, click **New**. A new row is added to the **OS Credentials** table.

Note: Alternatively, you can import existing credentials. For more information, see Importing Credentials on page 37.

6. Configure in accordance with the table **Service Credential Options** below.
7. Click **Save** in the upper right of the screen. Your settings are saved.

Service Credential Options

Option	Description
Type	Determines the basic parameters for IP addresses on which to use the credentials. Options include: • **Single**: Uses the credential to attempt discovery on a single IP address. • **Range**: Uses the credential to attempt discovery on a range of IP addresses. • **Network**: Uses the credential to attempt discovery on a network address. • **Any**: Scans all IP ranges configured in the services tab of the selected discovery scan.
Start IP	The first IP address in a range, or only IP address when configuring credentials for a single or network IP address. Use CIDR notation.
End IP	The last IP address in a range.
Host (RegExp)	Not required. Enables configuring a rule to use a single set of credentials for multiple hosts, when a system was used for creating host names.
Auth	Authentication used to connect.
Login	OS login used to connect to the service (required).
Password	Authorized password for the Operating System (required).
Confirm Password	Same as the password.
Order	Used to determine in which order credentials are attempted and as a means of identification when reporting on the success or failure of authenticating an operating system.

Importing Credentials

As opposed to manually configuring credentials, you can import them from a Comma Separated Values (.csv) file. This can save time and trouble. Files for upload should be formatted as follows:

- **Server Credentials**: Should include the following fields: IP, User, and Password. When imported, type is automatically set to single and authentication method is set as SSH.

To upload credentials from a file:

1. In the **Main** workspace, choose **Discovery & Classification > Scans**. The **Scans** window appears.
2. In the **Scans** pane, select the service discovery scan you want to configure. That scan's options are displayed in the **Details** pane.
3. In the **Details** pane, click the  **Credentials** tab. Credentials options are displayed.
4. Click **Upload** from below the relevant table.
5. Browse to the file containing the credentials.
6. Check the option **The File Includes Title** if the file contains a row with the column titles in it.
7. Click **Upload**. The credentials are imported into SecureSphere.

Running a Scan

Once you have configured a scan as required you can choose to run it immediately.



Note: When scanning Solaris systems, tests may take an extended period of time to complete. This is due to Solaris having a built in rate limit for sending RST packets used to determine if the port is open. You can turn off this feature in Solaris using the following command:

```
ndd -set /dev/tcp tcp_rst_sent_rate_enabled 0
```

Alternatively, you can create a group scan to make this change on all Solaris systems used in the organization.

To run a scan now:

1. In the **Main** workspace, choose **Discovery & Classification > Scans**. The **Scans** window appears.
2. In the **Scans** pane, select the scan you want to run. That scan's options are displayed in the **Details** pane.
3. In the **Details** pane, click the **Schedule** tab. Schedule options are displayed.
4. Click **Run the Scan Now**. The discovery scan is run.



Note: You can also run a Discovery scan by right-clicking that scan in the **Discovery Scans** pane and choosing Run the Scan Now.

Understanding Scan History

The History tab displays details of all executions of the scan.

The active icon  is used to identify which execution's results are displayed in the discovery results area. Errors and warnings generated by the scan can be viewed by clicking on the link in the status column.

Managing Discovered Servers

Once a service discovery scan has been run, discovered servers are displayed in the Discovery Results window in which they can be viewed, modified, accepted into the SecureSphere architecture, or rejected. SecureSphere provides both graphic and tabular means of examining discovered servers, then enables you to determine a means of managing discovered servers. This section reviews the various aspects of working with discovered servers including the following:

Analyzing Discovered Servers 38

Analyzing Discovered Servers

Analyzing discovered servers involves viewing the results of service discovery using the various discovery views, then taking action as required.

To view the results of service discovery:

- In the **Main** workspace, choose **Discovery & Classification > Discovered Servers**. Discovered Servers are displayed in tabular format.

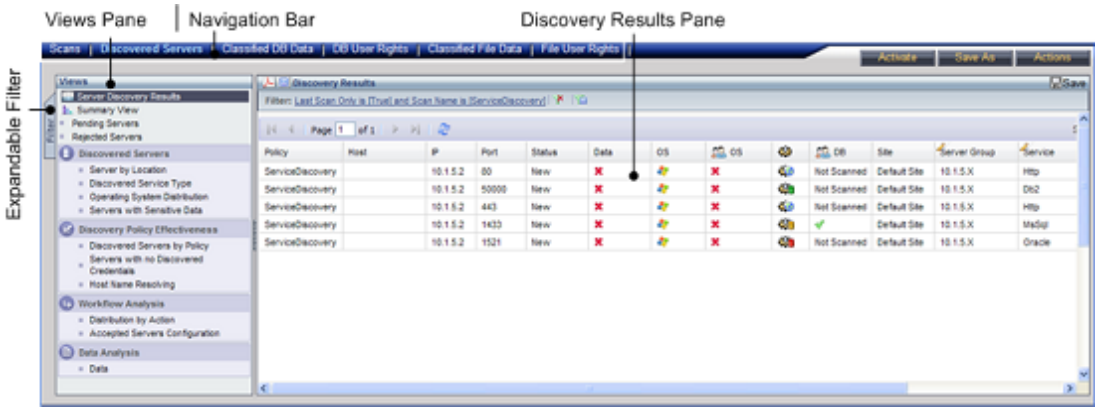


Figure 4: The Discovered Servers Window

The Discovered Servers window displays a summary of discovered servers and their details by default, and enables you to configure a variety of parameters regarding the discovered servers. It additionally offers a number of graphical views with charts that assist in visually analyzing discovery results.

SecureSphere offers two primary ways of viewing discovered servers:

Viewing Discovered Server Details in Tabular View 38

Viewing Discovered Servers in Graphical Views 39

Viewing Discovered Server Details in Tabular View

Discovered servers can be viewed and managed from Tabular view. Various parameters regarding the discovered servers can be configured directly from Tabular view including assigned service and action regarding whether to accept or reject the discovered service. The following table displays the various options available in the Discovered Servers window. For information on how to manage discovered servers, see Working with Discovered Servers on page 41.

Discovered Servers Description

Field Name	Description
Scan	Name of the discovery scan that discovered the services.
Host	Hostname of the discovered service.
IP	IP address of the discovered service.
Port	Port of the discovered service.

Field Name	Description
Status	Whether the service is newly discovered or exists in SecureSphere.
Data	Whether sensitive data was found on the service.
OS	Details of the operating system the service runs on. Hovering over the field displays additional information such as service pack level.
 OS	Whether operating system credentials were configured for this service.
Service Type	Details of the service type found, hovering over the field displays full details of the service.
Site	Site in the SecureSphere site tree the scan was created for.
Server Group	Server group the discovered service is currently. Enables you to assign the service to a different server group if scan is not configured to automatically accept.
Service	Service the discovered service currently belongs to. Enables you to manually edit the service name if scan is not configured to do so automatically.
Date	The date the service was discovered.

Viewing Discovered Servers in Graphical Views

In addition to Tabular view, SecureSphere enables you to view discovered servers in intuitive, easy to read charts that represents the services discovered in your network.

To view discovered servers in graphical format:

1. In the **Main** workspace, choose **Discovery & Classification > Discovered Servers**. The **Discovery Results** window appears. Discovered Servers are displayed in tabular format.
2. From the **Views** pane, select the desired view. For example, **Server by Location**. That view is displayed.

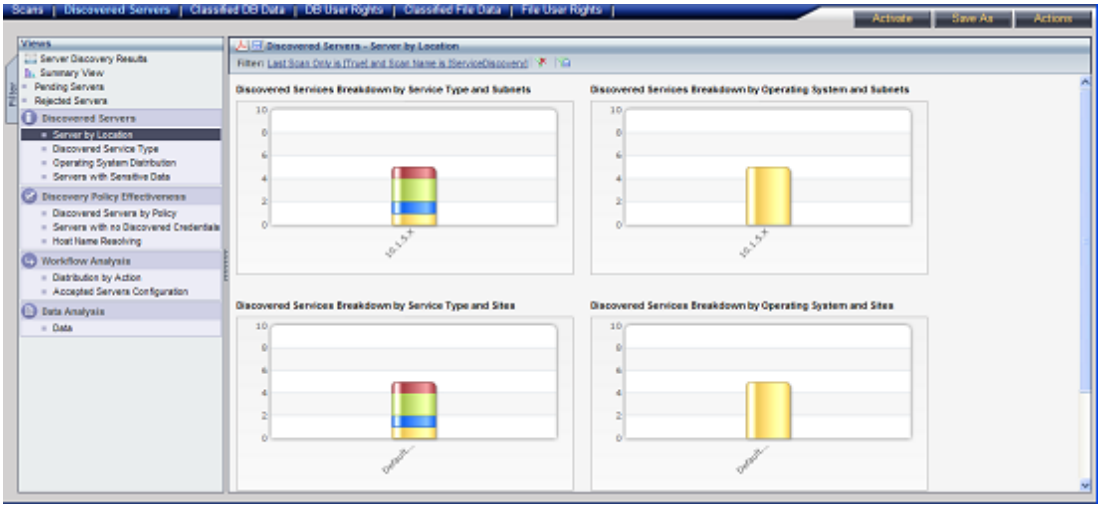


Figure 5: Discovered Servers - Server by Location

SecureSphere has a number of discovered servers views that show data in comprehensible format. Views display data in graphic format using tables and charts. The following table list the views that are available for the specific scan selected in the Scope pane.

Discovered Servers Views

Name	Description
Service Discovery Results	Displays discovered servers in table format and enabled you to configure service names. For more information, see Analyzing Discovered Servers on page 38.
Summary View	Displays data in a number of graphs to assist in understanding discovered basic information regarding these services including IP addresses, Operating System types, services with credentials discovered, and more.
Pending Servers	Displays discovered servers who actions is pending, or in other words, that need to accepted or rejected into SecureSphere. This is done from the Sever Discovery Results window as described inManually Accepting and Rejecting Discovered Servers on page 42.
Rejected Services	Services that have previously been discovered but rejected. For more information see Editing Discovered Servers on page 41.
Discovered Servers	Displays discovered servers based on their location including SecureSphere site and server group. Available views include: **Server By Location**: Shows discovered servers by IP addresses and server types. **Discovered Service Type**: Shows number of IP addresses by service types as well as discovered subnets, ports and versions by service type. **Operating System Distribution**: Shows discovered service IP addresses by operating system as well as discovered subnets, versions and service types by operating system. **Servers with Sensitive Data**: Shows the number of servers discovered with sensitive data by IP address and a breakdown of discovered servers by service type.
Discovered Policy Effectiveness	Displays discovered servers based on the policy which discovered them. Available views include: **Discovered Servers by Policy**: Shows discovered service IP addresses by policy as well as service type, operating systems by policy; and services with sensitive data. **Servers with No Discovered Credentials**: Shows discovered servers by whether or not credentials have been discovered. Shows a breakdown of servers with no credentials by policy name. **Host Name Resolution**: Shows the rate of successful host name resolution and a breakdown of successful resolution by policy name.
Workflow Analysis	Displays discovered servers based on their status as configured in SecureSphere. For example, if a discovered service has been added to the SecureSphere site tree. Available views include: **Distribution by Action**: Shows discovered servers by the action that has been taken on them since being discovered, service types and operating systems by action. **Pending Servers by Age**: Displays discovered servers that have yet to be managed (accepted or rejected) by the amount of time passed since they were discovered. **Accepted Servers Configuration**: Shows discovered servers that have been accepted and the methods by which they were accepted. Additionally shows accepted servers by service type and by operating system. **Rejects Services**: Displays discovered servers that have been rejected.

Name	Description
Data Analysis	Displays discovered service and whether or not data was discovered on the service. Available views include: **Data**: Shows the number of services on which data was discovered by service, the number of services on which sensitive data was discovered.

Working with Discovered Servers

After a discovery scan is run, discovered servers are displayed in the **Discovered Servers** window and can be accepted into SecureSphere's site tree, adding them to the relevant SecureSphere site.

Editing Discovered Servers 41
Manually Accepting and Rejecting Discovered Servers 42
Reinstating Rejected Discovered Servers 42

Editing Discovered Servers

Once discovery has been conducted, discovered items are displayed in the **Discovered Servers** window. You can edit various parameters about the discovered servers.

To edit discovered servers:

1. In the **Main** workspace, choose **Discovery & Classification > Discovered Servers**. **Discovered Servers** and their options are displayed.
2. In the **Views** pane, select **Server Discovery Results**. Discovered servers are displayed. The table **Discovered Server Parameters** below shows the discovered server parameters that can be edited.
3. Modify the parameters as required.



Note: To modify multiple entries at once, select the desired rows and right-click, then choose **Multiple Edit**. Configure the parameters as required.

4. Click **Save**. Your changes are saved, If you accepted a service, it is added to the site tree. If you rejected a service, it is removed from the Discovered Servers window and can be reinstated at a later time if desired. For more information on reinstating rejected services, see Reinstating Rejected Discovered Servers on page 42.

Discovered Server Parameters

Option	Description
Server Group	Enables you to assign the discovered service to an existing SecureSphere server group.
Service	Enables you to modify service name of the discovered service.
Action	Enables you to accept (add it to the site tree) or reject a service that has been discovered. For further information about accepting or rejecting a service, see Manually Accepting and Rejecting Discovered Servers on page 42. **Note:** Rejected services are excluded from future scan runs.

Manually Accepting and Rejecting Discovered Servers

Services that have been discovered can be manually accepted or rejected from the Discovered Servers window. For information on having them automatically accepted, see Using Service Discovery to Populate a SecureSphere Site on page 32.

To accept or reject discovered servers:

1. In the **Main** workspace, choose **Discovery & Classification > Discovered Servers**. **Discovered Servers** are displayed.
2. In the **Views** pane, select **Service Discovery Results**.
3. Select one or more services, then right-click the services, the right-click menu appears.
4. Select the desire option. Services are accepted and added to the site tree, or rejected as desired.



Note: You can additionally accept or reject a service by selecting the desired option from the **Action** column on the right-hand side of the **Discovered Servers** window.

Reinstating Rejected Discovered Servers

Services that have been previously rejected can be reinstated. This enables you to resolve issues where you've mistakenly rejected a service without having to re-run a discovery scan.

To reinstate a rejected service:

1. In the **Main** workspace, choose **Discovery & Classification > Discovered Servers**. The **Discovered Servers** window appears.
2. In the **Views** pane, select **Rejected Servers**. The **Rejected Servers** window opens.
3. In the **Rejected Servers** window, click  reinstate the service. The service reappears in the Discovered Servers window. You can now modify its parameters, accept it or reject it as required.

CHAPTER 3

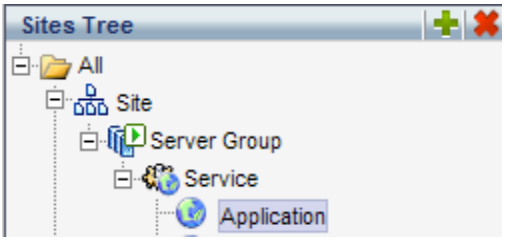
Basic SecureSphere Configuration

This section reviews the tasks required to create basic SecureSphere components and configure them to reflect your network. It reviews the following topics:

Basic Configuration Overview 44
Basic Configuration Guidelines for Gateways, Management Servers and Agents 45
IPv6 Support 45
Basic Site Configuration 49
Basic Server Group Configuration 50
Understanding Server Group Parameters 54
Basic Web Services Configuration 59
Managing Web Applications 69
General Basic Configuration Tasks 72
Configuring Action Sets and Followed Actions 75
Activating Settings 75

Basic Configuration Overview

The primary objective of basic configuration is to identify traffic you want to protect. This is accomplished by defining the basic building blocks that comprise the SecureSphere domain using a number of parameters including IP addresses, ports, protocols and more. The screen shot below illustrates basic configuration of these components.



Basic components are configured in the following order:

1. **Sites**: Defines the primary container of all other objects. For information on creating and managing sites, see Basic Site Configuration on page 49.
2. **Server Groups**: Defines one or more servers at a specific location based on IP addresses. For information on creating and managing server groups, see Basic Server Group Configuration on page 50.
3. **Services**: Defines the service types hosted by a server group based on ports.
   - For information on creating and managing web services, see Basic Web Services Configuration on page 59.
4. **Applications**: The actual application running in your network.

For information about Advanced Configuration, see Advanced SecureSphere Configuration on page 317.

Note: Creation of building blocks is defined in the relevant section. To delete a building block, select it from the Site tree and click . You can delete any item in the Site Tree hierarchy except default applications.

Browser Support

SecureSphere is accessed and managed through an internet browser and also utilizes Adobe Flash. The following are versions of browsers and Flash supported by SecureSphere:

- **Flash Support**: Versions 9 and 10
- **Microsoft Internet Explorer:** Versions 7- 9
- **Firefox:** Versions 3, 3.6 and 19
- **Safari:** version 3.2 and 5.1
- **Google Chrome:** 16 and 25h.

Exporting and Importing Individual Objects

You can export and import individual SecureSphere objects. This capability is useful for copying objects from one SecureSphere environment to another.

Note:

- Most, but not all, SecureSphere objects can be exported and then imported.
- The source and target SecureSphere environments should be running the same version of SecureSphere.

To export a SecureSphere object:

1. Start the SecureSphere GUI.
2. Select the object to export.
3. Click **Save As**.
4. Select **Export** from the menu. A progress bar is displayed in the **Exporting Items** window.
5. When the export is complete, click **Download the file containing...** to save the file.

To import a SecureSphere object:

1. Start the SecureSphere GUI.
2. Click **Actions**.
3. Select **Import** from the menu.
4. In the **Import** window, navigate to the previously-exported file and click **Upload**.
5. After the file is imported, you will be asked to resolve references for individual objects in the file and then click **Import**.

Basic Configuration Guidelines for Gateways, Management Servers and Agents

This section describes a number of practical guidelines of which administrators should be aware.



Note: This section provides general guidelines. Actual maximum values may vary, depending on factors such as your environment, profile size, and traffic volume.

- **Gateways**: The following guidelines apply to SecureSphere Gateways:
  - The number of server groups per Gateway should not exceed **30**.
  - On a Onebox, the number of server groups per Gateway should not exceed **15**.
- **Management Servers**: The following guidelines apply to SecureSphere Management Servers:
  - The number of **SSL certificates** uploaded to a Management Server should not exceed **40K** (key size 1024).
  - The number of **server groups per Management Server** should not exceed **100**.
  - The number of **Gateways per Management Server** should not exceed **15**.

IPv6 Support

SecureSphere supports IPv6 in the following SecureSphere features:

- WAF
- MX-Gateway communications
- VRRP (see the **SecureSphere Administration Guide** for information about VRRP) communications with external systems which support IPv6 (see IPv6 Support in External Systems on page 48)



Note: Additional information regarding IPv6 support is given in the "Configuring SecureSphere" chapter of the **SecureSphere Administration Guide**.

This section describes the support for IPv6 in SecureSphere and contains the following:

IPV6 Address Formats 46
CIDR Notation 47
IPv6 Support in Hostnames 47
IPv4 / IPv6 Dual Stack 47
IPv6 Support in SecureSphere MX and Gateway 47
IPv6 Support in WAF 47
IPv6 Support in Server Groups 48
IPv6 Support in IP Groups 48
IPv6 Support in Active Modules 48
IPv6 Support in Lookup Data Sets 48
IPv6 Support in ThreatRadar Reputation Services 48
IPv6 Support in External Systems 48

IPV6 Address Formats

IPv6 addresses are 128 bits long and are written in eight groups of four hexadecimal digits separated by colons, for example, **2001:0db8:85a3:0000:0000:8a2e:0370:7334**. The hexadecimal digits are case-insensitive.

An IPv6 address can be abbreviated according to following rules:

- Leading zeroes within a 16-bit value may be omitted. For example, the address **fe80:0000:0000:0000:0202:b3ff:fe1e:8329** may be written as **fe80:0:0:0:202:b3ff:fe1e:8329**.
- A single occurrence of any number of consecutive groups of zeroes within an address may be replaced by a double colon. For example, **fe80:0:0:0:202:b3ff:fe1e:8329** becomes **fe80::202:b3ff:fe1e:8329**.

So the same IPv6 address can be represented in several different ways. For example, the following are all equivalent:

2001:db8:0:0:1:0:0:1	2001:0db8::1:0:0:1
2001:0db8:0:0:1:0:0:1	2001:db8:0:0:1::1
2001:db8::1:0:0:1	2001:db8:0000:0:1::1
2001:db8::0:1:0:0:1	2001:DB8:0:0:1::1

Whenever an IPv6 address is entered into SecureSphere, it is displayed in the "long" format once it has been saved. For example, when the IPv6 address as "**2001:DB8::**" is entered in the GUI and saved, it is then displayed as "**2001:0DB8:0000:0000:0000:0000:0000:0000**". This does not apply to `impcfg`, where one of the short forms (not necessarily the shortest) of the address is used.



Note: SecureSphere is able to recognize the equivalence of these different forms of the same IPv6 address except in certain cases where the IPv6 address is treated as a character string, for example, in lookup datasets and in ThreatRadar.

CIDR Notation

Classless Inter-Domain Routing (CIDR) notation explicitly specifies the number of bits in an IP address denoting the network, in contrast to the netmask notation which indirectly specifies the number of bits. For example, a netmask of **255.255.255.0** indicates (for IPv4 addresses) a network in which the first 24 bits of an IP address denote the network. In CIDR notation, one simply writes "/24" after the IP address (for example, **123.124.125.126/24**). The advantages of CIDR notation over the netmask notation are simplicity and scalability from IPv4 to IPv6.

The IP address/netmask notation is no longer permitted for both IPv4 and IPv6 addresses. Instead, CIDR notation is used both in the SecureSphere GUI and `impcfg`.



Note: CIDR notation is used only when specifying a network or when configuring a local IP address where it is necessary to specify the subnet. CIDR is not to be used when specifying a remote IP address, for example, that of an NTP server or a DNS server, or when defining the Management Server of a SecureSphere Gateway. When `impcfg` requires that CIDR notation be used, it specifies so in the prompt, for example, "Enter IP address/CIDR". If "CIDR" is not specified in the prompt, it must not be entered.

IPv6 Support in Hostnames

When entering a hostname, the ":" is not allowed.

IPv4 / IPv6 Dual Stack

The source and destination IP addresses of an IP packet must both be either IPv4 or IPv6. It is not possible for one of the addresses to be IPv4 and the other to be IPv6. The solution is to assign to a device both an IPv4 and an IPv6 address, so that the device can communicate on both protocols.

For example, if an MX manages both IPv4 and IPv6 Gateways, the MX's management or LAN interface (whichever one the Gateways communicate with) must have both an IPv4 and an IPv6 address.

At various points in the SecureSphere GUI and in `impcfg`, users are prevented from trying to define an IPv4-IPv6 connection.

IPv6 Support in SecureSphere MX and Gateway

SecureSphere Management Servers and Gateways can have IPv6 addresses. If an MX's management interface or its LAN interface has both an IPv4 and an IPv6 address, that MX can simultaneously manage Gateways with IPv4 addresses together with Gateways with IPv6 addresses.

A SecureSphere GUI user can point a browser at an IPv6 SecureSphere MX.

IPv6 is supported for Gateway syslog, MX syslog and MX-Gateway communications.

In the **Gateway** window, if the MX's management interface has both an IPv4 and an IPv6 address, then the one the gateway uses is displayed first and the other one is displayed after it, on the same line.

IPv6 Support in WAF

IPv6 addresses can be used in SecureSphere WAF in the definition of server groups.

IPv6 Support in Server Groups

You can mix IPv6 and IPv4 addresses in a single server group. With this restriction: a server group with an IPv6 address (for a Web server) cannot also contain a DB or File server. So it is not possible to:

- Add a server with an IPv6 address to a server group which contains a DB or File server.
- Add a DB or File service to a server group containing an IPv6 address.

IPv6 Support in IP Groups

You can enter both IPv4 and IPv6 addresses in any of the three IP Group types (single, range, and network), though a range can consist of addresses of only one type.

IPv6 Support in Active Modules

IPv6 addresses are not supported for Active Modules.

IPv6 Support in Lookup Data Sets

Both IPv4 and IPv6 addresses can be used in Lookup Data Sets. However, IPv6 addresses are used exactly as they are entered. They are not converted to the long form and SecureSphere will fail to match two identical IPv6 addresses if they are differently expressed: for example, SecureSphere will not recognize that **2001:db8::1:0:0:1** and **2001:db8:0:0:1:0:0:1** are the same.

IPv6 Support in ThreatRadar Reputation Services

ThreatRadar Reputation services feeds currently track only IPv4 addresses. Because ThreatRadar feeds are passed through Lookup Data Sets, the same restrictions apply to ThreatRadar IPv6 addresses.

IPv6 Support in External Systems

SecureSphere MX can communicate over IPv6 with the external systems listed below:

External System	IPv6 Support
HTTP Proxy	Admin > System Definitions > External HTTP Settings
SMTP	SMTP servers with IPv6 addresses.
SNMP	SNMP nodes with IPv6 addresses.
LDAP endpoint access	LDAP servers with IPv6 addresses.
LDAP authentication	
FTP archive	Archiving audit files to an IPv6 archive via FTP.

SCP archive	Archiving audit files to an IPv6 archive via SCP.
Remote Access to Infra to IPv6	Enables remote access to servers, to run shell scripts remotely.

SecureSphere MX **cannot** communicate over IPv6 with the external systems listed below:

External System		Reason
HTTP/HTTPS archive		Admin > System Definitions > Action Interfaces
BMC Remedy		This product is not yet IPv6 compliant.

Basic Site Configuration

Sites are the primary container in which all other network objects are located. Creating new sites enables you to design the topology of your SecureSphere architecture to reflect your actual physical topology. For example, if a site is meant to reflect your finance department, is would be configured with all servers, services and applications used by that department. A site's main purpose is to assist in logically organizing your network structure in SecureSphere, making large systems more manageable. Small networks may only include one or two sites, while large networks many include a dozen or more sites. For information on how to rename a site, see Renaming Sites on page 50.



Note: SecureSphere does not support multi-tab mode. You can access SecureSphere from only a single tab within a specific browser. You can, however, access a single SecureSphere management server from more than one browser at a time.

Information regarding basic site configuration in SecureSphere includes the following:

Site Best Practices 49
Creating a New Site 50
Renaming Sites 50

Site Best Practices

Sites typically represent a physical or geographical layout. By creating sites you create a topology through which you can manage traffic related to the sites. The following are various ways that you may want to lay out your sites:

- **Geography**: For example, perhaps you have operations in New York, San Francisco and Los Angeles. You may want to create sites representing each of these physical locations.
- **Physical Layout**: You may want to create sites to represent actual location in a building such as rooms or floors, or even buildings if your organization operates a MAN. For example, perhaps your network reflects a school campus with a number of buildings, you can create a separate sites to manage resources per building.
- **Environment**: You may want to create sites to represent the various environments running in your network. For example, you could set up a site for both a staging environment and a production environment.
- **Company or Customer**: When used by ISPs to provide added-value services, each site can be created to represent a company, enabling you to exert exacting control and providing specific access rights to each site based on the company.
- **Other**: You can organize sites to represent another logical structure as may be required.

Creating a New Site

Creating a new site enables you to scale the size of your network as reflected in SecureSphere.

To create a new site:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. In the **Sites Tree** pane, click **New**. The **Create Site** dialog box appears.
3. Type a name for the new site.
4. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Renaming Sites

At any time you can rename a site to better reflect your network.

To rename a site:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. Click the site you want to rename. The site is selected and its details appear in the **Sites** window.
3. In the **Name** field, type a new name for the site.
4. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Basic Server Group Configuration

Server groups are a representation of one or more servers located in a specific site. A server in SecureSphere is a set of services or applications belonging to a single IP address. Server groups enable SecureSphere to identify the specific entities at a site that need to be protected, and whose operations you want to track and audit. They also allow you to select the traffic that you want to manage using the following features: Ignore IP, limit monitoring to specific IP addresses, manage applied policies on the server group level, and set server group status (active, simulation or disabled). You may want to create a new server group to provide separate control over a certain set of services, enabling you to move them between actively being monitored to simulation mode without impacting the protection of other services in your network.

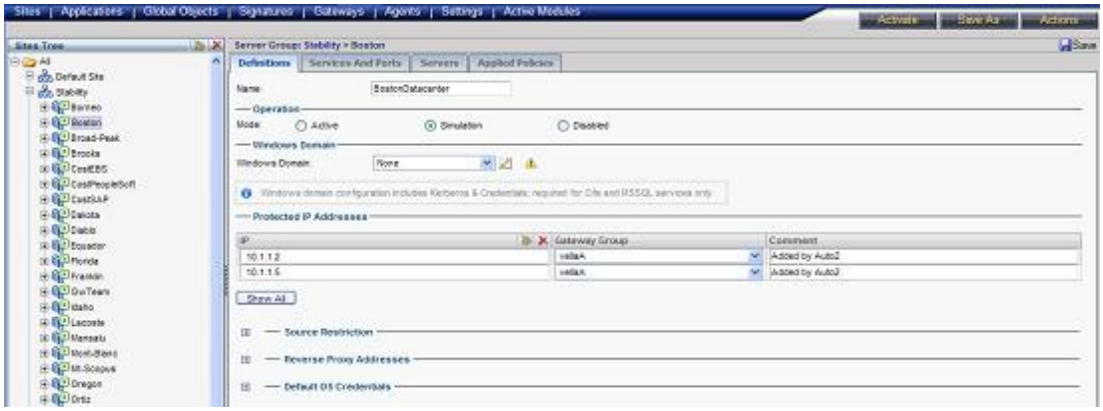


Figure 6: Sites Window Displaying Server Group Definitions

- Information regarding basic group configuration in SecureSphere includes the following:

  Server Group Best Practices 51

Basic Server Group Configuration Tasks 51

Server Group Best Practices

Server groups typically represent the operational aspect to your network components. The following are various ways that you may want to lay out your server groups:

- **Per Gateway**: You should not define more than 30 server groups per gateway.
- **IP Based**: Represents all services related to a single IP address. An IP address can only be assigned to a single server group.
- **Per Application**: Represents a single application running in your network, which enables you to move a server group between operation modes (active, simulation, or disabled) without affecting other applications.
- **Common Code Based**: Represents applications that have a similar code set. For example, you may want to create server groups that run applications based on cold fusion, .NET, or Java. Organizing server groups in this manner reduces profile size and subsequently streamlines SecureSphere operations. Recommended method for maximizing appliance performance.
- **Other**: You can organize server groups to represent another logical structure as may be required. This can also include geography, physical location, environment or company, or customer as described in Site Best Practices on page 49.

Basic Server Group Configuration Tasks

Server groups have a number of parameters that can be configured. To get the server group online, only a handful of items need to be configured. Many can be configured from the server group window once a new server group has been created. The following table lists items that may need to be created or modified to define the content of the server group and its operating mode.

Server Group Configuration Tasks Overview

	Action	Description	For more information, see...
1	**Create a New Server Group**	Creating a server group creates a logical representation of a group of servers in SecureSphere.	Creating a New Server Group on page 52
2	**Configure Simulation Mode**	Simulation is the operation mode configured by default during installation.	Configuring Operation Mode on page 52
3	**Add IP Addresses and Assign to Gateway Group**	Add all the IP addresses used by the various servers in the server group and assign them to relevant gateway groups.	Adding IP Addresses and Assigning to a Gateway Group on page 52
4	**Add Services**	Once basic configuration of the server group has been completed, you need to add services.	Basic Web Services Configuration on page 59

Creating a New Server Group

Prerequisite: Existing site as described in Creating a New Site on page 50.

Server groups are created to enable SecureSphere to group services and applications in logical entities, and facilitate the management of traffic in these entities.

To create a server group:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. In the **Sites** window, click the site in which you want to create the new server group. The selected site information appears in the **Details** pane.
3. In the title bar of the **Sites Tree** pane, click  **New**. The **Create Server Group** dialog box appears.
4. Type a name for the new server group, then click **Create**. The new server group is created and added to the selected site in the **Site Tree**.
5. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

   For an overview of the various parameters that appear in the Server Group window, see Understanding Server Group Parameters on page 54.

Configuring Operation Mode

Operation mode defines the behavior of SecureSphere regarding current traffic. There are three operation modes: active, simulation and disabled. For a more detailed description of the available operation modes, see Server Group Definition Parameters on page 55.

To configure the operation mode:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. Click the server group whose operating mode you want to modify. The server group details appear in the **Sites** window.
3. Click the **Definitions** tab. The Definitions options are displayed.
4. Select the option button of the desired **Operation Mode**. The options include:
   - Active
   - Simulation
   - Disabled

   For a description of the supported operating modes, see Server Group Definition Parameters on page 55.
5. Click **Save**. The settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Adding IP Addresses and Assigning to a Gateway Group

You can manually add IP addresses to be managed by SecureSphere. Add these from the Server Groups definitions tab. This enables you to accommodate additions to your network or add existing server IP addresses for SecureSphere to manage. Additionally, IP addresses can be assigned to relevant Gateway Groups for monitoring. This reduces the overhead involved in monitoring traffic.

To add servers to a server group and assign to a gateway group:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. Click the **Server Group** to which you want to add IP addresses. Server group details appear in the **Sites** window.
3. Click the **Definitions** tab. The Definitions options are displayed.
4. You can specify protected IP addresses either by:
   - importing them from a CSV file (Importing Protected IP Addresses from a CSV File on page 53)
   - adding them individually (Adding Protected IP Addresses Individually on page 54)

Importing Protected IP Addresses from a CSV File

To add Protected IP addresses by importing them from a CSV file:

1. Click **Upload from CSV**.
2. In the **CSV Upload** window, in the **General** tab, type the name of the CSV file in **Select file to upload** or click **Browse** and browse to the file.

   The format of the CSV file should be as follows:

   - If the first line contains only headers, check **Use first line as header** in the **Advanced** tab.
   - In a 32-bit configuration, a maximum of 10,000 Protected IP Addresses can be uploaded at one time.
   - In other configurations, a maximum of 20,000 Protected IP Addresses can be uploaded at one time.

   Note: Importing large number of protected IP Addresses is a computation-intensive task and should be performed during periods of low system activity.

   - Each line should contain a single IP address in either IPv4 or, for Web servers, in IPv6 format.

   Note: You can mix IPv4 and IPv6 addresses in a single server group, with this restriction: a server group with an IPv6 address (for a Web server) cannot also contain a DB or File server. So it is not possible to:

   - Add a server with an IPv6 address to a server group which contains a DB or File server.
   - Add a DB or File service to a server group containing an IPv6 address.

1. In **Gateway Group**, select the gateway group from the dropdown list.
2. In the **Advanced** tab, optionally specify the parameters in the table **Import from CSV Options** below.

   Note: You can specify defaults for the fields in the Advanced tab in Admin > Systems Definitions > CSV Upload Default Settings.

3. You can move the CSV upload job to the background by clicking Move to Background.

   Only one CSV upload job can be running at the same time. You can see previous CSV upload jobs in Admin > Job Status.

Import from CSV Options

Parameter Name	Description
Upload Settings	Select one or more of: Case Sensitive - The data in the CSV file is case sensitive, so if existing data matches the data in the CSV file, they are not considered duplicates. Note: In the case of Protected IP Addresses, this parameter has no effect, since IP Addresses are numbers, including IPv6 addresses which can contain hexadecimal digits. SecureSphere does not distinguish between "a" and "A", for example, in IPv6 addresses. Use first line as header - The first line of the CSV file does not contain data to be imported, but column headings. Override existing entries - If the data in the CSV file matches existing data, then the data in the CSV file will overwrite the exiting data.
Format	Select UNIX or Windows.
Encoding	Select either ASCII or UTF8.

Adding Protected IP Addresses Individually

To add Protected IP addresses individually

1. In the **Protected IP Addresses** pane, click **New**. A new row is added at the bottom of the IP table.



Note: A single IP address can only be added to one server group in SecureSphere. You cannot add the same IP address to multiple server groups.

2. Complete the IP addresses details as follows:
   - Type a valid IPv4 or, for Web servers, an IPv6 address.
   - Select a Gateway Group to which the IP address should be assigned.
   - Type a comment if desired. Maximum number of characters is 4000.
3. Repeat steps 1 to 2 as required.
4. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.



Note: You can mix IPv4 and IPv6 addresses in a single server group, with this restriction: a server group with an IPv6 address (for a Web server) cannot also contain a DB or File server. So it is not possible to:

- Add a server with an IPv6 address to a server group which contains a DB or File server.
- Add a DB or File service to a server group containing an IPv6 address.



Notes:

- Once an IP address has been added, you can configure Direct Access information on the **Servers** tab.
- IP addresses can also be added directly from the Servers tab., though to protect traffic on IP addresses added in this way you must then click **Add to Protected IPs** on the **Servers** tab.

Understanding Server Group Parameters

Server group parameters enable SecureSphere to understand exactly what traffic to monitor, the current mode of the server group, and other details regarding the servers operating in the group.

To see an example of the Sites window displaying server group parameters, see Basic Server Group Configuration on page 50.

SecureSphere server group options include the following:

Server Group Definition Parameters 55
Server Group Services and Ports 57
Server Group Server Information 57
Server Group Applied Policies Information 59

Server Group Definition Parameters

Defines basic information used to identify server groups and manage their operation.

Server Group - Definitions Parameters

Parameter Name	Description
Name	Defines the name of the server group. Enables you to modify a group name.
Operation (Mode)	Determines the operation mode of the server group and that of all defined services and applications belonging to the server group. The three available operation modes are: **Active**: Configures SecureSphere to actively monitor traffic and apply policies. This means that alerts are generated and traffic is blocked when required by a policy. **Simulation** (default): Configures SecureSphere to simulate monitoring. It monitors traffic and generates alerts and violations, but does not prevent traffic from reaching its destination (block). It is recommended to use simulation mode when SecureSphere is learning traffic. **Disabled**: Suspends all monitoring and blocking activities. Can be used to help troubleshooting. Traffic flows uninterrupted through SecureSphere. Disabled mode can be used as a temporary fix to overcome problems with configuration that generates a hail of alerts or blocks sources that need to have access to the network. Use disabled mode carefully. For instructions on how to modify the operation mode, see Configuring Operation Mode on page 52.
Windows Domain	You can create a new Windows domain or edit an existing one by clicking . **Configure Windows Domain** dialog opens: **Settings** • **Domain Credentials**: For authorizing a user for purposes of User Rights Management. • **Active Directory Settings**: For synchronization of users and other data transferred to MX for audit purposes, alerts, reports, etc. • **User Data Enrichment**: For providing additional user information extracted from the active directory such as role, email, department, etc. Appears in SecureSphere in alerts, reports, and audit data. **Kerberos**: For retrieving the Kerberos keys
Source Restrictions	Defines the source IP addresses to which monitoring is to restricted. The options are: **Ignore IP Group**: Attaches an IP group global object with pre-configured IP addresses that belongs to the selected server group, and excludes traffic from this IP group from being monitored. For example, you can select the internal address IP group, which stops all monitoring and enforcement on traffic belonging to internal IP addresses. You can create a new IP group or edit an existing one by clicking . **Limit Monitoring to IP Group**: Attaches an IP group global object which restricts SecureSphere monitoring to only IP addresses belonging to this group. This can be useful before going into production by assisting SecureSphere to focus on traffic from a limited group of IP addresses, teaching SecureSphere standard behavior for this particular site.You can create a new IP group or edit an existing one by clicking . For further information on configuring source restrictions, see Configuring Web Service Restrictions on page 416.
Reverse Proxy Addresses	Lists IP addresses used by SecureSphere in Reverse Proxy mode. If Reverse Proxy mode is not configured for the server group, no IP addresses are listed. For more information, see the **SecureSphere Administration Guide**.
Protected IP Addresses	Defines IP addresses associated with the current server group. For instructions on adding IP Addresses, see Adding IP Addresses and Assigning to a Gateway Group on page 52.
Default OS Connection	Configures credentials required for accessing the Server OS.

Server Group Services and Ports

Defines service and port numbers for applications running in the server group and subsequently monitored by SecureSphere.

Server Group - Services and Ports Parameters

Parameter Name	Description
Port Mapping Definition	Attaches a Global Ports Group global object that maps TCP/UDP ports to their associated protocols, as used in your network. This list is subsequently used to identify and monitor traffic. SecureSphere comes with a default port mapping list. While the default list cannot be directly modified, you can create a port list based on the default to include special port assignments as may be required in your network. For further information about configuring port mapping, see Working with Global Ports Groups on page 347.
Supported Applicative Services	Lists services that have been created in the SecureSphere server group. If no services have been configured, this section is empty.

Server Group Server Information

The Servers tab presents server information for each server group and enables you to add IP addresses for assessment and discovery.



Note: IP addresses here are not automatically assigned to gateways, automatically profiled, or monitored. In order to add IP addresses listed here to a gateway for profiling and monitoring, click the **Add to Protected IPs** button.

Server Group Server Information Parameters

Name	Description
IP (Address)	IP address assigned to the server.
Host Name	Host name assigned to the server during SecureSphere configuration.
OS	Operating system running on the server.
Credentials	Indicates if credentials have been identified by SecureSphere discovery.
Kerberos	Indicates if a matching Kerberos machine account exists for this server host name.
Discovered	Indicates if the IP was discovered or added manually.
Protected	Indicates if the IP is protected.
Creation Date	Date the server was created.
Scan Proxy	To specify that scans of this server should be run using an SSH Proxy, select the SSH Proxy from the dropdown list. Select **No Proxy** to specify that scans will be run directly by the MX. **Note:** Scanning via an SSH proxy is required when, because of the network topology, the MX cannot access the servers to be scanned. SSH Proxies are defined in **Admin** > **System Definitions** > **External Systems**. See the Systems Definitions chapter in the SecureSphere **Administration Guide** for more information.
Windows Domain Definitions	Expand a row to display this option. If you select **Override Default Domain Credentials**, enter the credential details.
OS Credentials	Expand a row to display this option. If you select **Override Default OS Credentials**, enter the credential details. Overriding default OS credentials is required for the Local Users option.
Local Users	(NFS only) Expand a row to display this option. Before you select **Retrieve Local Users**, you must override default OS credentials.

Server Group Servers Tab Function Buttons

Name	Description
Show All	Opens a separate window that displays details of all the servers if more than 50 IP addresses are configured.
Resolve Hostnames	Click to find the host names for IP addresses that are displayed without host names. Resolving host names is required for using the Kerberos Definitions button.
Kerberos Definitions	Opens the Configure Windows Domain window. On the Kerberos tab, you can configure how to automatically obtain passwords from the domain controller. See Configuring Kerberos Support. Before you click **Kerberos Definitions**, you must resolve host names.
Add to Protected IPs	Adds the IP address in the selected row to the Protected IP Addresses table. See Server Group Definition Parameters on page 55.

Server Group Applied Policies Information

When adding a new server group to the site tree, this server group automatically comes with default policies applied to it, see Using Default Security Policies on page 172. You can modify the applied policies. In the Applied Policies tab you can only view the applied policies. Adding of new policies and applying other policies to an object is done in the Security window.

Server Group - Applied Policies Information Parameters

Definitions	Default Policies Applied to Server Group
Basic Security Policies	Options include: **Firewall Policies**: Policy used to monitor IP and port traffic. **Network Protocol Violations**: Policy used to monitor network protocol traffic such as TCP/IP. For instructions on how to turn off policy enforcement for these components, see Tuning Security Policies in the Setup Window on page 180.
Additional Security Policies	Enables applying additional security policies on the server group level. The Stream Signature policy is configured by default. You can add additional policies as required. These policies must first be configured before adding. For information on configuring security policies for server group components, see **Using Default Security Policies**.

Basic Web Services Configuration

Web services in SecureSphere are the web services installed and running on servers in a server group whose traffic SecureSphere monitors. Adding a web service defines the services monitored for IP addresses defined in the server group. This is a required step for basic configuration, depending on the service type being monitored.



Note: A Web server group can contain either IPv4 or IPv6 addresses.

Information regarding basic web services configuration in SecureSphere includes the following:

Basic Web Service Configuration Tasks 59
Creating New Web Services 60
Understanding Web Services Parameters 64

Basic Web Service Configuration Tasks

Web services have a number of parameters that can be configured. To get the web service online, only a handful of items need to be configured. Many can be configured from the web service window once a new web service has been created. The following table lists items that may need to be created or modified to start learning the application structure and profile the protected web servers, URLs, host names, login URLs, and more.

Configuring Web Services Task Overview

	Action	Description	For more information, see...
1	**Create a New Web Service**	Creating a web service creates a representation of the service in SecureSphere.	Creating New Web Services on page 60
2	**Configure Name and Ports**	Service name and ports are configured with values when a single web service is defined. You may want to modify them to reflect your server's configuration.	Modifying Service Name and Port on page 72
3	**Configure Character Set**	Default character set is automatically configured when a web service is defined. It needs to accurately reflect the character set used by your server.	Modifying the Character Set on page 73
4	**Add SSL Keys**	If your server hosts encrypted traffic, SSL keys need to be added to SecureSphere. It is recommended that you obtain the private key, certificate and its password for SSL traffic from the relevant party before getting started.	Adding SSL Keys on page 73
5	**Configure Host/URL to Applications Mapping (optional)**	Mapping host names or URLs to actual applications installed on a server allows SecureSphere to learn and profile their web hosts as individual entities. This improves control and the ability to set different policies for each host.	Configuring Host/URL to Application Mapping on page 63
6	**Configure Default Error Page (optional)**	The default error page provides a generic error page when a block action is used while the server group is active mode. This enables you to supply a meaningful error page and prevent data leakage of internal server information.	Configuring the Default Error Page on page 61
7	**Activate Settings**	Depending on the currently configured SecureSphere activation mode, you may need to activate settings once they have been made.	Activating Settings on page 75

Creating New Web Services

Prerequisites: Existing server group as described in Creating a New Server Group on page 52.
Web services are created in SecureSphere to identify web traffic that needs to be profiled and protected by SecureSphere.

To create a new web service:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. In the **Sites Tree**, click the **Server Group** in which you want to create the new web service. The **Server Group** is selected and its details appear in the **Sites** window.
3. In the **Sites Tree** title bar, click **New**. The **Create Service** dialog box appears.
4. Type a name for the service.
5. From the dropdown menu, select **HTTP Service**.
6. Click **Create**. The new service is added to the server group as it appears in the **Sites Tree** together with a default HTTP application**.**

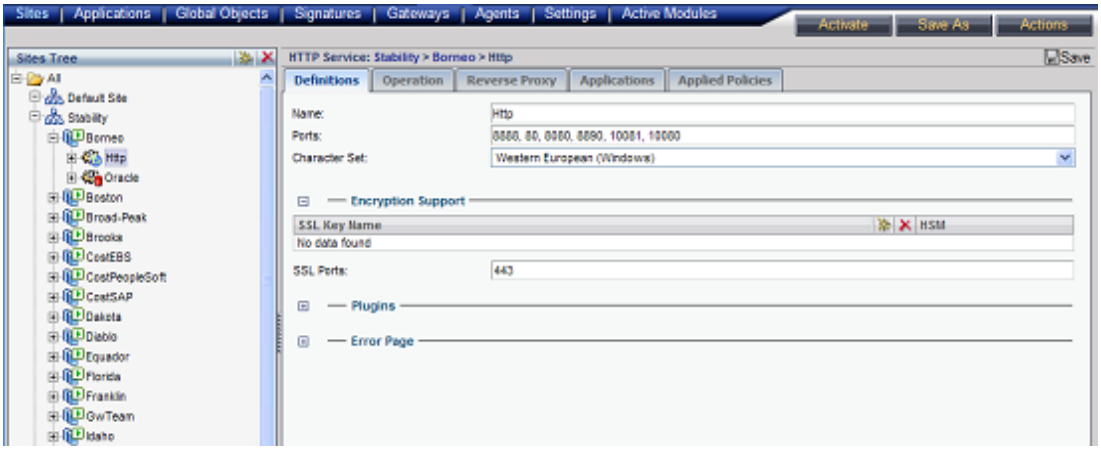


Figure 7: Sites Window Displaying Web Service Definitions

Note: Multiple services can be defined under the same server group with different ports to monitor. Multiple ports can be defined under the same service.

- For an overview of the various parameters that appear in the Web Services window, see Understanding Web Services Parameters on page 64.
- For information on configuring the default error page, see Configuring the Default Error Page on page 61.
- For information on configuring Host/URL to Application Mapping, see Configuring Host/URL to Application Mapping on page 63.

Configuring the Default Error Page

The default error page is a generic error page that is displayed to users when they try to access a forbidden page. This enables you to supply a meaningful error page and prevent data leakage of internal server information. Instead of receiving an error message such as "The system is not responding," you can provide users with meaningful explanations and actionable information. Some examples of information you may want to include in your default error page include:

- What went wrong, for example, a quarantine page saying: "You have been blocked for 1 hour."
- What users can do about correcting the problem, for example: Contact Customer Care Error # 3582." This would enable customer care personnel to use the Event ID to identify the alert that caused the violation.
- A general message such as: This URL is allowed for internal access only.



Notes:

- You may want to consider not using an error page. A lack of any response may prevent malicious users from being able to determine if a specific service is online.
- These placeholders can be used to search for alerts and violations. Please note while Session_ID and Host are available in both alerts and violations, Event_ ID is only available in alerts.

To configure the default error page:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. In the **Sites Tree**, expand the relevant site and server group, then click on the web service you want to configure. Web service options appear in the **Sites** window.
3. In the **Sites** window, click the **Definitions** tab. Definitions tab options are displayed.
4. Expand the **Default Error Page** option, then verify the **Use Default Error Page** checkbox is selected.
5. Select and configure one of the following:
   - **Redirect**: Enables you to redirect users to any URL.
   - **Page**: Enables you to customize a default message using html and placeholders displayed when users encounter errors.
   - **HTTP Response Code**: Enables you to configure a response code which is provided to users with the default error page. Response codes must be in the format of a 3 digit number and need to include a descriptive string. It is recommended that you conform with standard status codes as described in RFC2616. For example, you may want to use 200 OK (default in SecureSphere) to let users that no error has actually occurred. You may alternatively want to use a code like 403 Forbidden or 404 Not Found.
6. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

EXAMPLE: ERROR PAGE CONFIGURATION EXAMPLES

The following are examples for configuring a redirect and error page.

Error Page Configuration Examples

Example Type	Example Syntax
Redirect Example	http://$(HOST)/errorpage.html
Error Page HTML Example	<html><header><title>Error</title></header> <body><H2>Error</H2> This page can't be displayed due to a security violation. Contact support for additional information. Event id: $(EVENT_ID) Session id: $(SESSION_ID) </body></html>

Configuring Host/URL to Application Mapping

A single SecureSphere web application may protect a number of different web hosts and related URLs. For example, your company's main website may have the hostname www.site.com, while its support website has the hostname support.site.com, with both of these sites being different web hosts belonging to the same SecureSphere application. SecureSphere enables you to map these hosts individually, enabling you to profile web traffic and apply policies on a per host basis. Additionally, SecureSphere enables you to define an application based on the combination of a host name and URL by either including or excluding groups of URLs, providing exacting control over what traffic is profiled and monitored, and resulting in detailed information in reports and audits.



Note: Configuring Host/URL to Application mapping is recommended, however it is also optional. If mapping is not configured, all hosts are mapped to the default application.

Configuring host/URL to application mapping involves the following two procedures:

1. Adding a New Web Application on page 70
2. Mapping Host or URL to a Web Application on page 63

Mapping Host or URL to a Web Application

Mapping hosts or URLs to a web application enables SecureSphere to identify specific HTTP hosts with part or all of their URLs as belonging to a SecureSphere web application. By default, all hosts that are not mapped to a specific web application are assigned to the default application. To map a host to an application other than the default application, you need to create a new web application.

Additionally, SecureSphere enables you to create groups of URLs, enabling you to define what traffic to profile and monitor based on URL. You can assign URLs to a group, then decide whether to include or exclude these URLs from being mapped, and subsequently profiled and monitored.

To configure host/URL to web application mapping:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. In the **Sites Tree**, expand the relevant site and server group, then click on the web service you want to configure. Web service options appear in the main window.
3. In the **Sites** window, click the **Applications** tab. Application tab options are displayed.
4. Click  **New**. A new row is added to the **Host** table.
5. In the empty **Host** row, type the name of the host to be mapped, either exactly as it appears (its full name), its prefix, or its suffix. This enables you to define a group of hosts by prefix or suffix. For example, if you type **support**, (as in the URL support.company.com), then in step 6 select prefix; all URLs that belong to the support prefix are mapped to the selected application.
6. From the **Host Match Type** dropdown menu, select one of the following:
   - **Exact**: Configures a complete URL for mapping to the selected application.
   - **Prefix**: Configures all hosts with the entered prefix for mapping to the selected application.
   - **Suffix**: Configures all URLs with the entered suffix for mapping to the selected application.
7. From the **Include URLs** dropdown menu, select a previously configured URL Prefixes/Directories Group global object if desired. This **includes** all URLs to be profiled, monitored and reported.
8. From the **Exclude URLs** dropdown menu, select a previously configured URL Prefixes/Directories Group global object if desired. This **excludes** all URLs from being profiled, monitored or reported on.



Note: For further information on URL Prefixes/Directories Group global object, see Configuring URL Prefixes / Directory Based Restrictions on page 420.

9. Enter a priority for the host mapping. This determines the priority of the current mapping in case of conflict or overlap with another mapping.
10. From the **Application** dropdown list, select the application you want to map to the host.

11. Repeat steps 4 - 10 for each URL group you want to map.
12. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

EXAMPLE: HOST / URL TO WEB APPLICATION MAPPING - E-COMMERCE WEBSITE

The website has a handful of sensitive URLs that need to be closely monitored including the login, shopping cart, checkout and registration pages. It additionally has dozens of other pages with product descriptions which do not need to be monitored. You can create two URL groups, one for the sensitive URLs, the other for the non-sensitive URLs. You can then include sensitive URLs while excluding the group of non-sensitive URLs. This assists in fine tuning SecureSphere operation, reducing alerts and providing a higher level of monitoring precisely where you need it.

Understanding Web Services Parameters

Web service parameters enable SecureSphere to understand what web service traffic to monitor, currently installed plug-ins and SSL keys, and more.

SecureSphere web service options include the following:

Web Service Definition Parameters 64
Web Service Operations 65
Web Service Reverse Proxy 68
Web Service Applications 68
Web Service Applied Policies Information 69

Web Service Definition Parameters

Defines basic information used to identify web services and subsequently traffic related to those services.

Web Services - Definitions Parameters

Parameter Name	Description
Definitions	Defines basic aspects of web service configuration including: **Name**: Name of the web service. Required parameter. **Ports**: Ports used by the web service that are parsed. Default 80. Minimum: 0. Maximum: 65535. **Character Sets**: Default web service character set. Default: Western European (Windows). For instructions on how to modify name and ports, see Modifying Service Name and Port on page 72. For instructions on modifying character sets, see Modifying the Character Set on page 73.
Encryption Support	Enables the management of SSL keys required to decrypt data and includes: **SSL Key Names**: Lists currently installed SSL keys. For information on adding SSL keys, see Adding SSL Keys on page 73. **SSL Ports**: Lists ports currently configured for SSL traffic. Multiple ports can be entered using a comma to separate values. **Manually Configured Kerberos Keys**: Enables you to associate a service with Kerberos keys. For more information, see (Recommended) Associating a Service with Kerberos Keys.
Plug-ins	Lists currently installed plug-ins. Plug-ins provide extended capabilities to SecureSphere operation. For information on plug-ins, see Plugins on page 478.
Error Page	Lists the configuration of the Default Error Page, used to provide a customized error page to users and prevent the leakage of sensitive data using the following: **Use Default Error Page**: Configures SecureSphere to display the default error page for access violations. **Redirect**: Redirects traffic to a pre-configured URL as the result of a policy violation that causes a block using the 302 redirect status code on behalf of the protected server. **Page**: Enables you to customize the content of the web page using html and pre-defined placeholders. **HTTP Response Code**: Determines the status code assigned to the default error page. For information on configuring the default error page, see Configuring the Default Error Page on page 61.
Web Error Page Policies	Determines which error page to send to the user based on a violation, providing meaningful data to users. This may include a default error page, the masking of sensitive data, and more. For information on configuring Web Error Page Policies, see Configuring Web Error Page Policies on page 421.

Web Service Operations

Defines items used by the web service to assist SecureSphere in determining what traffic to profile and monitor, guidelines for masking data, error pages, advanced settings, and more.

Web Services - Operations Parameters

Parameter Name	Description
User Tracking	Session Tracking Lists the currently configured session tracking list which contain session IDs used to track sessions. Enables you to attach different tracking lists. For further information on tracking sessions, see Adding Session Identifiers Used for Session Tracking on page 75. **Session Identifiers**: Attaches a Session Identifier Group global object that lists session identifiers currently used in your network to guide session tracking. **Web Application User Tracking**: Selects the mechanism by which SecureSphere conducts user tracking.
Restrictions	Lists currently configured monitoring restrictions (items that SecureSphere does not monitor). There are a number of elements that pose no threat to the network or data. By excluding them from monitoring you reduce SecureSphere overhead. Available restriction items include: **Ignore Parameters**: Attaches a Parameter Groups global object, consisting of parameters for SecureSphere to ignore. You can edit the default list or create a new one, see Configuring Parameter Groups on page 416. Use this option to select a group of parameters that you want to ignore. The parameters that appear in the Parameter Groups list are not examined by the gateway and the following violations are not invoked for these parameters: Parameter Read-Only, Violations, Parameter Type Violation, Parameter Unknown, Parameter Value Length Violation, Required Parameter Not Found. **Ignore Soap Elements**: Attaches a SOAP Element Group global object consisting of soap elements for SecureSphere to ignore. You can edit the default list or create a new one. For further information on configuring restrictions, see Configuring Web Service Restrictions on page 416. **Note:** Ignoring static files is defined in **Profiling URLs without parameters** under **Advanced Settings** in the web application's **Definitions** tab. See Web Application Definitions Parameters on page 71.
Data Masking	Lists elements currently configured to mask sensitive data. Enables you to customize data masking using the following options: **Mask Data in Elements**: Attaches a Data Masking Group global object used to define data in parameters, headers and cookies, and subsequently uses this list to mask data. You can edit the default list or create a new one. **Sensitive Dictionary Groups**: Attaches a Sensitive Data Dictionary Group global object used to define specific patterns such as credit card and social security numbers, and subsequently uses this list to mask sensitive data. You can edit the default list or create a new one. For information on configuring data masking, see Configuring Sensitive Data Protection on page 350.
Forwarded Connections	**Identify real client ID according to HTTP forwarding header**: Enables SecureSphere to identify the real client when monitoring packets from a forwarded connection. This feature is used in configurations that include a load balancer, it enables presenting of actual source IP addresses and not only the balancer's address. For more information, see Configuring Forwarded Connections on page 434.

Parameter Name	Description
Advanced HTTP Settings	Configures advanced rules that tell SecureSphere how to treat specific information types or formats when encountered including: **Learn JSessionID parameters as regular parameters:** If selected, SecureSphere treats JSessionID parameters as URL parameters. In **Delimiter**, you can specify the character delimiting JSessionID parameters. **Connection Timeout**: Determines the amount of time an idle connection retains contents of a session. Default: 300 seconds. Minimum: 1. Maximum: n/a. **Case Sensitive URLs**: Enables SecureSphere to distinguish similar URLs that only differ by letter case. This enables the profiling of two similar URLs separately. Default: disabled. **Support Microsoft Encoding in URLs**: Enables SecureSphere to identify Microsoft Encoding when used in URLs and translates encoded URLs, preventing false positives and displaying URLs in unicode in web profiles. Default: disabled. **Combine parameter names when value is empty**: Enables SecureSphere to identify empty parameters that are passed in URLs, subsequently building a cleaner profile and promoting more effective monitoring. When IIS is used, parameters of URLs do not need to be followed by an equal sign. Default: enabled. For information regarding configuring advanced settings, see Configuring Advanced Web Service Settings on page 424.
XML Parsing	Configures how SecureSphere parses XML data when creating Web application profiles, and offers the following options: **Parse XML only if SOAP**: When selected, only parses XML over SOAP. Non-SOAP XML is not parsed. Default: enabled. **Ignore Attributes**: Ignores attributes used in elements when parsing XML. Default: enabled. **Parse Empty Elements**: Parses empty XML elements, learning them as part of a profile even if the elements contain no parameters. Default: disabled. **Parse Nested XML Values:** Enables the parsing of nested XML for URLs with SOAP or XML values. Default: disabled. **Maximum Parsing Depth**: XML parameter names are fully parsed, but only the last **Maximum Parsing Depth** parts of the path are preserved. For example, if the path is `a/b/c/d/e` and **Maximum Parsing Depth** is 3, the parameter name is `c/d/e`. Default: 20. Minimum: 1. Maximum: n/a. For information regarding configuring advanced settings, see Nested XML Parsing on page 249.

Parameter Name	Description
NTLM Authentication	Customizes support for NTLM as part of Universal User Tracking, which is then used to determine if a login passed or failed, and identify the user. The following options are available: **Type**: The type of identifier taken from traffic used to determine the status of a login. **Value**: The actual value that when present, that determines the status of a login. **Login Result**: Identifies whether the login passed or failed based on the Type and Value combination. **Default Decision**: Determines how to handle traffic if it does not match any rule. Default: Success. For information on customizing NTLM Authentication, see Configuring NTLM Support on page 289.

Web Service Reverse Proxy

Defines the configuration of Reverse Proxy details.

Web Services - Application Parameters

Parameter Name	Description
Reverse Proxy	Lists details regarding Kernel Reverse Proxy mode if applicable. Kernel Reverse Proxy mode is configured during initial SecureSphere installation. **Note:** For information on Kernel Reverse Proxy operation, see the "Gateways" and "Configuring SecureSphere" chapters in the **SecureSphere Administration Guide**.
Transparent Reverse Proxy	Lists details regarding Transparent Reverse Proxy if applicable. **Enable Transparent Reverse Proxy**: Enables the use of a Transparent Reverse Proxy.
URL Rewrite	Configures the URL Rewrite Rule used to determine if URL rewriting is necessary and if so the syntax of the rewritten URL. This option is available in both Transparent Reverse Proxy mode and Kernel Reverse Proxy mode. Not available in sniffing or bridge mode. For information on configuring URL Rewrite Rules, see URL Rewrite on page 428.

Web Service Applications

Defines the mapping of URLs or host names to multiple applications on a single IP address and port.

Web Services - Application Parameters

Parameter Name	Description
Host/URL to Application Mapping	Maps host names to web applications and uses URL groups to identify URL traffic to be monitored. **Default Application**: Determines the default application assigned to all traffic not mapped to other applications. **Hosts**: Maps web hosts to their related applications and identifies URL groups to be monitored. For information on configuring host/URL to application mapping, see Configuring Host/URL to Application Mapping on page 63.

Web Service Applied Policies Information

When adding a new web service to the site tree, it automatically comes with default policies applied to it, see Using Default Security Policies on page 172. You can modify the applied policies. In the Applied Policies tab you can only view the applied policies. Adding of new policies and applying other policies to an object is done in the Security window.

When the  icon is displayed to the right of a policy, you can create a new policy or edit an existing one by clicking the icon.

Web Services - Applied Policies Information Parameters

Definitions	Default Policies Applied to Web Services
Basic Security Policies	Options include: **HTTP Protocol Validation**: Policy used to monitor and validate HTTP protocol usage. **Web Service Correlated Validation**: Policy used to monitor and analyze sophisticated attacks enabling the correlation of information related to attacks such as cross site scripting or SQL injection. **Cookie Signing Validation**: Policy used to validate signed cookies when in Reverse Proxy mode. For instructions on how to turn off policy enforcement for these components, see Tuning Security Policies in the Setup Window on page 180.
Additional Security Policies	Enables applying additional security policies on the web service level such as HTTP protocol signatures and web service custom policies. These policies must first be configured before adding. For information on configuring and adding security policies for web service components, see Using Default Security Policies on page 172.
Policies Applied to Server Group	List the policies currently configured on the parent server group. Information here is for viewing purposes only. It cannot be edited.

Managing Web Applications

Web applications in SecureSphere represent actual web applications operating in your network, and enable the monitoring of traffic for each application. By defining applications you enable SecureSphere to monitor and report on traffic on a per application basis, providing further granularity when monitoring and reporting.

SecureSphere web application options include the following:

Adding a New Web Application 70
Understanding Web Application Parameters 70

Adding a New Web Application

Prerequisite: Existing web service as described in Basic Web Service Configuration Tasks on page 59.

To create a new web application:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. In the **Sites Tree**, expand the **Server Group** and **Web Service** in which you want to create the new application. The **Service** is selected and its details appear in the **Sites** window.
3. In the **Sites Tree** title bar, click  **New**. The **Create Application** dialog box appears.
4. Type a name for the application, then click **Create**. The new application is added to the service as it appears in the Sites Tree.

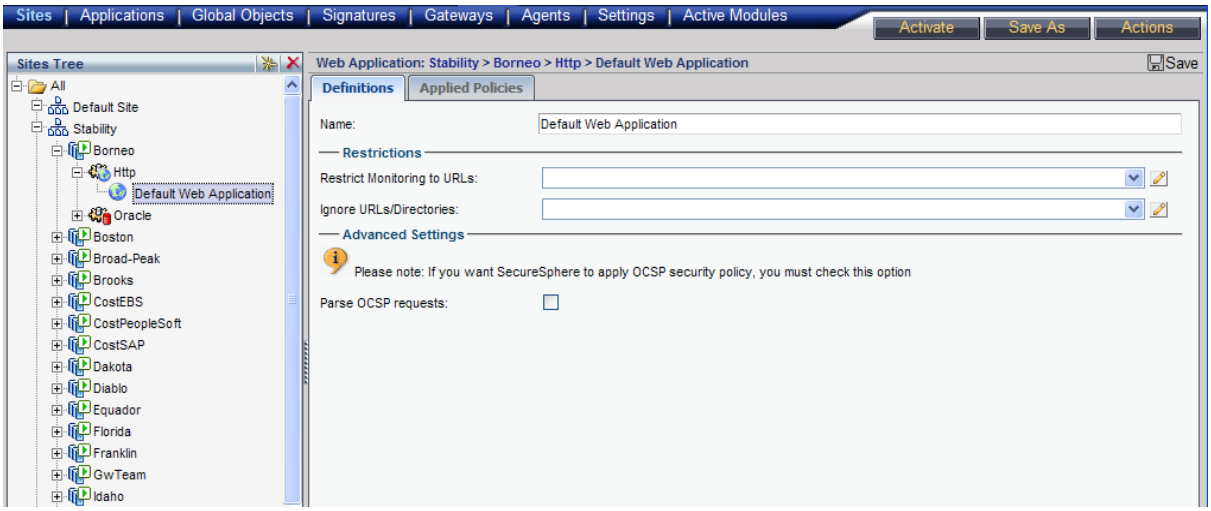


Figure 8: Sites Window Displaying Web Application Definitions

5. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

   For a description of web application parameters, see Understanding Web Application Parameters on page 70.

Understanding Web Application Parameters

Web application parameters define the web application running in your network including restrictions and security policies applied to relevant traffic. The following tables review the various web application parameters in detail. SecureSphere web application options include the following:

Web Application Definitions Parameters 71
Web Application Applied Policies Information 71

Web Application Definitions Parameters

Defines information used to identify web application traffic to be monitored.

Web Applications - Definitions Parameters

Parameter Name	Description
Name	Name of the web application.
Restrictions	Lists various restrictions that can be applied to web application monitoring including: **Restrict Monitoring to URLs**: Attaches a URL Prefixes/Directory Group global object that restricts monitoring to these URLs. Other URLs will not be learned, profiled or related to. **Ignore URLs / Directories**: Attaches a URL Prefixes/Directory Group global object for SecureSphere to ignore. For instructions on how to modify URL Restrictions, see Configuring URL Prefixes / Directory Based Restrictions on page 420.
Advanced Settings	**Parse OCSP requests**: Configures SecureSphere to parse OCSP requests for this application. This parses and profiles internal parameters and fields of the request. If you configure this option, it will not parse regular HTTP requests. For detailed instructions on configuring OCSP support, see Configuring OCSP Support on page 424. **Note:** 32-bit SecureSphere Gateways are able to parse OCSP requests up to 1K in size, and generate an "Extremely Long HTTP Request" alert when this limit is exceeded. **Profiling URLs without Parameters**: Select one of the following: • **Learn only URLs with parameters** - If a URL has no parameters, do not learn it in the profile. • **Learn all URLs, with or without parameters** - Learn all URLs in the profile, whether they have parameters or not. • **Learn all URLs, except for static URLs without parameters** - Learn URLs without parameters only if they are non-static (dynamic) URLs. The default list of static file types is defined in **Global Objects** > **File Type Groups** > **Default Static File List**. You can define your own list of static file types in addition to the default list, and you would then choose which static file list is relevant to the application in **Static Files** in the web service's **Definitions** tab. The file types in the specified static file list are not learned in the profile, but signatures and protection engines (XSS, SQLi *etc.*) are applied to them. For more information, see Configuring File Data Types. For more information, see Web Application Profiles on page 239.

Web Application Applied Policies Information

When adding a new web application to the site tree, it automatically comes with default policies applied to it, see Using Default Security Policies on page 172. You can modify the applied policies. In the Applied Policies tab you can only view the applied policies. Adding of new policies and applying other policies to an object is done in the Security window.

Web Applications - Applied Policies Information Parameters

Parameter Name	Default Policies applied to Web Applications
Basic Security Policies	Options include: **Web Profile**: Determines the policy applied to profiled web elements once the they have moved from learning to protect mode. **Worm Profile**: Determines the policy applied to used to identify and block zero day web worms. For instructions on how to turn off policy enforcement for these components, see Tuning Security Policies in the Setup Window on page 180.
Additional Security Policies	Enables applying additional security policies on the web application level. These policies must first be configured before adding. For information on configuring and adding security policies for web applications, see Adding New Security Policies on page 179.
Policies Applied to Services	Lists the policies currently configured on the parent web service. Information here is for viewing purposes only. It cannot be edited.
Policies Applied to Server Group	List the policies currently configured on the parent server group. Information here is for viewing purposes only. It cannot be edited.

General Basic Configuration Tasks

A number of general items need to be configured or modified to get SecureSphere up and running. These include the following:

Modifying Service Name and Port 72
Modifying the Character Set 73
Adding SSL Keys 73
Adding Session Identifiers Used for Session Tracking 75

Modifying Service Name and Port

When a service is defined, it is configured with a service name and port number that determine what is profiled and monitored. You can modify these defaults to match the ports used in your network.

To modify service details:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. Expand the **Server Group** that hosts the service you want to modify. Configured services are displayed.
3. Click the service whose details you want to modify. Details appear in the main window.
4. Click the **Definitions** tab. Definitions for the service are displayed.
5. Review the following items and modify if required:
   - **Name**: Type a name for the service.
   - **Ports**: Modify the existing port number or add additional ports separating each port number with a comma. Ports listed here must also appear in the Global Ports Group global object.
6. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Modifying the Character Set

When a service is installed, it is automatically configured with a default character set (Western European (windows). This must be changed if your server is using a character set other than this default. Additionally, SecureSphere has built in support for UTF-8 character sets , So for example, if your server uses both UTF-8 and Japanese, you would configure the SecureSphere character set as Japanese, because UTF-8 is provided by built-in support. SecureSphere does not support UTF-7.



Note: It is important to configure the proper character set. If your network manages data using a different character set than that configured in SecureSphere, data learned by SecureSphere is incomprehensible, putting excessive load on the system and resulting in data not being properly profiled, monitored or protected.

To modify the character set:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. Expand the server group that hosts the service you want to modify. Configured services are displayed.
3. Click the service whose details you want to modify. Details appear in the main window.
4. Click the **Definitions** tab. Definitions for the service are displayed.
5. From the **Definitions** tab, open the Character Set dropdown menu and select the character set used by your server.
6. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Adding SSL Keys

If your network includes encrypted data, you need to upload your server's Public and Private SSL keys to SecureSphere to enable the decryption of these packets and analyze their content. Once SSL keys have been uploaded to SecureSphere, it decrypts the packets, then inspects them for malicious or otherwise troublesome data. If the data complies with the configured policies, it is forwarded to its destination. If not, the packets are treated as defined by the policy, followed actions are taken and alerts are generated as configured.



Notes:

- SSL keys are stored using encryption within the management server's internal database. When used, the SSL key is maintained in a secure environment and never stored in unprotected format. For a list of SSL ciphers, see SSL Ciphers on page 74.

To add SSL Keys:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. In the **Sites Tree**, expand the relevant site and server group, then click on the service you want to configure. Service options appear in the main window.
3. In the **Sites** window, click the **Definitions** tab. Definitions tab options are displayed.
4. Under **Encryption Support**, click  **New**. The **Add SSL Keys** dialog box appears.
5. Type a meaningful **Name** for the SSL Key.
6. If you are using an HSM device such as nCipher, click the **HSM** checkbox. Note that enabling HSM requires some configuration on the gateway. See the SecureSphere Administration Guide for information on HSM vendors and instructions on how to enable and configure HSM support.
7. Select the radio button that determines the format of your SSL keys, then configure their details. Supported formats include:
   - **PEM**: Navigate to the locations of both the public and private keys.
   - **PKCS12**: Navigate to the location of the PKCS12 file, then type the password used to protect the file.
8. Click **Upload**. The relevant key files are uploaded to the SecureSphere gateway.

9. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

SSL Ciphers

SecureSphere supports a number of ciphers to enable the decryption and inspection encrypted packets in your network. The following table lists the ciphers supported by SecureSphere.

Supported SSL and TLS Ciphers

Supported in	SSL Cipher
SSLv3	SSL_NULL_WITH_NULL_NULL
	RSA
	SSL_RSA_WITH_NULL_MD5
	SSL_RSA_WITH_NULL_SHA
	SSL_RSA_WITH_RC4_128_MD5
	SSL_RSA_WITH_RC4_128_SHA
	SSL_RSA_WITH_DES_CBC_SHA
	SSL_RSA_WITH_3DES_EDE_CBC_SHA
	SSL_RSA_EXPORT_WITH_RC4_40_MD5
	SSL_RSA_EXPORT_WITH_DES40_CBC_SHA
TLSv1	**RSA**
	TLS_RSA_WITH_AES_128_CBC_SHA
	TLS_RSA_WITH_AES_256_CBC_SHA
	TLS_RSA_EXPORT1024_WITH_RC4_56_MD5
	TLS_RSA_WITH_AES_128_CBC_SHA256
	TLS_RSA_WITH_AES_256_CBC_SHA256
	TLS_RSA_WITH_RC4_128_MD5 = { 0x000x04 }
	TLS_RSA_WITH_RC4_128_SHA = { 0x000x05 }
	TLS_RSA_WITH_DES_CBC_SHA = { 0x000x09 }
	TLS_RSA_WITH_3DES_EDE_CBC_SHA = { 0x000x0A }

Supported SSL and TLS Ciphers - Diffie-Hellman

Supported in	SSL Cipher
SSLv3	**Diffie-Hellman**
	SSL_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA
	SSL_DHE_RSA_WITH_DES_CBC_SHA
	SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA

Supported in	SSL Cipher
TLSv1	**Diffie-Hellman**
	TLS_DHE_RSA_WITH_AES_128_CBC_SHA
	TLS_DHE_RSA_WITH_AES_256_CBC_SHA
	TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
	TLS_DHE_RSA_WITH_AES_256_CBC_SHA256



Note: For both SSL and TLS, the Diffie-Hellman (DHE) ciphers are:

- Supported only in Kernel Reverse Proxy and Transparent Reverse Proxy modes, not in bridge or sniffing modes.
- By default configured for 1024 bit encryption, but can be configured for 512 bit encryption. For more information on how to change the key size, refer to the Imperva Customer Portal Knowledge Base article: "How To change Diffie Hellman key size from 1024 bit to 512 bit".

Adding Session Identifiers Used for Session Tracking

Session identifiers are pieces of data used to identify a session. SecureSphere uses session identifiers as an alternative to IP addresses to identify users when monitoring and blocking traffic. They are also used to track session related violations such as read only parameter, cookie protection, web data base correlation, and more. It is important the SecureSphere have the correct sessions to ensure that operations is properly conducted such as profiling of session related elements.

SecureSphere is automatically installed with a list of standard session identifiers for most major applications. However, your network or applications may use custom identifiers. In order to get SecureSphere to properly identify all relevant traffic, you need to add these identifiers. Session identifiers are a global object. For instructions on how to configure the Session Identifiers global object, see Configuring Session Tracking on page 298.

Configuring Action Sets and Followed Actions

As part of basic SecureSphere configuration, you may want to configure certain policies to send alerts by e-mail or to a third party event management system such as Arcsight or Remedy. You can configure an action set to meet your needs, then attach it as a followed action to the required SecureSphere object. For instructions on how to configure Action Sets and Followed Actions, see Working with Action Sets and Followed Actions on page 326.

Activating Settings

The need to activate a setting once it has been configured depends on the current activation mode. SecureSphere has two activation modes:

- **Immediate Activation mode (Default)**: Immediately saves and updates the configuration on the server and sends it to the connected gateways.

- **Delayed Activation mode**: Enables you to make a number of changes to the configuration, then activate them at once. This provides better control over changes and helps to minimize mistakes. When in delayed activation mode, changes made in the following locations require activation after saving:
  Main > **Setup**
  Main > **Policies**
  Admin > **ADC**



Note: Depending on the number of changes that have been made, activation may take an extended period of time. The more items that need to be activated, the longer activation takes.

To change the activation mode:

1. Login to the SecureSphere management server machine as root after the SecureSphere Server is up and running.
2. Execute the following commands (on one line):

   vi /opt/SecureSphere/server/SecureSphere/jakarta-tomcat-secsph/webapps/SecureSphere/WEB-INF/bootstrap.properties

3. Set the `activate-settings.activation-mode` attribute to `Immediate` as follows:
   `activate-settings.activation-mode = <desired Value>`
   where `<desired Value>` is either `Immediate` or `OnActivateSettings`
   Note the spaces separating the attribute from its value.
   For example:
   ```
   # Activate settings: "Immediate" (disabled) or "OnActivateSettings" (enabled)
   activate-settings.activation-mode = OnActivateSettings
   ```
4. Run the following command to restart the server:
   `impctl server restart`

To activate settings which have been configured and saved:

1. In any SecureSphere window, click **Activate**. The **Activate Settings** window appears, displaying detailed information about all changes that have not yet been activated.
2. In the **Activate Settings** window, click **Activate**. The Activate Settings window closes and the progress bar appears, indicating the status of activation.
3. Click **OK**. All changes are activated.

CHAPTER 4

Monitoring

Monitoring is a key phase in the SecureSphere application data management lifecycle. SecureSphere is equipped with a user friendly monitor that clearly displays generated information in a central location. Real-time information that is generated includes system events, alerts, violations, blocked sources, gateway status, system warnings, and archiving information.

The Monitoring workspace contains a number of options that represent the various aspects of monitoring in SecureSphere. This section reviews the aspects involved in monitoring and includes the following topics:

Monitoring Models in SecureSphere 78

Sample SecureSphere User Types 78

Use Case: Small to Medium Business Implementation 79

Use Case: Security Officer Management 79

Use Case: ISP Implementation 79

Understanding Alerts, Events and Violations 80

Monitoring Checklist 83

Working with the Dashboard 85

Managing Alerts 88

Managing Violations 99

Managing Blocked Sources 105

Web Error Pages and Response Pages in Alerts 106

Managing System Events 107

Filtering Alerts and Violations 109

Monitoring Models in SecureSphere

The process of monitoring events, alerts and violations can take on many aspects. Depending on your specific implementation, there may be a number of different types of users with different roles in monitoring alerts. You can determine actual severity of specific alerts as opposed to rated severity determined by SecureSphere, determining if an alert is a false positive or an attack, and much more. This section reviews various types of SecureSphere users, and also a couple of use cases based on different SecureSphere implementations.

The following topics provide background information and assist in understanding the SecureSphere monitoring model:

Sample SecureSphere User Types

For the purposes of this document, user types are various users that may have different roles in an organization and responsibilities in a network. They are either directly responsible for SecureSphere management, or a participant in the monitoring workflow. User types here are given for illustration purposes only. The following table describes the various types of people that may use SecureSphere.

SecureSphere Users

User Type	Description
Security Officer	Responsible for the security of network and data components in a company including the firewall, routers, SecureSphere operation, and more. Since security officers are responsible for multiple components, they want SecureSphere to be efficient, and not receive alerts for every minor issue.
SecureSphere Administrator	Considered a high-level administrator. May be a network administrator, or an individual dedicated to monitoring and maintaining SecureSphere operation. The administrator checks ongoing status of SecureSphere, reviews alerts and determines actual severity. May manage risk by consulting with other users with more expertise regarding various aspects of traffic. They may consult with relevant users, or they may make all decisions independently. Typically has complete access and editing rights.
Web Master	Web masters may play a role in the analysis of alerts and violations as they pertain to the company website and web service infrastructure. They may be asked to help determine the risk posed by specific alerts and violations based on security events that involved the web server, its files and applications. They may also be provided advanced privileges which would enable them to modify specific policies which may be related to web server activity.
ISP Administrator	Considered a high-level administrator, responsible for SecureSphere operation and management when installed in an ISP. In this scenario, the ISP provides an added-value service to its customers. Each customer may have their own access to the specific SecureSphere server group, service or application installed in their network. They may have complete access rights to their products, but no access rights to other products. The ISP administrator is responsible for overall management functions, for the various ISP customers and providing them individually with the access rights for their specific applications.

Use Case: Small to Medium Business Implementation

In this scenario, our client is a small to medium business. There is an administrator who in addition to administering the network must also administer SecureSphere. SecureSphere has been installed and has undergone fine-tuning. The administrator comes in first thing in the morning and takes a look at the SecureSphere dashboard to determine system status, and review the alerts for serious threats.

The SecureSphere administrator consults with relevant staff regarding relevant violations. A workflow has been created by which the SecureSphere administrator marks all alerts for follow-up as important. Custom filters have been created by <administrative personnel>. These users filter alerts by the important flag and by the services that fall under their jurisdiction.

Twice a day, at 10:00 am and at 16:00 the relevant staff member conducts a search using this filter. If any relevant alerts are found, they are examined and risk is determined.

The webmaster only has reading rights. The webmaster conducts a search while filtering for web service related alerts that are marked as important. The webmaster then review the alerts. If the alert is benign, they flag it as dismissed. If they believe further action needs to be taken, they consult with the SecureSphere administrator.

Use Case: Security Officer Management

Security Officers are responsible for managing the various security components in their company including the company firewall, access to routers and SecureSphere operation. Each one of these systems can produce multiple violations. They arrive first thing in the morning and check their syslog manager. They have configured all security systems installed in their network to generate syslog messages for mission critical issues that put the system at risk. They do not want to be bombarded by messages announcing minor security violations that have taken place in the network. If they find any messages from SecureSphere, they open then up, review the problem and try to resolve them immediately.

Once important alerts for the different systems in their network have been handled, they sit down and review all SecureSphere high severity alerts and handle them accordingly based on their knowledge of the system. They may also consult with other individuals to understand the source, threat and impact of violations and changes they may make to SecureSphere, but have no set workflow or additional participants in the monitoring workflow. While the security officer is responsible for security aspects of SecureSphere operation, the company's Network Administrator is responsible for the management of users and permissions within SecureSphere.

Use Case: ISP Implementation

ISP implementation adds an additional layer to the monitoring model of SecureSphere. In this model the ISP has installed SecureSphere to monitor the web servers of various companies they host. Each company is completely invisible to the other companies, and each company has its own login to SecureSphere. All companies are viewable to the ISP manager.

In ISP implementation, there are multiple administrators.

The high-level administrator based in the ISP is responsible for configuring new server groups based on customers (companies) who have paid for an added value service in the form of SecureSphere to help monitor their web server. The ISP SecureSphere administrator is responsible for the adding and modifying components and users, and ongoing maintenance of each company's SecureSphere definitions. The focus is also on the status of the SecureSphere gateways, maintaining their operability and uptime.

The lower-level administrators are based in the individual companies. They monitor alerts based on security events, evaluating threats and modifying the SecureSphere profile for their particular server group, services and applications. They may also consult with other users in their network.

Understanding Alerts, Events and Violations

The SecureSphere monitoring lifecycle involves a number of phases and consists of a number of building blocks to keep you up-to-date on the events taking place in your system. Understanding these building blocks and being able to distinguish between them is important to understand how SecureSphere monitors and protects traffic.

Before reviewing the building blocks, it is important to understand that the mechanisms that bring these items to life are policies. Policies are key elements in SecureSphere monitoring. For further information on policies, see Introducing Security Policies on page 166.

The following table reviews building blocks used in SecureSphere monitoring.

Monitoring Building Blocks

Parameter	Description
Regular Event	A generic term for any change or action that has been detected by SecureSphere.
Network Event	An event in network traffic that has been detected by the SecureSphere gateway. There are two types of network events: **Learning Event**: The discovery of a new item to be protected in the network. **Security Event**: A violation by network traffic of a security policy.
Learning Event	The discovery of a new item to be protected in the network. A learning event involves the following phases as illustrated in Monitoring Architecture on page 83. • A network event is detected by SecureSphere Gateway on incoming traffic or by an agent installed on a server in the network. The gateway determines it is a learning event. • The learning event is sent to the SecureSphere Management server. • The new item to be learned is added to the relevant profile. • The first time this item is detected, SecureSphere management server generates a system event. **Example**: SecureSphere has discovered a new URL in the network. The URL is added to the web profile, a system event is generated and sent to SecureSphere management server with the system event appearing in the System Events window.
Security Event	The discovery of an event that violates a security policy on monitored traffic. Security events themselves are not displayed in SecureSphere, only their resulting alerts and violations. A security event involves the following phases as illustrated in Monitoring Architecture on page 83 Figure 1. • A network event is detected by SecureSphere Gateway on incoming traffic or by an agent installed on a server in the network. The gateway determines it is a security event. • The security event is sent to the SecureSphere Management server. • A **violation** is generated for each policy that has been violated. • Violations are analyzed, correlated, then assigned to **alerts**. SecureSphere correlates and **aggregates** violations based on known attacks. This helps identify attacks as they take place. **Example**: SecureSphere detects that a user has violated a policy that determines who can access specific data. A violation is registered and a security alert is created and displayed in the Alerts window.
Violation	An indication that a security event has taken place which violates a policy. A single security event can trigger multiple violations. Violations are displayed in the Violations window, a summary of violations can also be seen in the Details pane of the Alerts window. A violation is generated for each policy that has been violated, unless Server group is in active mode and blocking policy has already been violated by the event, at which point the transaction is blocked and no further action is taken. **Example**: SecureSphere detects the signature of an SQL injection attack (a security event) which violates a policy. A violation is created and an alert is generated and displayed in the Alerts window. Additionally, the violation appears in the Violations window.

Parameter	Description
Alert	Notifications that an event or group of events (security events) have taken place that violate a policy. While a violation is an indication of a single security event, an alert is a notification that one or more violations have taken place. A single alert may list up to (aggregate) 30 violations. **Example**: SecureSphere determines that the checksum of a cookie signature is incorrect. This violates a policy that generates an alert which is displayed in the Alerts window, with the violation details appearing in the Violations window.
Aggregation	The process of grouping related violations into a single alert to assist in identifying attacks. If an aggregation already exists related to the generated violation, the new violation is added to it. If no related aggregation exists, a new aggregation is created and future related violations are added to this aggregation. Violations are displayed in the Violations window. **Example**: SecureSphere detects that similar signatures have been detected in traffic from a number of different sources. Violations are aggregated and an attack is identified as being from e-mail collector robots.
System Event	An event related to the system, including user login, system errors and warnings, updates, items added to a profile for learning, and changes to configuration. System events may have compliance implications. A System Event in SecureSphere is not considered an alert. Instead, system event notifications are displayed in the System Events window. For an illustration of how system events are generated, see Monitoring Architecture on page 83. **Example**: SecureSphere adds a item to learning it has detected, or an Administrator logged into the system. An event is created and displayed in the System Events window.

Monitoring Architecture

The figure below illustrates the monitoring architecture of SecureSphere. The left-hand side of the diagram illustrates SecureSphere architecture, while the right-hand side illustrates alert architecture, and what is displayed in the Dashboard.

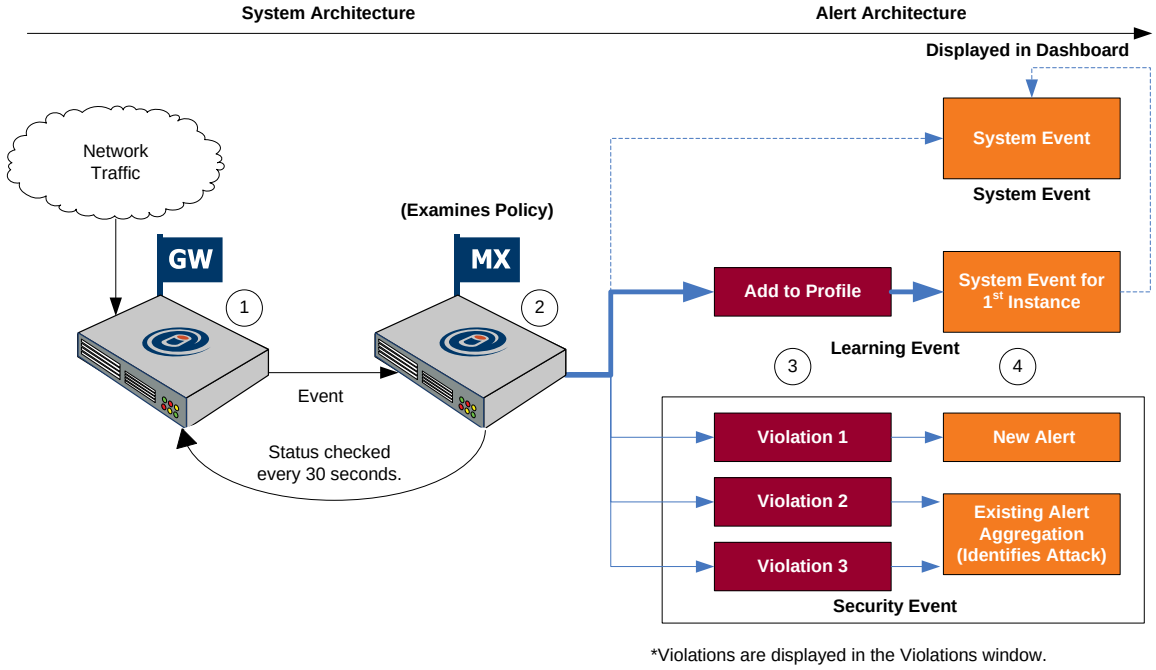


Figure 9: Monitoring Architecture

Monitoring Checklist

Monitoring the ongoing status of your system includes a number of standard tasks. The following list provides a summary of these tasks. A more detailed description of these tasks is available later in this chapter, as indicated. The primary workflow for SecureSphere management is as follows: **Check System Events > Review Alerts > Analyze violations and conduct weekly maintenance**.

Monitoring Checklist

Parameter	Description
System Events	System events help gauge the status of the SecureSphere gateways. Basic workflow for reviewing system events include: • Check gateway statistics, verify the gateway is online and its operation is normal. • Review system events. Look for reoccurring problems, for example, alerts regarding high CPU utilization. An occasional spike is normal, but if a problem like this reoccurs, it can indicate issues with hardware or configuration. • Check that there are not a large number of failed logins. • Check there are not a lot of alerts with high severity. • If you have scheduled archiving, verify that it has been conducted.
Alerts	Alerts can be viewed in the Dashboard, though the Dashboard only displays the last few to be generated. It is therefore recommended that once you have reviewed recent alerts, you then view the alerts in the Alerts window. Basic workflow for reviewing alerts includes looking for the following: • High severity alerts. • Alerts containing a large number of aggregated violations, this may indicate either an attack taking place, or an inefficient profile which needs to be fine-tuned. • Number of times the a specific violation took place. A large number of the same violations may indicate a false positive. • Multiple signatures from a single IP. This may indicate someone trying to hack the system. • Distributed over time - whether all the alerts were generated approximately at the same time, or evenly distributed over time as displayed in bar chart at the top of the Alert Details pane. This can help indicate if an attack is taking place or alternatively is a false positive. • When viewing alerts generated by users trying to access a number of items; whether a user should be permitted to access these items. If so, you might add the violation to the profile. • If you need assistance determining if specific users should have access to an item, consult with others in your organization. You may want to consult with web master regarding web based issues.
Violations	Many violation details can be viewed from an aggregated alert by expanding an individual violation. You can additionally see violation details from the Violations window. Information can be easily scanned in the details pane of the violations window. Do the following: • View additional violations displayed in the additional Violations pane. This displays other violations that are related to the selected violation. • View the response page. This displays the source code response if the response triggered the violation. • View related information as described in Violation Analysis Tools on page 103.

Parameter	Description
Weekly Status Checkup	• In addition to regular tasks, it is recommended that you conduct a number of actions to help you get a high-level overview of items which may be related, as follows: • **View Overall Status Over Time**: To do this, in the **Main** workspace, click **Profile**, then drill down to a specific application. • For example, select your company's web server. From the **Display** button in the upper right-hand side of the window, select **Overall Status Graphs**. This displays valuable information providing insight into the status of the application you're viewing. For example, when looking at a web application, you should see a general downtrend in the **Number of Added URLs Over Time** graph. • **Generate weekly reports**: Including **Top 10** reports, **Configuration** reports to see how many alerts were added, and a System event report. For example, displaying the number of failed login attempts to SecureSphere. This may help you determine for example, if someone is trying to hack into the SecureSphere gateway. For information on how to generate reports, see Reporting on page 28.

Working with the Dashboard

The Dashboard is the central monitor for all data generated in SecureSphere. It summarizes the various types of monitored data in a central location, and provides your first indication regarding attacks taking place, system events and other issues your system might be experiencing. Data is displayed in near real-time (every few seconds).

To view the Dashboard:

- From the **Main** workspace, select **Monitor > Dashboard**. The dashboard appears.

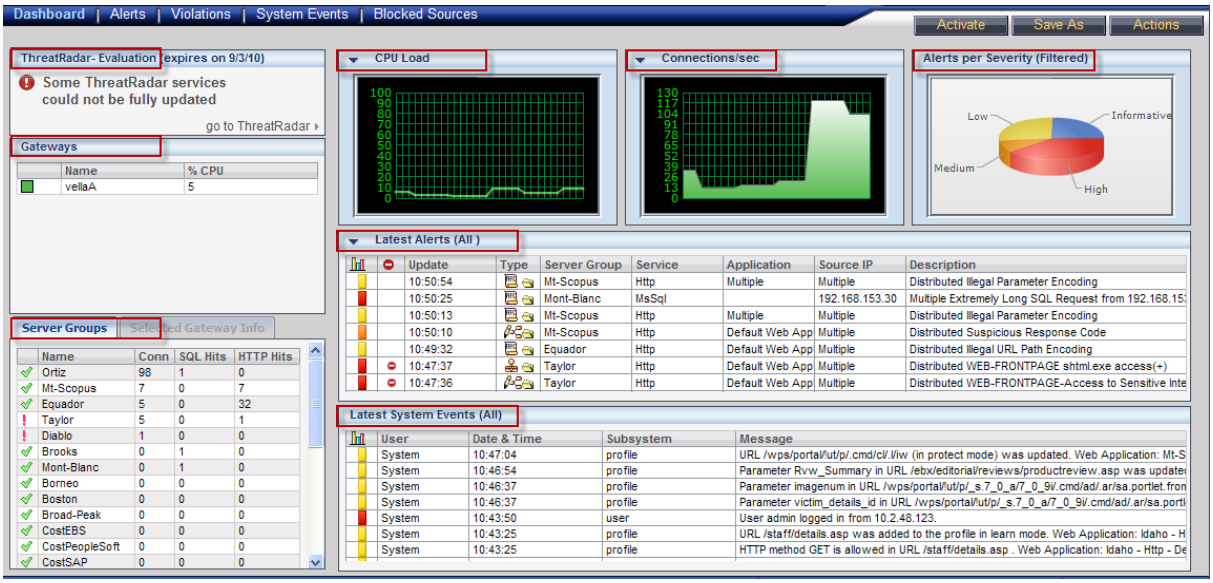


Figure 10: SecureSphere Dashboard

- Server group statistics are updated every 30 seconds based on the last update.
- The two left-hand graphs (CPU Load / Hits per Second Graph and Throughput / Connections per Second Graph) are updated every 5 seconds.

  Details displayed in the Dashboard include the items listed in The following table.

Dashboard Components

Parameter	Description
ThreatRadar	Shows the status of ThreatRadar Reputation Services, a feature of SecureSphere that actively protects you from large scale automated attacks from known attack sources. ThreatRadar is only active when licensed. If you are not licensed and find your network under attack, you can enable an emergency license without prior coordination with Imperva. For more information in ThreatRadar, see ThreatRadar Fraud Prevention Services on page 206.
Gateways Pane	Lists the gateways currently being managed by SecureSphere.
Server Group Pane	Lists the server groups configured for the selected gateway, provides statistics regarding traffic on the server groups, and provides a visual cue as to the current operating status by using a green checkmark (running), or red exclamation mark (warning). When a warning is displayed, you can hover over the warning to display a tooltip with details regarding what is causing the warning. When no gateway is selected, server groups for all gateways are displayed.
CPU Load / Hits per Second Graph	Displays a graph for all gateways showing either the current CPU load or hits per second. You can manually change between the two graphs, or configure the display to automatically alternate.
Throughput / Connections per Second Graph	Displays a graph for all gateways showing either the current gateway throughput or connections per second. You can manually change between the two graphs, or configure the display to automatically alternate.
Alert Severity Graph	Displays a breakdown of all alerts in the system by their severity. If a filter has been applied in the latest alerts pane, represents only those alerts that match the filter.
Latest Alerts Pane	Displays a list of alerts that have been generated by SecureSphere. To open an alert's details, click on the alert. Retrieves data every 10 seconds while graphs are refreshed every 3 seconds. If a filter has been applied, displays only those alerts that match the filter, with the type of filter appearing in the title bar. List displays amount of alerts that can fit in window.
Latest System Events Pane	Displays a list of system events generated by SecureSphere. Retrieves data every 10 seconds.

Alert and Event Notification Capacity

SecureSphere maintains two tables for notifications. Once the first table is filled up, SecureSphere begins to use the second table. When this table is filled, SecureSphere clears the alerts and events in the first table, then begins writing the new alerts and events. The approximate amount of time it takes to completely fill up both SecureSphere notification tables depends on the amount of traffic being protected. It can range anywhere from 2 to 4 years, based on the size of your implementation.

Dashboard Management Workflow

The Dashboard provides you with your primary interface with SecureSphere alerts and violations. It enables you to get an overall view of the components being monitored in your network and their status. There are a number of primary indicators that show problems with the system, whether it be problems with the configuration of SecureSphere or violations taking place in monitored traffic.

1. Configure graphs to alternate every minute. For instructions on how to configure graphs, see Configuring Gateway Graphs on page 88.
2. Check the **Gateway Panel** and verify that its status is green, meaning it is online.
3. Click on the Gateway whose status you want to view. All other data displayed in the Dashboard update to reflect the selected gateway.
4. Review the **Server Groups** pane to verify that all servers are running. Search for warnings (indicated with an exclamation point). If a server group has a warning, hover over the red exclamation point to obtain a tooltip with details regarding the warning.
5. Review **CPU Graphs** and verify that performance falls within reasonable levels. Typical performance should not be above 60 or 70% for the various values. If graphs indicate levels above 60-70%, it may indicate either a problem with the physical gateway or some form of attack taking place. You can also view the **Latest System Events** pane to search for any events that may be impacting performance.
6. Review the **Latest Alerts** looking for severe alerts. If a severe alert is present, open it by clicking it once. Analyze the violations in the alert to understand why it is taking place, and subsequently whether any action needs to be taken.

Filtering Alerts in the Dashboard

This procedure describes how to filter alerts that are displayed in the Dashboard **Latest Alerts** pane.

To filter alerts:

1. In the **Main** workspace select **Monitor > Dashboard**. The **Dashboard** appears.
2. In the **Latest Alerts** pane title bar, click ▼. Filtering options are displayed. Filtering options available can be seen in the table **Alert Filtering Options** below.
3. Select the desired filtering option.

   Example: From the title bar of the Latest Alerts pane, click ▼, then select **Severity** > **High**. The graph displays all alerts with a high severity.

Alert Filtering Options

Option	Description
Severity	Filters alerts by information, low, medium or high.
Immediate Action	Filters alerts by the immediate action taken after the violation was detected.
Alert Type	Filters alerts by the type of alert, including Firewall, Signature, Protocol, Profile, Correlation, Profile, Worm, Correlation, Custom, Anti Automation or Block.
Alert Flag	Filters alerts by flags that have been manually applied to the alert. Available flags include: Important, Acknowledged, Dismissed and None.
Server Group	Filters alerts based on the sever group to which they belong.
Custom	Filters alerts by custom filters that have been created.

Monitoring Gateway Status

SecureSphere gateways are monitored to detect problems before they can impact system performance. Additionally, abnormal gatewayt behavior can often be the first sign of an attack. Whether it be due to hits per second caused by a DoS attack, or unusual load. SecureSphere offers a number of easy to read graphs that are updated in near real-time that display CPU load, hits per second, connections per second and throughput, in addition to other per gateway statistics such as connections,and HTTP hits. Status regarding the selected gateway is updated and refreshed every 30 seconds.

The various tasks that can be conducted to monitor gateway and agent status include the following:

Viewing a Specific Gateway Details 88
Configuring Gateway Graphs 88

Viewing a Specific Gateway Details

The Dashboard displays information regarding all SecureSphere gateways.

To view details of a specific gateway:

1. In the **Main** workspace, select **Monitor > Dashboard**. The **Dashboard** is displayed.
2. In the **Gateway** pane, click the gateway whose details you want to display. Graphs in the dashboard update to display details regarding the selected gateway.

Configuring Gateway Graphs

Each gateway graph can display two different types of information and can also be configured to automatically alternate between these options at a regular interval.

To configure graphs:

1. In the **Main** workspace, select **Monitor > Dashboard**. The **Dashboard** is displayed.
2. From the title bar of the graph you want to configure, click ▼. Available configuration options for the selected graph are displayed.
3. Click the desired option. The selected setting is immediately applied.

   Example: From the title bar of the CPU graph, click ▼, then select Alternate > 1 min. The graph alternates every minute between CPU usage and Hits per Second.

Managing Alerts

Alerts are notifications that a violation or group of violations (of security policies) have taken place on monitored traffic. A single security attack might contain a long sequence of violations, each generating an individual alert and resulting in an alert storm. To correlate alerts into a of logical group, prevent alert storms and assist in identifying attacks, SecureSphere aggregates violations based on attack type, then displays aggregated violations as an alert in the Alerts window.



Notes:

- By default SecureSphere displays alerts in the Alerts table for the past three days. To display alerts from more than the past three days, configure and apply a filter. For example, configure the Last Few Days filter for 7 days. For information on configuring filters, see Filtering Alerts and Violations on page 109.
- If the Management Server is restarted, the numbers assigned to new alerts jumps to the next increment of 1,000. For example, if the last alert number was 3,112, when SecureSphere is restarted the next alert will be assigned 4,000.

The following subjects discuss managing alerts:

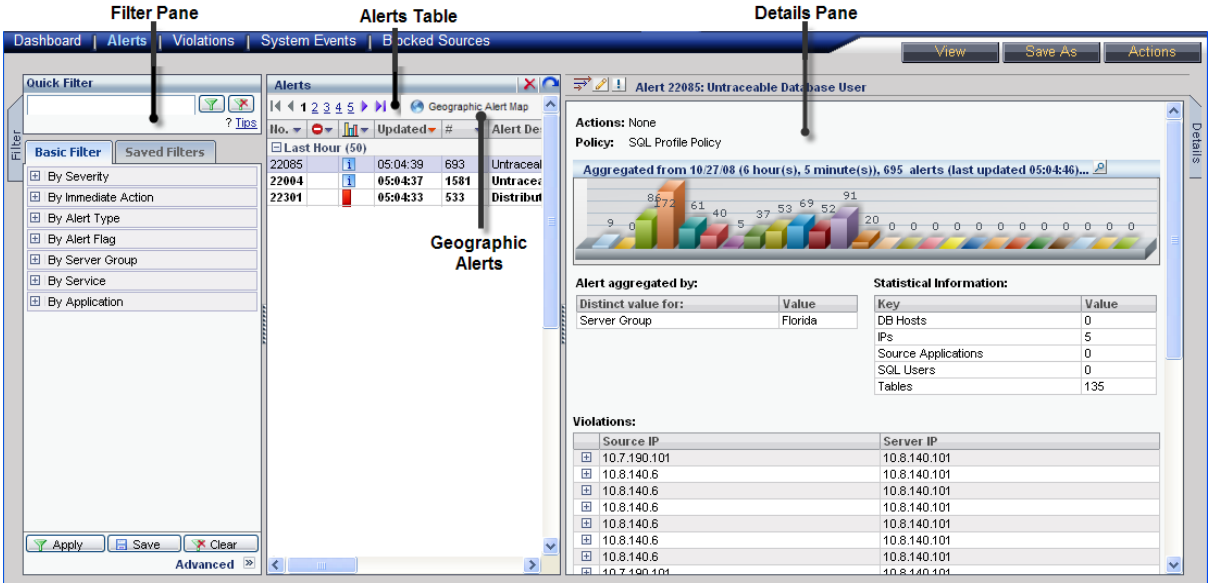
Viewing Alerts in the Alerts Window 89
Geographic Alert Map 89
Alert Management Workflow 90
Understanding the Alerts Table 91
Analyzing Alerts in the Details Pane 93
Flagging Alerts 97

Creating a Multiparty Review Workflow 98
Taking Action on Alerts and Violations 98

Viewing Alerts in the Alerts Window

To view alerts:

- In the **Main** workspace, select **Monitor > Alerts**. The **Alerts** window appears.



The Alerts window has three primary sections:

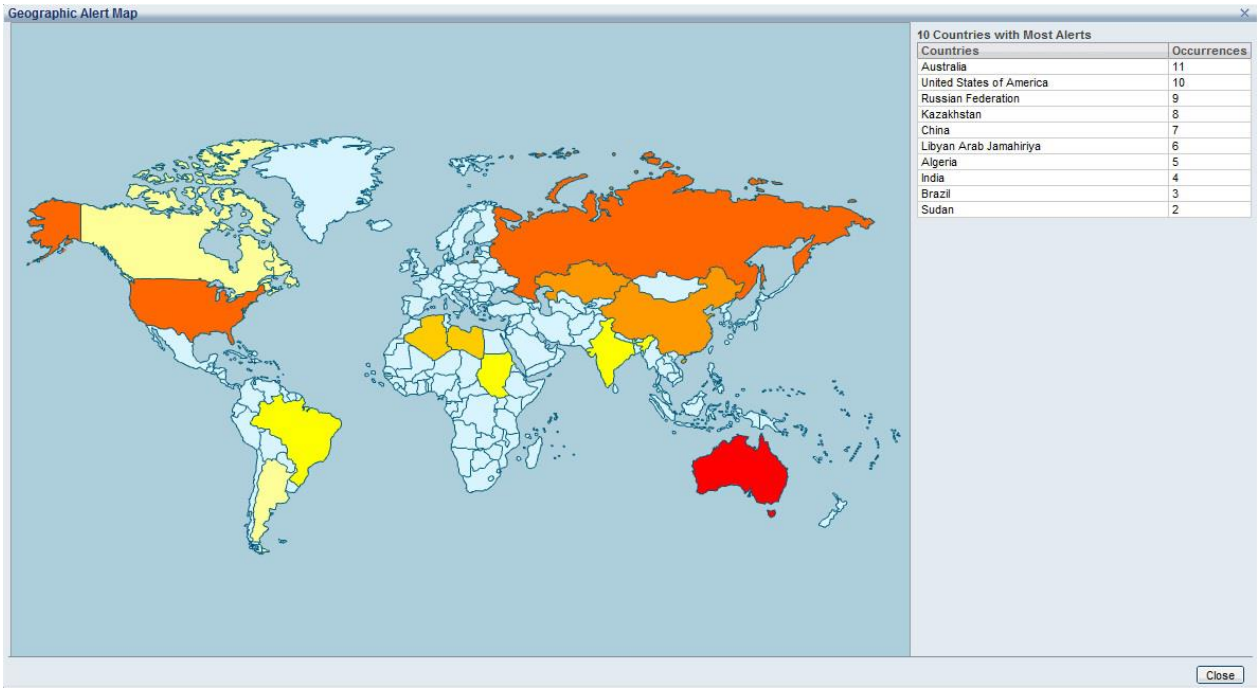
- **Filter Pane**: Enables you to filter items using one of three filter types: Quick Filter, Basic Filter and Advanced Filter. For instructions on how to filter alerts, see Filtering Alerts and Violations on page 109.
- **Alerts Table**: Lists alerts and their details summarizing information in a single row. For information regarding alerts in the Alerts table, see Understanding the Alerts Table on page 91.
- **Geographic Alert Map**: Opens a color coded world map which displays countries by number of alerts. For more information see Geographic Alert Map on page 89.
- **Details Pane**: Provides detailed information regarding the selected alert. For information on how to analyze alerts using the Details pane, see Analyzing Alerts in the Details Pane on page 93.

Geographic Alert Map

Users with a license for the Geo Location feature can view alerts by geographic distribution together with color coding which indicates what countries are most active. The 10 most active countries are shown colored ranging from yellow to red based on their relative number of alerts (red meaning the most alerts), all other countries that are not the ten most active but still are generating alerts are shown in a light yellow.



Note: The Geographic Alert map is only available with a valid Geo-IP license. If you have an expired license, the map is displayed but data are not updated.



Legend

Countries are displayed by number of alerts, color coded as follows:

- Most Active Country
- 2nd and 3rd Most Active
- 4th and 5th Most Active
- 6th and 7th Most Active
- 8th, 9th and 10th Most Active
- All other countries with alerts
- Countries with no alerts

Alert Management Workflow

Alerts are a key aspect of SecureSphere Data Lifecycle Management. They contain aggregated information that assists you in analyzing events taking place in your network and alert you to suspicious or problematic activity.

Note: When you select an alert, it is considered Opened and marked as read, and it is no longer bolded in the Alerts table.

The following is an overview of the workflow for working with alerts:

1. Scan for alerts with high severity in the Alerts window.
2. View alert descriptions and other details and try to understand what sort of a threat it poses.
3. Try to determine if the alert is a false-positive. For information on false positives, see Identifying False Positives on page 95.
4. If the alert seems important, click the alert to display the violations that generated the alert in the details pane, and analyze its contents.

5. If you believe an alert deserves further examination, you can view violation details by right-clicking and open the corresponding violations.
6. Determine if the violations generating the alert pose a threat. Take action such as adding an exception, modifying the policy that generated the alert to match the desired treatment of similar events in the future, or adding the behavior to the SecureSphere profile as described in Taking Action on Alerts and Violations on page 98.
7. Additionally, you can flag the alert to assist in management. You can flag alerts as Important, Acknowledged or Dismissed. For information on flagging alerts, see Flagging Alerts on page 97.
8. Finally, you can delete alerts to remove them from the Alerts window. Alerts are permanently deleted, though their source violations are always preserved in the system.
9. For a checklist of other items to review, see Monitoring Checklist on page 83.

Understanding the Alerts Table

The Alerts table provides a summary of information regarding a single or aggregated alert generated by SecureSphere. The Alerts table is customizable and can display a range of basic information. The Figure below displays the Alerts table.

Note: Additional columns may be available. For information on how to configure displayed columns, see Adding / Removing Columns from the Alerts and Violations Tables on page 92.

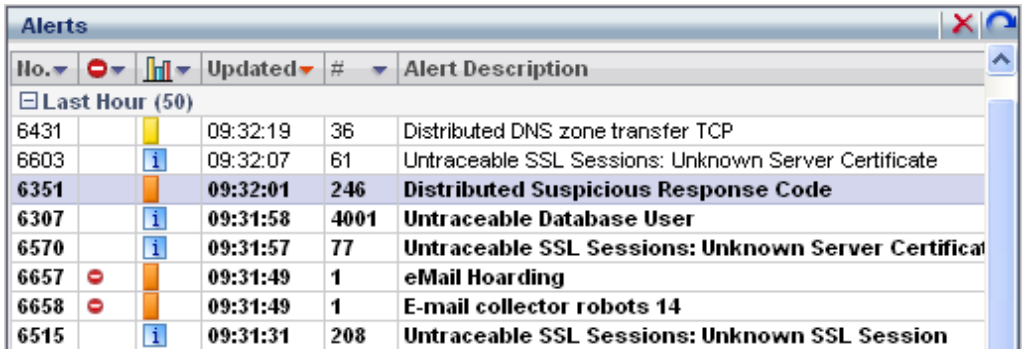


Figure 11: Alerts Table

The following table reviews columns available in the Alerts table.

Note: To sort columns in a table, click on the desired column heading. The arrow direction indicates whether the sort order is ascending or descending based on the selected column.

Alerts Table Contents

Column Heading	Description
No.	**Alert Number:** Alerts are numbered in sequential order.
	Blocked: The violation was blocked by an immediate action, which was defined in the policy related to this violation.
	Severity: The level of **severity** configured for the violation. There are four levels of severity: High Medium Low Informative **Note:** Informative is used for very low severity or when an event has taken place which is not a security violation. It may provide additional information.
Type	The type of attack that has occurred. Attack types include: Anti Automation Correlation Custom Correlation Firewall Profile Profile Protocol Signature Worm
Updated	The last time the alert occurred.
#	Number of violations aggregated in the alert.
Alert Description	A short description of the alert. It may contain the name of the policy, rule, signature that triggered the violation, or information regarding aggregation.
Server Group	The server group to which the alert relates.
Service	The service which is involved in the alert.
Source IP	The IP which caused the alert to be generated.

Adding / Removing Columns from the Alerts and Violations Tables

The Alerts and Violations table are customizable and enable you to add or remove predefined columns. This enables you to customize the information displayed.

To add or remove columns:

1. In the **Main** workspace select **Monitor > Alerts**. The **Alerts** window appears.
2. From the right-hand side of the **Alerts** window, click **View > Select Columns**. The **Choose Columns** dialog box appears.
3. In the **Choose Columns** dialog box, click once on the item you want to add or remove. The item is selected.
4. Click the **Add** or **Remove** arrows. The item is moved to the desired column.
5. Click **OK**. The item(s) are added or removed from the table.

Example: In the Alerts window, click **View > Select Columns**. From the choose columns dialog box, add Source IP, then click **OK**. The source IP Column appears in the Alerts table.

Analyzing Alerts in the Details Pane

The Details Pane displays comprehensive information regarding alerts in both graphic and textual format. Data displayed in the Details pane is context sensitive, and changes based on the type of alert being viewed. The following templates are used to display alerts:

Individual vs. Aggregated Alerts

Option	Description
Individual Alerts	Used to display an alert with a single violation. Displays in-depth information regarding this violation. When an alert with an individual violation is generated, it may signify that an attack is taking place, or someone may be probing your network looking for vulnerabilities. For information on understanding violations, see Analyzing Violations in the Details Pane on page 102.
Aggregated Alert	Used to display an alert containing multiple violations. Displays overall information for the aggregated alerts. When an alert is generated that contains multiple violations, it may indicate either that an attack is taking place, or alternatively that the profile related to the alert may need fine tuning. For further information on aggregated alerts, see Understanding Aggregated Alerts on page 94.

The following figure shows an example of the Aggregated alert Details pane.

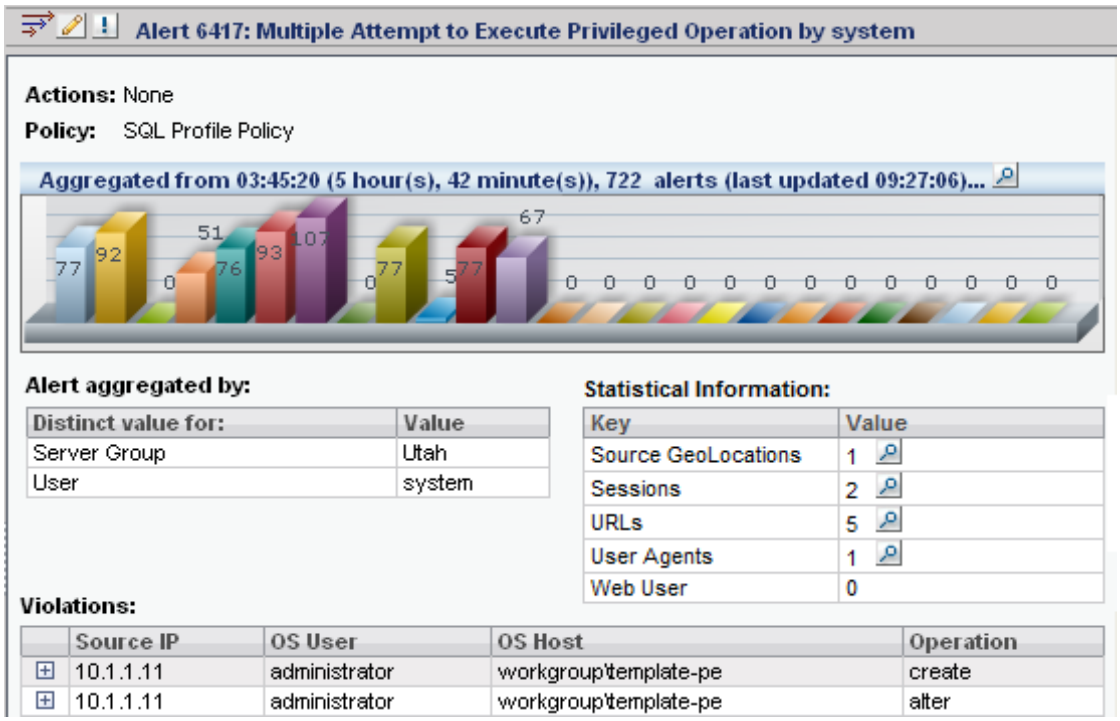


Figure 12: Aggregated Alert Details

The following table reviews the contents of the **Aggregated Alerts** pane.

Aggregated Alert Information

Parameter	Description
Actions	The immediate and followed actions that were defined for the violated policy.
Policy	The name of the violated policy.
Alerts distribution graph	A graphic representation of the aggregated alert distribution over time. The graph displays data over a 12 hour period, in 1/2 hour intervals. The time frame in which the distribution takes place is displayed in the graphs title bar. SecureSphere displays the time at which aggregation started and the time at which it ended. If the aggregation has not ended, the time of the last alert in this sequence is displayed. To read a description of the aggregation rule, click .
Description	A description of the aggregation rule. **Note:** When the word "distributed" appears in an alert description, it refers to an alert that aggregates violations which originated from multiple IP addresses.
Alert aggregated by	The alert aggregation method. For further information, see Analyzing Aggregated Alerts on page 95.
Source GeoLocation	Displays the geographic origin of IP addresses that are part of an aggregated alert. This field displays the source Geolocation if you have an active license for ThreatRadar Reputation Services. Otherwise it remains empty.
Statistical Information	Lists a number of details regarding the alert including OS hosts, IP addresses and applications that were the source of the alert, SQL users and tables that were identified in traffic of the alert, and more.
Violations	A list of aggregated violations that make up the alert. Individual violation details can be viewed by either expanding an individual violation, or opening the **Violations** window. A default maximum of 30 violations are displayed in an alert. This default is configurable.

Understanding Aggregated Alerts

Aggregation is the process in which violations are correlated and grouped in a logical manner that reflect various types of attacks. A single security attack might contain a long sequence of violations, each generating an individual alert for each violation and resulting in an alert storm. Aggregating multiple violations into a single alert prevents alert storms.



Note: Up to 500 security events can be aggregated into a single alert.

The real power in aggregation is the ability to get a high-level overview regarding violations that may be related to each other, their occurrence over the last 24 hours, distribution across various users, and the ability to drill down and see other details regarding violations that are part of the aggregated alert. Examining this information gives you insight into the actual threat an alert may pose, or alternatively may enable you to identify it as a false positive, indicating that you should fine tune SecureSphere. For information on identifying false positives, see Identifying False Positives on page 95. For information regarding analyzing aggregated alerts, see Analyzing Aggregated Alerts on page 95.

Analyzing Aggregated Alerts

Each row in the Alerts table is either a single alert or aggregated alert. An aggregated alert can be identified by seeing a number larger than one in the # row, indicating the number of violations aggregated in that alert, while its details pane contains more information than a single alert. An aggregated alert displays up to the first 30 violations that are related to the specific aggregation (this item is configurable by the SecureSphere administrator). The administrator can configure SecureSphere to further generate an alert for every violation under System Definitions > Alert Aggregation Configuration.

Some examples of how alerts may be aggregated to indicate specific problems include alerts aggregated by:

- URL, Server Group, Method, Parameter and host to understand Read Only parameter based alerts.

For a comprehensive list of Aggregated Alert keys, see Alert Aggregation Descriptions on page 646.

EXAMPLE: HOW AGGREGATION ASSISTS IN ANALYZING VIOLATIONS

Let's say that SecureSphere has detected the protocol violation Abnormally Long Request. The Abnormally Long Request violation is the result of a URL being longer than the permitted length.

- If you receive a single protocol violation (for example, Abnormally Long Request), it may indicate that a hacker is probing your network to gauge the limits of components in your network.
- If you receive multiple protocol violations from the same IP (in an aggregated alert), then this may suggest a scanner being used on the offending IP, perhaps for valid reasons (or alternatively, trying to size up your network).
- If you receive multiple protocol violations from different IP addresses (in an aggregated alert), it may either represent that the traffic should actually be permitted and your system needs to be fine-tuned (adding an exception, see Security Policy Exceptions on page 173), or alternatively that some type of malicious code is circulating in your network.

The only way to distinguish between acceptable and malicious activity is being familiar with your network and users, understanding the details of the violation, and consulting with relevant users.

Identifying False Positives

A false positive is an alert which has been generated on traffic that should be defined as acceptable behavior. Identifying false positives is an integral, on-going task of fine-tuning the operation of SecureSphere in your network. For example, you may see many violations in an alert that are distributed over time and all these alerts originate from the same source IP. This may indicate that these are the result of actions being taken by a test device such as a web security scanner being used by an authorized individual in the network. A lack of proper system tuning can result in alert storms which can be confusing and create unnecessary overhead. If you experience any of the below signs, they may be indications that your system needs to undergo fine-tuning.

Signs of false positives:

- Multiple violations distributed over time from multiple IP addresses
- Violations of different types distributed over time all from the same source IP
- Many different users all doing the same thing
- Specific, consistent violations (typically protocol violations), with source IP being some form of middle ware / CDN.

  For information about aggregated alerts and how they can assist you in identifying false positives, see Analyzing Aggregated Alerts on page 95.



Note: It is important to note that while all the above may be signs of false positives, they may also be signs of an attack.

Editing Policies from the Details Pane

SecureSphere enables you to directly access the policy of a generated alert and modify that policy from the Details pane. This can be useful, for example, when you are receiving alerts that do not represent a risk.

To edit policies from the Alerts window:

1. In the Alerts table, click the alert you want to manage. Alert details are displayed in the **Details** pane.
2. From the top of the **Details Pane**, click **Edit Policy** . The policy that generated the alert appears in a separate window.
3. Review or edit as required.

Opening Violations from the Alerts Table

Alerts displayed in the Alerts table are an aggregation of the violations that caused the alerts. If you would like more detailed information, you can open the violations that caused an alert.

To open a violation from the Alerts window:

1. In the **Main** workspace select **Monitor > Alerts**. The **Alerts** window appears.
2. In the **Alerts** table, select the alert whose violations you want to open.
3. In the **Details** pane, which on the right-hand side of the **Alerts** window, click + next to the violation whose details you want to review.

   You can now review the details of the various violations related to the selected alert.



Note: Violations can also be opened by right-clicking the desired alert, then choosing **Open Violation**.

Deleting Alerts from the Alerts Table

You can delete individual alerts from the Alerts table. This can assist you in managing a large number of alerts and provides an alternative to changing an alert's status.

To delete an alert from the Alerts table:

1. In the **Main** workspace select **Monitor > Alerts**. The **Alerts** window appears.
2. In the **Alerts** table, click once on the Alert you want to delete. The alert is selected.
3. Click . The alert is deleted.



Note: Violations that were the source of the deleted alert are preserved.

Deleting Filtered Alerts from the Alerts Table

SecureSphere enables you to delete filtered alerts from the Alerts pane. When you apply a filter, SecureSphere only displays those alerts that match filter criteria. You can then delete only those alerts that related to the applied filter.

Prerequisite: A currently applied filter.

To delete filtered alerts:

1. In the **Main** workspace select **Monitor > Alerts**. The **Alerts** window appears.
2. From the top right-hand side of the **Alerts** window, click **Actions**. The **Actions** menu appears.

3. From the **Actions** menu, click **Delete Filtered Alerts**. Filtered alerts are now deleted. The alerts pane is now empty.

Note: To remove the filter and display all remaining alerts, from the bottom of the Filter pane, click **Clear**.

Exporting Alerts

Exporting alerts enables you to produce versions of alerts in PDF format or comma separated values (CSV) format.

To export alerts:

1. In the **Main** workspace, select **Monitor > Alerts**. The **Alerts** window appears.
2. From the action buttons in the top-right hand corner of the window, click **Save As**. A dropdown menu is displayed.
3. Select an export format:
   - **PDF**: Exports alerts in graphic and table form to PDF format. You can distribute or print the PDF file.
   - **Alert CSV**: Exports alert data to Alert CSV format.
   - **CSV**: Exports alert data to CSV format. You can use the CSV format to conduct research on alerts in your system in third party tools, create custom charts in Excel for presentations, data mining, and more.
4. SecureSphere begins to query the database to build the file. When completed, the **Download Report** dialog box is displayed.

Note: Depending on the amount of alerts in your system, querying alerts may take an extended period of time.

5. In the **Download Report** dialog box, click **Download Report**. The report is downloaded to the selected location.

Flagging Alerts

Flagging alerts provides you with an effective means of identifying alerts for different reasons.

This might be used when multiple people participate in the review process. For information on using flagging in a multiparty review workflow, see Creating a Multiparty Review Workflow on page 98.

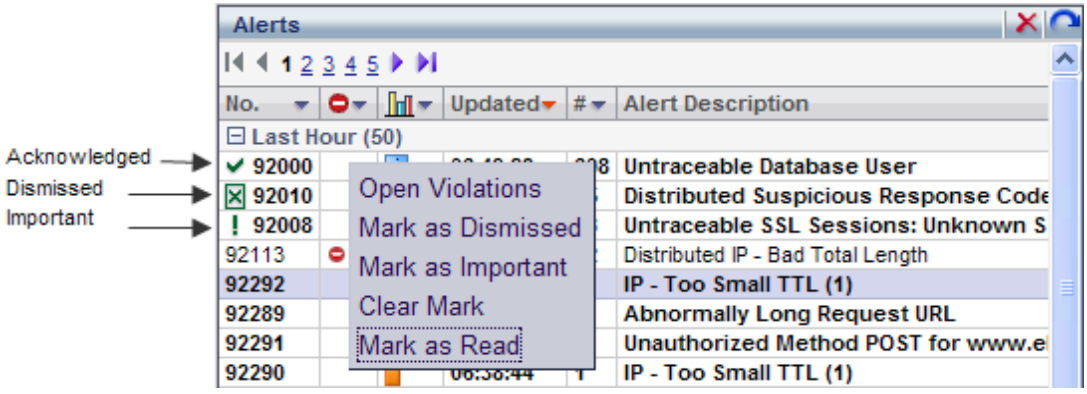


Figure 13: Flagged Alert Example

To flag an alert:

1. In the **Main** workspace select **Monitor > Alerts**. The **Alert** window appears.
2. Right-click on the alert whose status you want to change. The right-click menu is displayed.

3. Select the desired status, as described in the table **Alert Flags** below.

Alert Flags

Option	Description
Unread	Marks the alert as not having been read. This makes the alert bold. This option is only available if the alert has already been read.
Read	Marks the alert as having been read. This removes bold from the alert. This also automatically takes place when you click on an alert that has not yet been read.
Acknowledged	Marked with a green check mark. May be used to acknowledge that the alert has been reviewed, but no action has been taken to resolve the source of the problem. A system event is generated when this option is chosen.
Mark as Dismissed	Marked with a green X. May be used to indicate that an alert has been reviewed, and action has been taken to resolve the source of the problem, or alternatively, that the alert does not represent a threat. A system event is generated when this option is chosen.
Mark as Important	Marked with a green exclamation point. May be used to flag alerts for a third-party reviewer, who can run a filter on the important flag to display only those items that are important. An additional filter option can also be used to further drill down. For an illustration of this process, see Use Case 2: Regular Review and Maintenance. A system event is generated when this option is chosen.
Clear Mark	If the alert has previously been marked as either Acknowledged, Dismissed or Important, this option clears the marking. A system event is generated when this option is chosen.

Creating a Multiparty Review Workflow

In many cases, the SecureSphere administrator may not posses the expertise needed to understand if a violation is a false positive, such as an event conducted by a user in their day-to-day routine, or a threat to data or the system. In these cases, you may need to consult with other specialists to analyze specific alerts and their details.

You may want to consult with the Web Master regarding web based threats.

To do this you may want to develop a multi-party workflow. A multi-party workflow is a system for sharing alerts between users that utilizes SecureSphere alert flagging and the custom assigning of permissions to the various users. For an example of a multiparty review workflow, and how to use flags and filters to manage alerts, see Use Case 2: Regular Review and Maintenance.

Taking Action on Alerts and Violations

SecureSphere enables you to take action on alerts and violations directly from their respective windows, reducing the need to move between windows and search for specific policies. Available actions appear in the form of action buttons in the top left-hand corner of the **Alert and Violation Details** pane, when available.

To take action on alerts or violations:

1. In the **Main** workspace select **Monitor > Alerts**. The **Alert** window appears.
2. From the details pane of the relevant alert or violation, click the desired action button. The following options may be available:

Available actions vary according to alert and violation.

Option	Description
Knowledge Base	Available in all alerts and violations. Opens a separate window with detailed information regarding the attack that has taken place.
Add as Exception	Available in Protocol and Correlation type violations. Creates an exception for the event. For example, if adding an exception for a parameter based violation, SecureSphere might configure the system to ignore the combination of the violated URL and specific parameters. When adding an exception it is recommended you also add a comment explaining why the exception was made. For more information about exceptions see Security Policy Exceptions on page 173.
Edit Policy	Available in all alerts. Opens up the security policy that was violated, enabling you to manually edit it as required.
Add to Profile	Available in Profile violations. Enables you to fine-tune a profile directly from a violation by adding an item or event directly to the profile at the click of a button. For example, if SecureSphere suddenly detects that the @ sign is used as part of a user name when it is forbidden, you may check with your system administrator to discover that the @ sign is now a permitted element in user names. You can add the conditions of the event to the profile at the click of a button, (in this case, the @ sign in a user name) defining it as standard behavior for that profile.
View Signature	Available in Signature violations. Opens a separate window displaying the signature that was detected.

Managing Violations

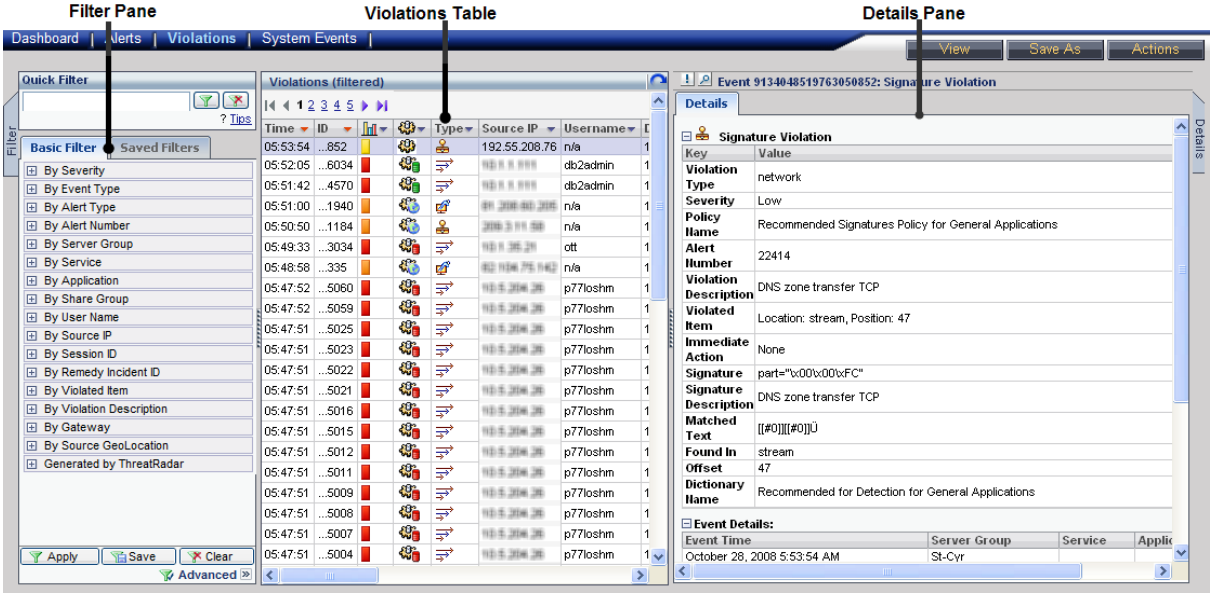
Violations are warnings that an event has taken place which violates a policy defined in SecureSphere. As opposed to alerts, violations provide a specific indication regarding an event that has taken place, including a number of details regarding the offending traffic. This assists you in analyzing violations and determining what type of threat they may pose to your data. This section reviews the following subjects:

Viewing Violations in the Violations Window 100
Understanding Violations 100
Analyzing Violations in the Details Pane 102

Viewing Violations in the Violations Window

To view violations:

- In the **Main** workspace, select **Monitor > Violations**. The **Violations** window appears.



The Violations window has three primary sections:

- **Filter Pane**: Enables you to filter items using one of three filter types: Quick Filter, Basic Filter and Advanced Filter. Violations are filtered in the same manner as Alerts. For instructions on how to filter violations, see Filtering Alerts and Violations on page 109
- **Violations Table**: Provides a summary of violations in a single row. For information regarding violations in the Violations table, see Understanding the Violations Table on page 101
- **Details Pane**: Provides detailed information regarding the violation selected in the Violations table. For information on how to analyze violations using the Details pane, see Analyzing Violations in the Details Pane on page 102

Understanding Violations

Violations contain details regarding a security event. Typically you are first alerted about this event in the Dashboard or Alerts window, then you can drill down to view the violations related to a specific alert. Violations provide detailed information regarding an attack to assist you in determining the circumstances surrounding the attack such as attacker IP, targeted service, and much more. This data can then be used to determine if the violation poses a risk to your data, or if the violation should not have been generated and you need to fine-tune SecureSphere. Fine tuning is conducted by editing a security policy, adding the violation as an exception, adding it to a profile, or generating a report of violations and consulting with the relevant SMEs (For example, web masters).

EXAMPLE: UNDERSTANDING VIOLATIONS

Assume a policy determines that tampering with cookies is a violation that requires further action. If SecureSphere detects that a cookie has been modified, it generates a violation listing the details of the violation, such as the user's IP address, the cookie content, or the name of the policy being violated. You can then analyze these details to determine whether an attack is taking place, or whether the cookie may be modified as part of its handling, and the policy that generated the violation should be fine-tuned.

An integral part of analyzing violations is viewing them in aggregated alerts as described in Understanding Aggregated Alerts on page 94.

Understanding the Violations Table

The Violations table provides a summary of violations generated by SecureSphere. The Violations table is customizable and can display a range of basic information. The following figure shows an example of the Violations table.

Note: Additional columns may be available. For information on how to configure displayed columns, see Adding / Removing Columns from the Alerts and Violations Tables on page 92.

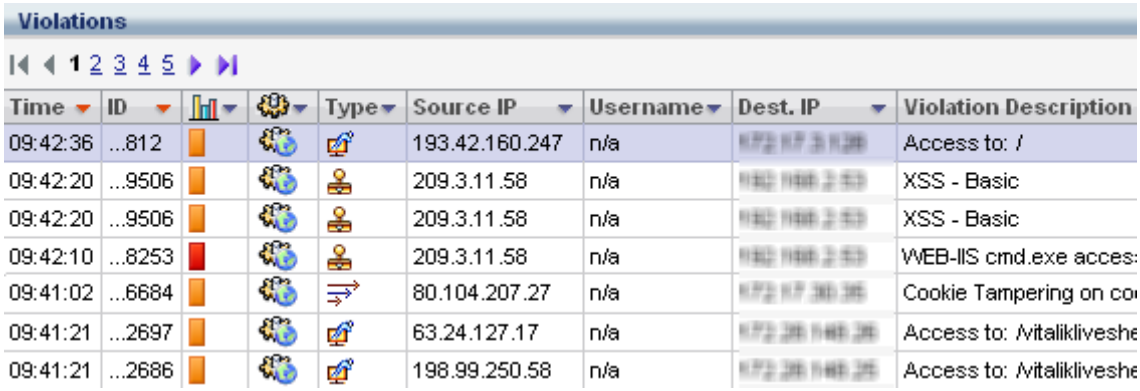


Figure 14: Violations Table

The below table reviews the various columns available in the Violations table.

Note: To sort columns in a table, click on the desired column heading. The arrow direction indicates whether the sort order is ascending or descending based on the selected column.

Violations Table Contents

Column Heading	Description
Time	**Time the violation took place.**
ID	ID number of the violation.
	The level of **severity** configured for the violation. There are four levels of severity: High Medium Low Informative **Note:** Informative is used for very low severity or when an event has taken place which is not a security violation. It may provide additional information.
	**Service** whose policies were violated. The icon represents the event type which can be either the network service, or HTTP service.
Type	The attack type as described in the Alerts table.
Source IP	The IP which caused the alert to be generated (attacker's source IP).
User Name	User name connected with the violation (attacker's user name). If the user name is not available, n/a is displayed.
Destination IP	Destination IP of the violating packets (IP being attacked).
Violation Description	A short description of the violation as defined by SecureSphere. The description is the name of the policy rule.
Server Group	The server group to which the alert relates (being attacked).
Service	Name of the service being used in the attack.
Violated Item	The item being attacked, such as URL, SQL user, SQL table, etc. Alternatively, the type of violation may be displayed (for example, custom violation).

Analyzing Violations in the Details Pane

Violations come in many forms. They can range from a signature violation in which SecureSphere has identified the signature of an attack such as SQL injection, to a cookie violation in which SecureSphere has identified that a cookie has been injected in an attempt to hijack a session, and much more. Once a violation has occurred, the circumstances surrounding the violation are displayed in the Details pane with comprehensive information in textual format. The Details pane is context sensitive, and changes based on the violation being viewed and its source event type. Additionally, it can contain any of a number of tabs to display additional information as may be relevant including the items listed in the table located in the Table below.



Note: For information on the various tools that enable you to see specific the information that caused a violation, see Violation Analysis Tools on page 103.

For information on examining violations that appear in an aggregated alert, see Analyzing Aggregated Alerts on page 95.

Tabs in Violation Details Pane

Parameter	Description
Details Tab	**Violation Name and Details:** Name of the violation that has taken place and details regarding the violation to assist in better understanding its content and context. This field is present in all violations. **Event Details**: Details regarding the event including time of the event, affected server group, service and application, connection IP addresses, direction, and payload (actual hexadecimal and ASCII values used in the offending attack), and Bind Variables. **Additional Information**: There are several primary types of templates: network, or web. A variety of different information may be provided depending on the type of attack. For example, if the violation involved cookie injection, the cookie content may be displayed. This information appears in addition to the violation and event details. **Parameters**: Lists parameters specifically related to the violation. **Cookies**: Lists cookies and their content that were involved with the violating traffic. **Enrichment Data**: Lists enrichment information that was attached to the event as part of an enrichment policy. For more information on enrichment, see Enriching Data on page 372.
Additional Violations Tab	Displays a list of additional violations that were generated from the same security event when available. This can assist you in getting a better overall picture as to the attack taking place. General information listed in this tab is similar to the violation information listed in the Details tab.
Response Tab	Displays the web response page provided to users. This can help you ascertain if any sensitive information may have been leaked that can help hackers access your system or steal information. The Response Tab is only available if is has previously been configured in a policy. It is only available for web violations. Not available for firewall, network protocol validation or stream signature policies, or database violations. For further information see Web Error Pages and Response Pages in Alerts on page 106.

Violation Analysis Tools

SecureSphere provides a number of tools to assist you in understanding the cause of a violation at a glance and drilling down to view further information and conduct analysis directly from the violation details. The table below lists the tools available, while the figure below illustrates the location of these tools in an actual violation.

Violation Analysis Tools

Parameter	Description
Hyperlink	Activates the IP Forensic service, see IP Forensics Service, which is one of the ThreatRadar Reputation services, see ThreatRadar Fraud Prevention Services on page 206 Runs a trace using WhoIS, and displays the location of the source of the violation, marking it on a Google Map with location details. The following information is available on IP addresses: • Geographical location: continent, country, region, state, city, latitude, longitude • Geographical attributes: postal code, time zone, time of the event normalized to source, viewer and MX time zone • Network attributes: AOL flag, connection type, connection speed, routing type, ASN, carrier, second level domain, top level domain • Reverse DNS • Malicious activity indications: www.dshield.org, www.njabl.org, www.spamhaus.org, etc. • Links to relevant Internet resources: DSHIELD, DNSTools and so on.
Filter 	Provides three functions. When this icon is clicked, a dropdown menu is displayed. Options include the following: **Add to Filter**: Adds the selected criteria to the filter, and subsequently marks the applied filter criteria in the filter pane using the filter icon. **Filter According to This Field**: Filters violations by the criteria accompanying the filter. For example, if you click the filter accompanying the session, then select Filter according to this field, all violations with matching sessions are displayed in the violations table. **Open Filtered Violations in a Dialog**: Opens a list of related violations based on the selected filter criteria in a separate dialog box. **Note:** The dynamic filter option is available with the following fields: Source IP, Server group, Service, Application, Session, Username.
Yellow Highlight	Indicates information that triggered the violation. The example in the figure below displays the highlighted information in a violation triggered by Session Hijacking (User Agent Mismatch).

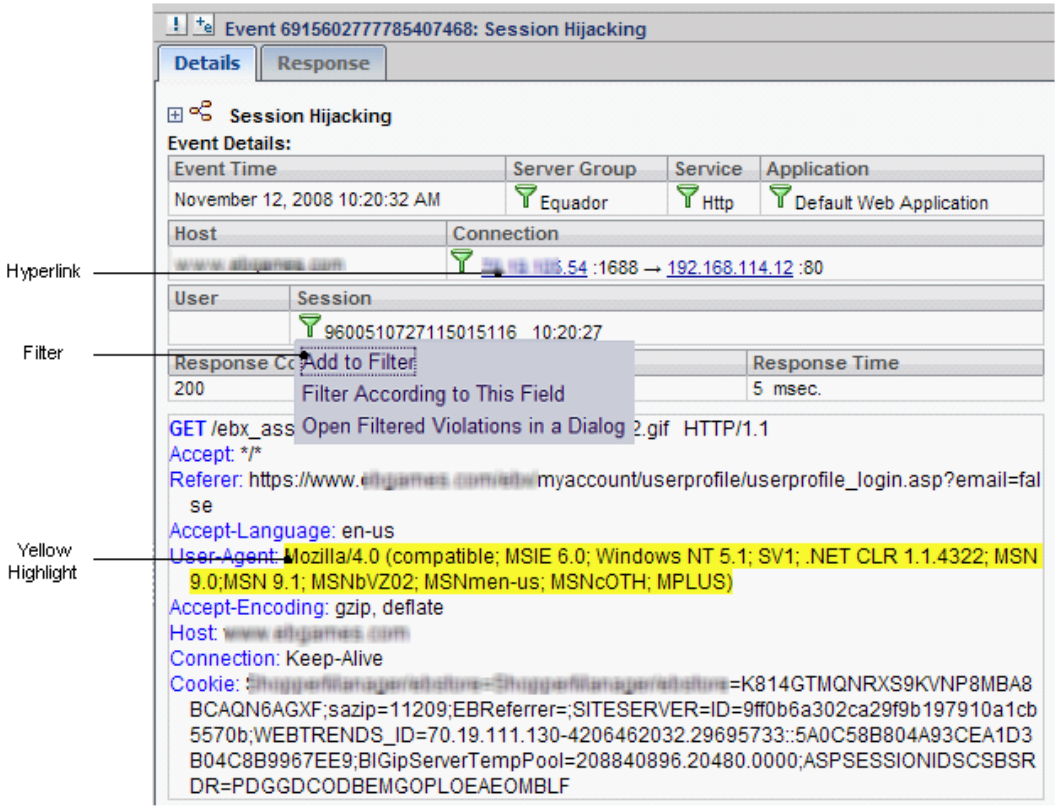


Figure 15: Violation Analysis Tools - Example

Managing Blocked Sources

Blocked sources are the source of communications that have been quarantined for violating a security policy, and have been blocked from accessing the working environment as the result of a followed action. Sources can be blocked by Session ID, IP Address or User, and can be blocked for a specified period of time. This provides you with enough time to examine an event to determine if it represents a threat, or alternatively decide it is valid traffic and unblock it.

Process overview: A security event triggers a followed action of blocking. From that moment on, any communication coming from that source is blocked for a specified period of time.

The process of blocking sources involves two primary steps:

Viewing Blocked Sources 105
Releasing Blocked Sources 106

Viewing Blocked Sources

To view blocked sources:

- In the **Main** workspace, choose **Monitor > Blocked Sources**. The **Blocked Sources** window appears.
  The Blocked Sources window has two primary sections:

- **Filter Pane**: Enables you to filter items by selecting the desired filter from the filter pane. Options include currently blocked sources, or sources blocked in the past 72 hours (even if they have been released). Blocked sources are stored in SecureSphere beyond the 72 hours available here.
- **Blocked Sources Table**: Provides a summary of blocked sources in a single row.

Releasing Blocked Sources

You can release a blocked source (permit traffic from this source in the network) directly from the Blocked Sources window.

To release blocked sources:

1. In the **Main** workspace, select **Monitor > Blocked Sources**. The **Blocked Sources** window appears.
2. Select the Blocked Source(s) you want to release. You can select individual blocked sources by holding down the CTRL key and clicking each source you want to unblock.
3. From the action buttons in the top-right hand corner of the window, click **Actions / Release Blocked Sources**. The sources are released and their communication is once again permitted in the network.

Web Error Pages and Response Pages in Alerts

Servers are often the target of malicious attacks. Typically these attacks generate violations in SecureSphere and are handled accordingly. Though if a server is not properly configured, even if a violation is generated, error pages may be displayed that inadvertently expose sensitive information such as the current operating systems, installed service packs, and other system information which can be used by hackers to gain access your system.

Additionally, sensitive data may be inadvertently displayed in error messages when reporting general errors. For example, credit card details may be displayed to system users that should otherwise have been masked.

In order to track data that was exposed to outside elements, inadvertently leaked or not properly protected, SecureSphere stores the response page (HTML source) in full, then lets you view it in a tab located in violations.

Viewing Web Error Pages and Responses

Usually violations occur in requests as attacks against a protected server. When a violation occurs in the response, it is usually related to information leakage.

Even if the violation was in the request, the response can still contain sensitive data that was leaked, without any violation actually occurring on the response itself. For example, if the gateway is in the emulation mode, the data is leaked from the organization, and the admin is able to review what was leaked using SecureSphere.

Reviewing the web error pages and responses provided to users assists you in evaluating the risk of the information that may have been exposed.

To view web error pages and responses:

1. In the **Main** workspace, choose **Monitor** > **Violations**. The **Violations** window appears.
2. Click the violation that contains the response you want to view. Details regarding the violation are displayed in the **Details** pane.
3. Click **Responses** at the top of the Details pane.



Note: The underlying policy that generates a violation has to be configured in order to display response information, see Enabling Response Page in Violations on page 175. Otherwise you receive a message stating that the **Response page setting is off for this policy**.

Information displayed in the **Response Page** tab depends on the immediate Action defined in the policy that was violated, as follows:

- If the request is blocked, the **Response Page** tab is empty, because no response was received from the server, since it never received the request.
- If the request is not blocked, the **Response Page** tab displays the full response as it was sent to the user.
- If the response is blocked, the **Response Page** tab displays only partial response information.
- If the response is not blocked, the **Response Page** tab displays full response information.
- When the violation is in the response page, the **Response Page** tab provides an indication about the context of the problem with the highlighted violation information, while the sensitive information is masked.

Managing System Events

System Events are events that have taken place in the system such as a login, or a direct result of system changes such as signature updates, changes to configuration, activation of settings, building profiles, automatic profile updates, server start/stop, archiving activities, system related errors or warnings (e.g., predefined threshold definitions being exceeded), and so on. System events may have security or compliance implications, and therefore need to be monitored in much the same way as alerts and violations.

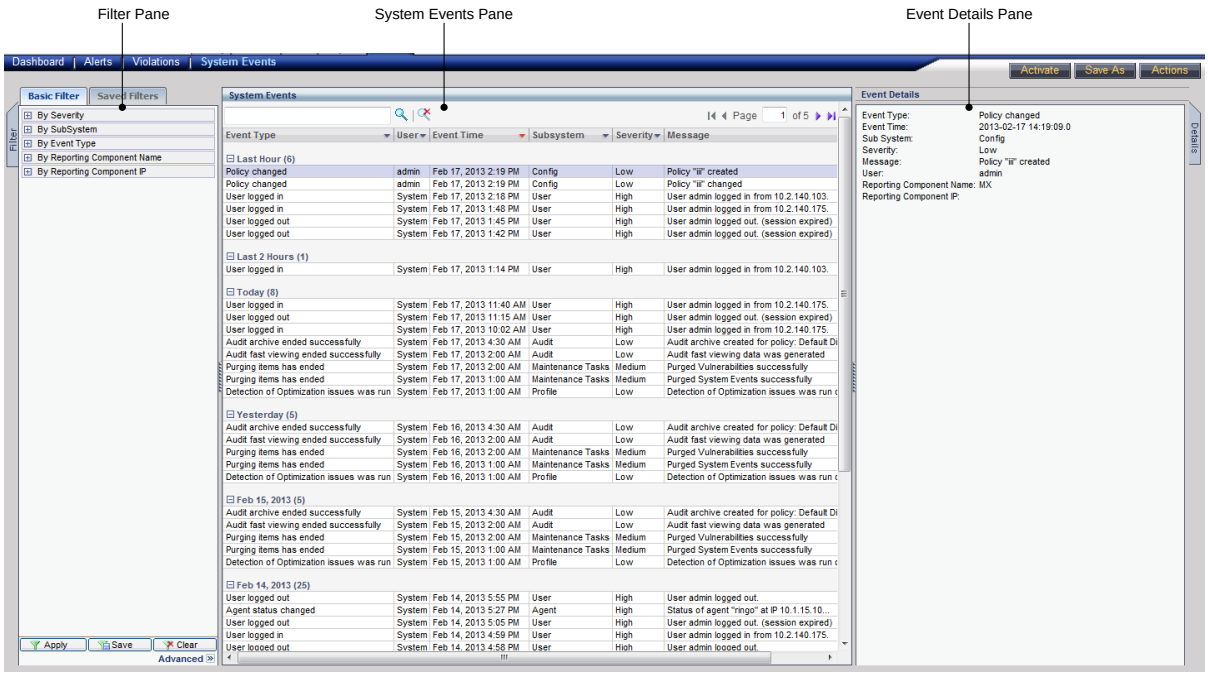
This section reviews the following regarding system events:

Viewing System Events 107
Understanding System Events 108

Viewing System Events

To view System Events:

1. In the **Main** workspace, choose **Monitor > System Events**. The **System Events** window appears.



The System Events window has three primary sections:

- **Filter Pane**: Enables you to filter items by selecting the desired filter from the filter pane. Options include Severity, SubSystem, or Custom.

- **System Events Pane**: Provides a summary of System Events in a single row. For further information about the System Events Table, see Understanding the System Events Table on page 108
- **Details Pane**: Provides detailed information regarding the System Events.

Understanding System Events

System Events provide information about events and changes that have taken place in the system. There are a number of types of system events. The one thing they have in common and that differentiates them from violations is that they were generated by SecureSphere due to changes in the system and not because a policy has been violated by monitored traffic. This assists you in tracking changes to your system that might otherwise have been overlooked, and may in any case represent some form of risk. For a complete list of system events, see List of SecureSphere Events. To understand the different types of system events, see Understanding the System Events Table on page 108.

Understanding the System Events Table

The System Events table provides a summary of System Events generated by SecureSphere.

The table below describes the contents of the System Events table.



Note: To sort columns in a table, click  on the desired column heading. The arrow direction indicates whether the sort order is ascending or descending based on the selected column.

System Events Table Contents

Column Heading	Description
Event Type	The type of event that was generated.
User	The user that generated the system event.
Event Time	The time the event took place.
Subsystem	The subsystem that generated the event.
Severity	The severity of the system event.
Message	A short description of the system event.



Note: Information displayed in the **Details** pane is identical to that displayed in the **System Events** table.

Configuring System Events to Interface with 3rd Party Tools

SecureSphere enables SYSLOG, SNMP trap-based notifications of SecureSphere system events to other event collecting systems. To configure interfacing with 3rd party tools, see Configuring Integration with Third Party Management Systems on page 394.

Filtering Alerts and Violations

SecureSphere contains a wide range of items that are used in day to day operations. Filtering enables you to easily search for these items. For example, you may want to search for a report that you conducted on the completion of Tasks as assigned in SecureSphere. Filtering enables you to easily filter items by a wide variety of categories. The **Filter** pane enables you to easily filter items by using up to three mechanisms. Not all filtering mechanisms are available for all features. The following filtering mechanisms are available:

- **Quick Filter**: Filters by keyword. Only available for Alerts and Violations, and Reporting (by Report name only). For further information, see Using the Quick Filter on page 109.
- **Basic Filter**: Filters by pre-defined categories. For further information, see Using the Basic Filter on page 110.
- **Advanced Filter**: Filters using pre-defined categories. For further information, see Using the Advanced Filter on page 112.

  You can delete filtered alerts from the Filter pane. For further information see Deleting Filtered Alerts from the Alerts Table on page 96.

Using the Quick Filter

Available for Alerts and Violations, the Quick Filter enables you to filter alerts on-the-fly by typing specific criteria and their parameters. SecureSphere then filters existing alerts displaying those that contain these keywords. Keys and values aren't case sensitive. Partial text can be given as an input for string based fields (for example, alert description, etc.). Numeric based fields have to be given as a whole.



Note: The Quick Filter is available for Reports, however search strings are restricted to the specific name of a Report.

Example: You want to quickly view all alerts generated by IP address 10.1.1.122. You simply type IP:10.1.1.122 then click Filter, and all alerts with that source IP address are displayed.

To use the Quick filter:

1. From the **Main** workspace, select **Monitor > Alerts**. The **Alerts** window appears.

   Note: The Quick Filter can also be used to filter Violations. Syntax in quick search is case-sensitive

2. In the **Quick Filter** pane, type a keyword or keywords.
3. Press **Enter** or click the **Filter** button. Alerts matching the filter you configured are listed in the **Alerts** table. To make results more accurate, it is recommended that you use the following syntax when filtering with the Quick Filter.

Quick Filter Criteria

Criteria	Syntax	Description
Number	number:xyz	Number assigned to the alert.
Session ID	session:xyz	Session ID that generated the alert.
Source IP	ip:xyz	IP address that was the source of the generated alert.
Event ID	event:xyz	Event ID that generated the alert.
Alert type	type:xyz	Type of alert: Firewall, Signature, Protocol, Correlation, Custom.
Server Group	server group:xyz	All server groups configured in SecureSphere and to which the logged in user has permission to view.
Violation	violation:xyz	Type of violation that was detected.
User	user:xyz	User with whom the alert is associated.
Description	description:xyz	Description provided by the alert.

Using the Basic Filter

The Basic Filter enables you to filter alerts based on a number of predefined categories. It enables you to select checkboxes that represent a parameter and additionally and in some cases type specific values for example IP address or port number. Additionally, once you have configured a filter, you can save it for future use, enabling its integration into a more organized workflow as described in Creating a Multiparty Review Workflow on page 98.

Example: You are the web master and you participate in the SecureSphere review workflow. The SecureSphere administrator flags all web related alerts for your review. You select the search criteria that include **high severity** and flagged as **important.** Once you apply the filter, you view all items that match the selected criteria, analyze the alert details, and determine whether the alert represents a significant threat.

To use the Basic Filter:

1. From the **Main** worskpace, select **Monitor > Alerts**. The **Alerts** window appears.



Note: The Basic Filter can also be used to filter Violations.

2. Click the plus sign (+) next to the Basic Filter category you want to apply. The category opens providing available filter criteria. Available filter criteria for the Basic Filter include the items in the table below.
3. Configure the filter criteria as required. This can range from selecting a checkbox to selecting operations and entering specific values.
4. Repeat step 2 and 3 to add additional filtering criteria as required.



Note: Using multiple filter criteria enables you to drill down to view specific alerts. However, using excessive filter criteria may lead to empty results. It is recommended to start your search with a minimum amount of criteria and increase as necessary to drill down.

5. Click **Apply**. Alerts are filtered based on the criteria configured and results matching search criteria are displayed.

Basic Filter Criteria

Option	Description
Severity	Filters by severity of the alert: Informative, Low, Medium and High.
Immediate Action	Filters by immediate action taken as part of the violation that generated the alert: None, Block.
Alert Type	Filters by the type of alert: Firewall, Signature, Protocol, Profile, Worm, Correlation, Custom.
Alert Flag	Filters by flag attached to the alert: Important, Acknowledged, Dismissed, and None.
Alert Number	Filters by the number assigned to the alert.
Server Group	Filters by server group configured in SecureSphere and to which the logged in user has permission to view.
Service	Filters by services configured in SecureSphere and to which the logged in user has permission to view.
Application	Filters by application. Enables you to select one or more applications defined in SecureSphere.
User Name	Filters by user to whom the alert is associated.
Source IP	Filters by IP address that was the source of the generated alert.
URL	Filters by URL (or part thereof) which may have been the source of the alert.
Session ID	Filters by session ID that generated the alert.
Event ID	Filters by Event ID that generated the alert. **Note:** When filtering by Event ID, alert details still displays all violations.
Remedy Incident ID	Filters by ID of a Remedy incident opened as a followed action on the alert.
Gateway	Filters by a specific gateway.
Source GeoLocation	Filters by a Geographic location. Only available with license.
Generated by ThreatRadar	Filters by whether or not event was generated by ThreatRadar Reputation Services.
Share Group	Filters by share group alert is associated with.
File Folder	Filters by a specific folder name.
File Parent Path	Filters by the path of the parent folder of a file.
File Name	Filters by a specific file name.
File Extension	Filters by a specific file extension.

Using the Advanced Filter

The Advanced Filter enables you to search for alerts using parameters available in the quick and basic searches in addition to a number of high-level parameters including within a specific time-frame, within the Last Few Days, and as the result of a custom policy. This provides a powerful mechanism to easily find specific alerts.

When filtering alerts, using the Custom Policy match criteria as a filter, only the alerts that were generated by custom policies included in the filter definitions are filtered.

Example: You have discovered that a user that should have been denied access to a specific page on a web site is now authorized to access that page. It is known that the user had not had privileges to that page 2 weeks previous, and that they accessed the page 3 days ago. You may want to search for an alert should have notified you if and when this users privileges were modified, and by whom. You configure a filter with a high severity and within a time frame that includes this period.

To use the Advanced Filter:

1. From the **Main** workspace, select **Monitor > Alerts**. The **Alerts** window appears.



Note: The Advanced Filter can also be used to filter Violations.

2. From the bottom of the **Filter** pane, click **Advanced**. The **Advanced Filter** opens in a separate window.
3. Click the Green Arrow  of the **Available Field** you want to add as part of the search. The field is moved up to the **Enabled Fields** pane. For a list of available advanced filter criteria, see Filter Criteria and Report Data Fields Overview on page 575.
4. Click the plus sign (**+**) next to the desired field.
5. Configure the filter criteria as required. For example, if you want to view all signature alerts, click + next to Alert Type, then add the signature criteria to the **Selected** pane.
6. Repeat steps 3-5 to add additional filtering criteria as required.



Note: Using multiple filter criteria enables you to drill down to view specific alerts. However, using excessive filter criteria may lead to empty results. It is recommended to start your search with a minimum amount of criteria and increase as necessary to drill down.

7. Click **Apply**. Alerts are filtered based on the criteria configured and results matching search criteria are displayed.

CHAPTER 5

ThreatRadar Services

This section describes and it includes the following topics:

- ThreatRadar Reputation Services:
  Introducing ThreatRadar Reputation Services on page 114
  Phishing URLs Service on page 117
  Anonymous Proxy Service on page 119
  Malicious IPs Service on page 121
  IP Forensics Service on page 122
  TOR IPs Service on page 122
- Community Defense Services:
  Introducing Community Defense on page 124
  Community Defense Back Feed on page 124
  Community Defense License on page 130
  Remote File Inclusion on page 130
  SQL Injection on page 131

Introducing ThreatRadar Reputation Services 114
Working with ThreatRadar Reputation Services 114
Licensing Information 117
Phishing URLs Service 117
Anonymous Proxy Service 119
Malicious IPs Service 121
IP Forensics Service 122
TOR IPs Service 122
Introducing Community Defense 124

Community Defense Back Feed 124
Community Defense License 130
Remote File Inclusion 130
SQL Injection 131

Introducing ThreatRadar Reputation Services

ThreatRadar Reputation Services are services that provide an additional security layer which enables identifying threats before they arrive to your system, limiting the effort of analyzing large amount of traffic. Using ThreatRadar services you can better understand where the traffic originates, who creates it and what are the risks. For example, if there is an attack on Internet sites, ThreatRadar provides this information and prevents possible attacks.

The following services are available:

- Anonymous Proxy
- GeoLocation
- IP Forensics
- Malicious IP addresses
- Phishing URLs
- TOR IP addresses
- Web Fraud Services. For more information on Web Fraud Services see ThreatRadar Fraud Prevention Services on page 206.

The services are based on various web data sources that are digested and made available through web interfaces on the ThreatRadar servers. SecureSphere connects to the ThreatRadar servers on a regular basis, to get updates on various attacks and then provides predefined security policies to protect against the attacks.



Note: To use ThreatRadar services, SecureSphere must be constantly connected to the Internet.

ThreatRadar services are hosted on remote web servers at Imperva and made available through the Internet for SecureSphere users providing the following:

- Using dedicated predefined security policies that are constantly updated.
- Analyzing incoming data using forensic/analysis web application.
- Updating existing services or introducing new services without upgrading SecureSphere.

To use ThreatRadar, you need to purchase a license.



Note: ThreatRadar feeds currently track only IPv4 addresses. This is an external issue - as soon as these feeds start tracking IPv6 addresses, ThreatRadar will use them, but they will treated as strings and SecureSphere will see them exactly as it receives them from the feed. IPv6 addresses will not be converted to the long form and SecureSphere will fail to match two identical IPv6 addresses if they are differently expressed: for example, SecureSphere will not recognize that **2001:db8::1:0:0:1** and **2001:db8:0:0:1:0:0:1** are the same.

Working with ThreatRadar Reputation Services

ThreatRadar services are managed using feeds, policies and reports, as follows:

- **Feeds**: A data stream on which the service is based that contains a predefined lookup data set and the information received from the ThreatRadar server. The configuration of the lookup data set defines the type of data that you are interested to get from the ThreatRadar server.
- **Policies**: Predefined policies are provided to protect against the threats. By default ThreatRadar services are configured to alarm only. You can set them to blocking manually by changing policy definitions of each service.
- **Reports**: Predefined reports provide an indication of the service value. You can create custom reports based on the predefined reports.

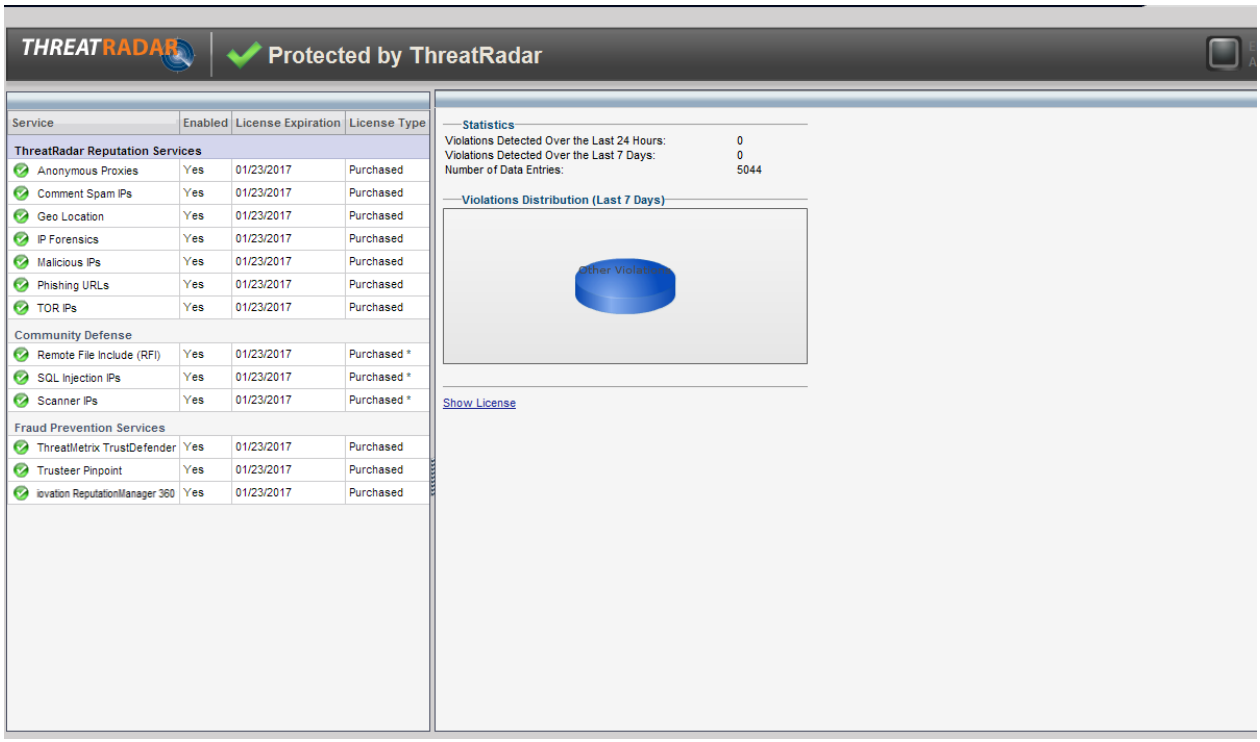


Figure 16: ThreatRadar Dashboard

The services are organized in the left pane of the ThreatRadar window, as described in the following table.

ThreatRadar Services Table

Item	Description
Service	Select the service that you want to view. The right pane displays the details of the service that you have selected in the left pane. **Default**: All Services.
Enabled	Shows if the service is Enabled/Disabled, as follows: • **Yes**: The service is enabled. • **No**: The service is disabled, meaning that no updates and feed additions can be performed and the protection policies are disabled as well. **Default**: Yes. You can enable or disable a service by right-clicking on it and selecting from the menu.
License Expiration	Shows the expiration date. At this date the service becomes disabled.
License Type	The following license types are available: • Evaluation • Emergency • Purchased - An asterisk (*) appearing next to "Purchased" indicates that the MX is participating in Community Defense.

To view the ThreatRadar window:

- In the **Main** workspace, select **ThreatRadar**. The **ThreatRadar** window appears with the **Welcome** window that provides a short review of the ThreatRadar services.
- You can enable or disable all the services together using the **Action** button in the upper right corner of the screen.

Disabling and Enabling ThreatRadar Reputation Services

You can disable ThreatRadar completely. Disabling ThreatRadar stops all the ThreatRadar activities, including the connection to ThreatRadar servers and adding new services, which is possible when services are disabled one by one.

You can enable/disable ThreatRadar in the ThreatRadar window or in the System Definitions window.

To enable/disable ThreatRadar in the ThreatRadar window:

- In the ThreatRadar window:
- Right-click the **All Services** and select Enable/Disable,

  Or
- Click **Actions** and select Enable ThreatRadar/Disable ThreatRadar.

To enable/disable ThreatRadar in the System Definitions window:

1. In the **Admin** workspace, click **System Definitions**.
2. In the **System Definitions** pane, select **ThreatRadar** and Enable/Disable ThreatRadar in the right pane.

All Services

When All Services option is selected, the All Services pane presents a general summary of all the threats found by ThreatRadar services. If you have license only for part of ThreatRadar services, then the following two tables are presented: Licensed Services, which contains all the licensed services you have, and Unlicensed Services, which contains the rest of ThreatRadar services.

The Violations Distribution diagram provides a graphical representation of the number of violations detected by ThreatRadar as a part of the total number of violations detected by SecureSphere. All the violations with their details are displayed in the Violations window, see Managing Violations on page 99.

The following table describes the general ThreatRadar statistics.

ThreatRadar Services Statistics

Statistics	Description
Violations detected over the last 24 hours	The number of violations that were detected by all the ThreatRadar services over the last 24 hours.
Violations detected over the last seven days	The number of violations that were detected by all the ThreatRadar services over the seven days.
Number of data entries	The total number of data entries that were created in Lookup data sets for all the existing feeds.

Licensing Information

Licensing package offered by Imperva contains all the services provided by ThreatRadar. When purchasing SecureSphere, you have 30 days out of the box ThreatRadar evaluation and you can ask for additional trial period of 30 days.

If you do not have a license and you need to protect your system immediately, you can buy a licence or use the Emergency Activation option. The Emergency Activation remains active in your system for three days. You can use up to two Emergency Activations.

To purchase a new license:

- In the top right corner of the ThreatRadar window, click **Buy**. The **Purchase a ThreatRadar Subscription** form appears.

To get a trial version:

- In the top right corner of the ThreatRadar window, click **Request a Trial**. The **Request a Free 30-Day Trial Extension to ThreatRadar** form appears.

To use Emergency Activation:

- In the top right corner of the ThreatRadar window, click **Emergency License Activation**. Now you can use all the ThreatRadar services.

Phishing URLs Service

The Phishing URLs service alerts on phishing incidents that took place in real time.

Phishing is the criminally fraudulent process of attempting to acquire sensitive information such as usernames, passwords and credit card details by masquerading as a trustworthy entity in an electronic communication. Communications purporting to be from popular social web sites, auction sites, online payment processors or IT administrators are commonly used to lure the unsuspecting public. Phishing is typically carried out by e-mail or instant messaging, and it often directs users to enter details at a fake website whose look and feel are almost identical to the legitimate one.

Phishing URLs Summary

When the Phishing URLs service is selected in the ThreatRadar window, the Summary pane presents a summary of all the threats found by that service.

The Data Retrieval Problems table provides a summary of problems that occurred during the update process. For example, if the predefined lookup data set configuration could not be applied.

The Violations Distribution diagram provides a graphical representation of the number of violations detected by the Phishing URLs service as a part of the total number of violations detected by SecureSphere. All the violations with their details are displayed in the Violations window, see Managing Violations on page 99.

The following table describes the Phishing URLs service statistics.

Phishing URLs Service Statistics

Statistics	Description
Violations detected over the last 24 hours	The number of violations that were detected by all the Phishing URLs service over the last 24 hours.
Violations detected over the last seven days	The number of violations that were detected by all the Phishing URLs service over the last seven days.
Number of data entries	The number of data entries that were created in Lookup data sets for all the feeds in this service.

Managing the Phishing URLs Service

The Phishing URLs service is managed by feeds and policies. In addition you can generate reports to indicate the value provided by this service.

Phishing URLs feeds are based on predefined lookup data sets provided by Imperva ADC, see Predefined Lookup Data Sets on page 366.

The ThreatRadar - Phishing URLs policy is a predefined policy provided by Imperva ADC. This policy is also based on the partially defined lookup data set, which means that to use the predefined policy, you need to complete the configuration.

You can create Phishing URLs lookup data sets. Using these data sets you can create custom policies. You cannot use custom data sets in the predefined policies.

To complete the configuration of the phishing lookup data set provided by ADC:

1. In the Phishing URLs Feeds section, click the pencil button. The **Lookup Data Set: ThreatRadar - Phishing URLs** window appears.
2. Select **Configuration** and set the table **Fishing URL Parameters** below.
3. Click **Update**.

Fishing URL Parameters

Parameter	Description
Data Set Name	A read-only name.
Add Domain Names	Add the domain names that you want to protect using the Phishing URLs service. Click the green **+** button and type the domain name.
Choose Fields	The Landing Formatted URL check box is read only and you cannot clear it.

To create a Phishing URLs lookup data set from ThreatRadar:

1. From the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Scope Selection** dropdown list, select **Lookup Data**. Existing **Lookup Data sets** are displayed.
3. In the **Lookup Data Sets** title bar, click  **New**, then click **From ThreatRadar** > **Phishing URLs**. The **ThreatRadar-Phishing URLs** dialog box appears.
4. Select **Configuration** and set the table **Phishing URL Dataset Parameters** below.
5. Click **Create**. The new **Lookup Data Set** is created and added to **Lookup Data** category in the **Global Objects** window and also in the **ThreatRadar** window.

Phishing URL Dataset Parameters

Parameter	Description
Data Set Name	Type the lookup data set name.
Add Domain Names	Add the domain names that you want to protect using the Phishing URLs service. Click the green **+** button and type the domain name.
Choose Fields	The Landing Formatted URL check box is read only and you cannot clear it.

Anonymous Proxy Service

The Anonymous Proxy service enables identifying anonymous proxy servers and assists to prevent hacking attempts by deploying transparent proxy detection policies in SecureSphere.

What is an Anonymous Proxy

In general, a proxy server is a server (a computer system or an application program) that acts as an intermediary for requests from clients seeking resources from other servers.

An anonymous proxy server (sometimes called a web proxy) attempts to anonymize web surfing. There are different varieties of anonymizers. One of the more common variations is the open proxy. Because they are typically difficult to track, open proxies are especially useful to those seeking online anonymity, including computer criminals.

Some anonymizing proxy servers may forward data packets with header lines such as HTTP_VIA, HTTP_X_FORWARDED_FOR, or HTTP_FORWARDED, which may reveal the IP address of the client. Other anonymizing proxy servers, known as elite or high anonymity proxies, only include the REMOTE_ADDR header with the IP address of the proxy server, making it appear that the proxy server is the client.

Proxies can also be installed in order to eavesdrop upon the dataflow between client machines and the web. In that case accessed pages, as well as all forms submitted, can be captured and analyzed by the proxy operator, who might be a hacker.

Anonymous Proxy Summary

When the Anonymous Proxy service is selected in the ThreatRadar window, the Summary pane presents a summary of all the threats found by that service.

The Data Retrieval Problems table provides a summary of problems that occurred during the update process. For example, if the predefined configuration could not be applied.

The Violations Distribution diagram provides a graphical representation of the number of violations detected by the Anonymous Proxy service as a part of the total number of violations detected by SecureSphere. All the violations with their details are displayed in the Violations window, see Managing Violations on page 99.

The following table describes the Anonymous Proxy service statistics.

Anonymous Proxy Service Statistics

Statistics	Description
Violations detected over the last 24 hours	The number of violations that were detected by all the Anonymous Proxy service over the last 24 hours.
Violations detected over the last 7 days	The number of violations that were detected by all the Anonymous Proxy service over the last seven days.
Number of data entries	The number of data entries that were created in Lookup data sets for all the feeds in this service.

Managing the Anonymous Proxy Service

The Anonymous Proxy service is managed by feeds and policies. In addition you can generate reports to indicate the value provided by this service.

Anonymous Proxy feeds are based on predefined lookup data sets provided by Imperva ADC, see Predefined Lookup Data Sets on page 366.

The Anonymous Proxy policy is a predefined policy provided by Imperva ADC. This policy is also based on the predefined lookup data set.

You can create Anonymous Proxy lookup data sets. Using these data sets you can create custom policies. You cannot use custom data sets in the predefined policies.

To create an Anonymous Proxies lookup data set from ThreatRadar:

1. From the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Scope Selection** dropdown list, select **Lookup Data**. Existing **Lookup Data sets** are displayed.
3. In the **Lookup Data Sets** title bar, click  **New**, then click **from ThreatRadar** > **Anonymous Proxies**. The **ThreatRadar-Anonymous Proxies** dialog box appears.
4. Select **Configuration** and set the options as shown in the table **Anonymous Proxy Service Options** below.
5. Click **Create**. The new **Lookup Data Set** is created and added to **Lookup Data** category in the **Global Objects** window and also in the **ThreatRadar** window.

Anonymous Proxy Service Options

Option	Description
Data Set Name	Type the lookup data set name.
Choose Filters	Choose the filters. Using these filters the lookup data set selects the data that you want to receive from the ThreatRadar server, as follows: • **All**: Select to get all the anonymous proxies. • **High Anonymous**: Select to get details of anonymous proxies with higher level of anonymity, which enables attackers to hide better. The highest level of anonymity is defined as camouflage proxies.
Choose Fields	Select the additional information that you can get from the ThreatRadar server.

Malicious IPs Service

The Malicious IPs service protects applications against known malicious IP addresses and sources.

The Malicious IPs service enables deploying policies that detect requests originating in known malicious IP addresses, for example, spam servers, bots, attack automation servers and so on. The data generator for this service collects and processes data from various free and paid IP black lists.

Malicious IPs Summary

When the Malicious IPs option is selected in the left pane, the summary of Malicious IPs found by ThreatRadar services is presented in the right pane.

The Data Retrieval Problems table provides a summary of problems that occurred during the update process. For example, if the predefined configuration could not be applied.

The Violations Distribution diagram provides a graphical representation of the number of violations detected by ThreatRadar as a part of the total number of violations detected by SecureSphere. All the violations with their details are displayed in the Violations window, see Managing Violations on page 99.

The following table describes the Malicious IPs service statistics.

Malicious IPs Service Statistics

Statistics	Description
Violations detected over the last 24 hours	The number of violations that were detected by all the Malicious IPs service over the last 24 hours.
Violations detected over the last 7 days	The number of violations that were detected by all the Malicious IPs service over the last 7 days.
Number of data entries	The number of data entries that were created in Lookup data sets for all the feeds in this service.

Managing the Malicious IPs Service

The Malicious IPs service is managed by feeds and policies. In addition you can generate reports to indicate the value provided by this service.

Malicious IPs feeds are based on predefined lookup data sets provided by Imperva ADC, see Predefined Lookup Data Sets on page 366.

The Malicious IPs policy is a predefined policy provided by Imperva ADC. This policy is also based on the predefined lookup data set.

You can create Malicious IPs lookup data sets. Using these data sets you can create custom policies. You cannot use custom data sets in the predefined policies.

To create a Malicious IPs lookup data set from ThreatRadar:

1. From the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Scope Selection** dropdown list, select **Lookup Data**. Existing **Lookup Data sets** are displayed.
3. In the **Lookup Data Sets** title bar, click  **New**, then click **from ThreatRadar** > **Malicious IPs**. The **ThreatRadar-Malicious IPs** dialog box appears.
4. Select **Configuration** and set the table **Malicious IP Configuration Option** below.
5. Click **Create**. The new **Lookup Data Set** is created and added to **Lookup Data** category in the **Global Objects** window and also in the **ThreatRadar** window.

Malicious IP Configuration Options

Option	Description
Data Set Name	Type the lookup data set name.
Choose Filters	Choose the filters. Using these filters the lookup data set selects the data that you want to receive from the ThreatRadar server, as follows: • **All**: Select to get all the malicious IP addresses. • **Persistently Intensive**: Select to get the IP addresses that have been persistently reported as malicious over the period of a few days. • **Recently Intensive**: Select to get the IP addresses that have been reported as malicious during the last day.
Choose Fields	Select the additional information that you can get from the ThreatRadar server.

IP Forensics Service

The IP Forensics service enables resolving IP addresses to their geographical location and displaying that location on a visual map (based on the Google Maps API).

The IP Forensics service is accessible from the Violations window, see Managing Violations on page 99. In the ThreatRadar window you can only view the information.

TOR IPs Service

The TOR IPs service protects applications against hackers that are using the TOR network to execute attacks.

The TOR IPs service enables deploying policies that detect requests originating from known TOR IPs. The data on TOR IPs are generated directly from the TOR network.

What is a TOR IP

The Onion Router (TOR) is a free software implementation of second-generation onion routing, which is a system that claims to enable its users to communicate anonymously on the Internet.

TOR network hides the original IP of the client, by thus providing an anonymous access to web applications. Although the original purpose of the network was to provide anonymity for noble causes such as freedom of speech, many hackers abuse that anonymity to conduct attacks against web applications.

A user of the TOR network runs a proxy server on his/her computer. Internet-facing software can then access TOR through a SOCKS interface. Once inside a TOR network, the traffic is sent from router to router, the TOR software periodically negotiating a virtual circuit through the TOR network, ultimately reaching an exit node at which point the cleartext packet is forwarded on to its original destination. Viewed from the destination, the traffic appears to originate at the TOR exit node. TOR employs cryptography in a multi-layered manner (hence the Onion routing analogy), ensuring perfect forward secrecy between routers.

TOR's anonymity can be exploited by hackers to clear their traces, therefore a free access to a TOR network might become a security threat to your system.

TOR IPs Summary

When the TOR IPs service is selected in the ThreatRadar window, the Summary pane presents a summary of all the threats found by that service.

The Data Retrieval Problems table provides a summary of problems that occurred during the update process. For example, if the predefined configuration could not be applied.

The Violations Distribution diagram provides a graphical representation of the number of violations detected by the TOR IPs service as a part of the total number of violations detected by SecureSphere. All the violations with their details are displayed in the Violations window, see Managing Violations on page 99.

The following table describes the TOR IPs service statistics.

TOR IPs Service Statistics

Statistics	Description
Violations detected over the last 24 hours	The number of violations that were detected by all the TOR IPs service over the last 24 hours.
Violations detected over the last 7 days	The number of violations that were detected by all the TOR IPs service over the last seven days.
Number of data entries	The number of data entries that were created in Lookup data sets for all the feeds in this service.

Managing the TOR IPs Service

The TOR IPs service is managed by feeds and policies. In addition you can generate reports to indicate the value provided by this service.

TOR IPs feeds are based on predefined lookup data sets provided by Imperva ADC, see Predefined Lookup Data Sets on page 366.

The TOR IPs policy is a predefined policy provided by Imperva ADC. You cannot change it.

Introducing Community Defense

The Community Defense service provides you with an additional level of enhanced protection, based on analyzing events received from the community of SecureSphere users.

As a SecureSphere customer, you can share your events on the Imperva cloud. The data that you and other users share on the cloud is examined by Imperva ADC and in response you receive Community Defense feeds, providing an additional level of defense to your system.

The sharing of events by the SecureSphere customers community provides a new source of information for Imperva ADC, allowing better analysis and faster protection.

The following options are available:

- Sharing event data on the Imperva cloud and the receipt of Community Defense feeds free of charge, see Community Defense Back Feed on page 124.
- Buying a Community Defense license that provides you with Community Defense feeds, see Community Defense License on page 130.

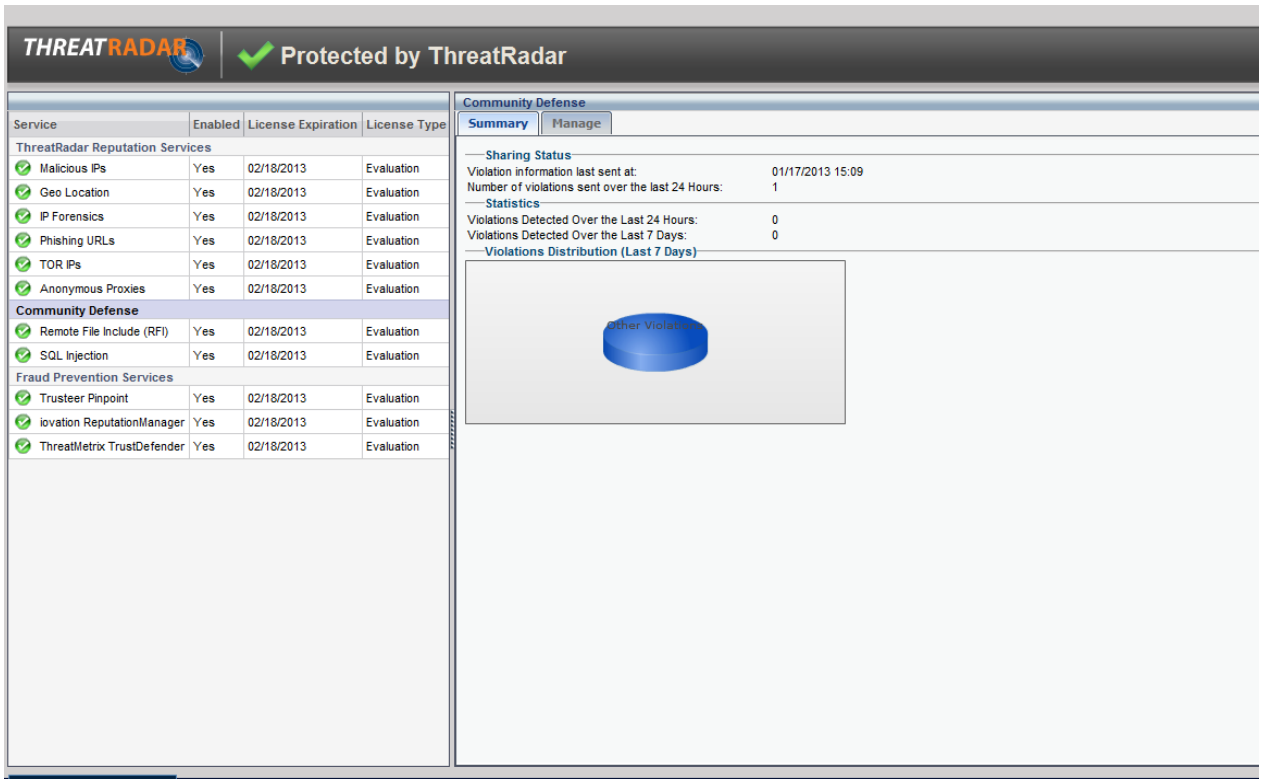


Figure 17: Community Defense Services

Community Defense Back Feed

When working in the Community Defense back feed mode, you send the violations information to the Imperva cloud. Imperva learns this information and then provides protection against all the violations that Imperva customers shared on the cloud. Constant learning and analyzing of various alerts and violations enables providing better and faster protection.

The following table presents an overview of all the steps required to configure violations sharing on the Imperva cloud.

Community Defense Back Feed Configuration Overview

#	To do	Description	See
1	**Set SecureSphere Policies**	Configure security policies to allow sending events to the Imperva cloud.	How Security Policies Work with Community Defense on page 125
2	**Define Communication Parameters**	Make sure the parameters required for the communication with the Imperva cloud are set.	Defining Cloud Communication Parameters on page 126
3	**Set up Violation Sharing**	Review the information that your SecureSphere is going to send to the cloud.	Managing Violation Data Sharing on page 126
4	**Edit Masking**	If required, you can mask fields that were not masked by default before sending the data to the cloud.	See the sub-procedure Edit Masking in Managing Violation Data Sharing on page 126
5	**View Summary**	You can view the status and the summary at any time.	Understanding Community Defense Summary on page 129

How Security Policies Work with Community Defense

In order to share violations data on the Imperva cloud, some security policies by default allow sending events to the cloud. The policies for which you can allow sending events are all predefined policies that were provided to you by Imperva ADC. You can change this configuration, so that SecureSphere will stop sending events to the cloud.

The following table presents the policies for which sending events to the cloud is allowed by default:

Policy Type	Policy Name
Stream Signature	Recommended Signatures Policy for General Applications
HTTP Protocol Validation	Web Protocol Policy
HTTP Protocol Signatures	Recommended Signatures Policy for Web Applications
Web Service Custom	All predefined policies
Web Worm	Web Worm Policy
Web Profile	Web Profile Policy
Web Service Correlated Validation	Web Correlation Policy
Cookie Signing Validation	Cookie Signing Policy
Firewall Policy	Firewall Policy

To configure security policy to stop sending events to the cloud:

1. In the **Main** workspace, click **Policies** > **Security**. The **Security Policies** window appears.
2. Select the policy you want to configure and open the **Advanced** tab on the right panel.
3. In the **Community Defense Settings** area, clear the **Allow sending event information to Community Defense** checkbox.

4. Click **Save**.

Defining Cloud Communication Parameters

Before sending the events information to the Imperva cloud, you need to make sure your system has obtained the exchange keys and the Cloud Customer ID required for the communication. The Key is used to encrypt and decrypt messages exchanged between the SecureSphere and the Imperva cloud. With the Key ID SecureSphere informs the cloud which key this SecureSphere uses. For each customer on the Imperva cloud there is a unique Cloud Customer ID.

The Key, Key ID and Cloud Customer ID are obtained from the Imperva cloud automatically. If these parameters were not obtained, Fraud Prevention Service appears as not being fully configured. In that case you need to generate the keys.

To generate the exchange keys:

1. Make sure SecureSphere is connected to the Internet.
2. Make sure SecureSphere is connected to the Imperva cloud.
3. In the **Main** workspace, click **ThreatRadar. The ThreatRadar window appears.**
4. In the **Services** pane, select **Community Defense** and in the right pane, select the **Manage** tab.
5. In the **Cloud Communication Parameters** area, click **Generate**. The **Confirm** dialog box appears, asking if you are sure you want to generate new Key Pair.
6. Click **Yes**. The new key and key ID are generated.

Managing Violation Data Sharing

The First step in the violation data sharing process is to collect and review the data that you are going to share on the Imperva cloud. You can send the data as is, while the default masking definitions are used. By default SecureSphere masks some event parts. You can edit masking definitions, hiding sensitive data that you do not want to expose when sharing the violation information.

To define violation data sharing:

1. In the **Main** workspace, click **ThreatRadar. The ThreatRadar window appears.**
2. In the **Services** pane, select **Community Defense** and in the right pane, select the **Manage** tab.
3. Select **Collect Violation Information**.
4. Wait enough time for some violations to be collected (a few seconds in a high-volume site, or a few minutes in a low-volume site).
5. In the Data sample area, click **Show**. The **Sent data sample** window appears.
6. Review the data presented in the **Sent data sample** window and decide if this is the data that you want to share or you need to change the masking definitions to hide your sensitive data.



Notes:

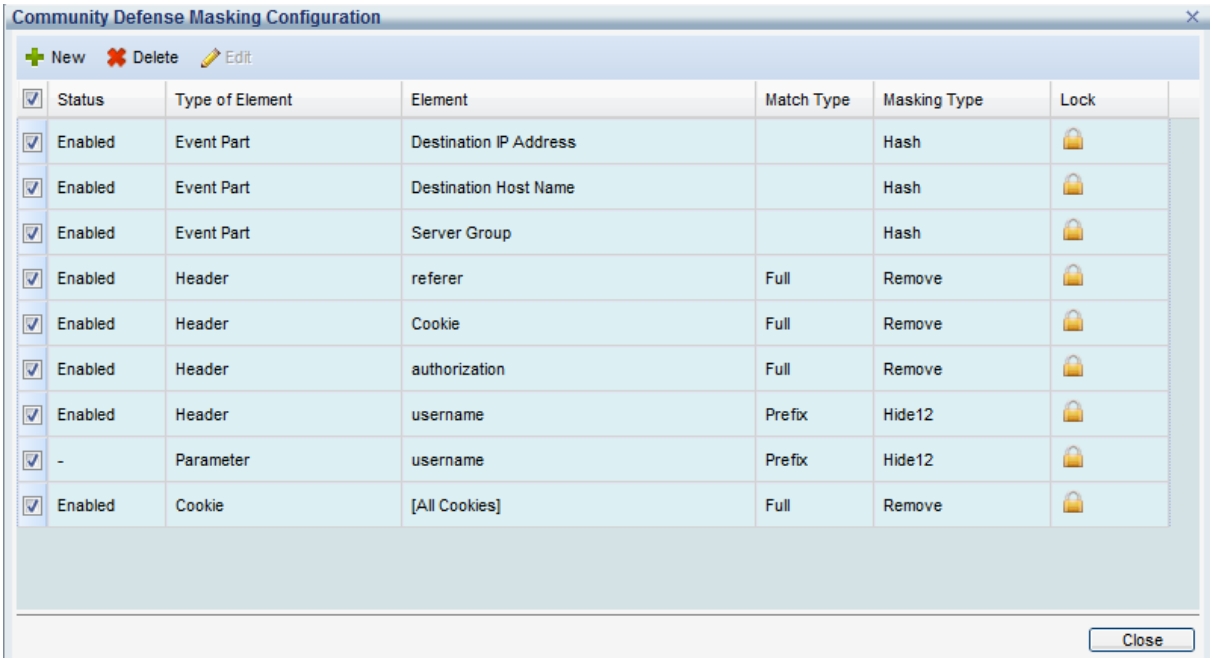
- It is recommended to use the default masking definitions.
- Talk to the application owners and find out if there are additional parameters or headers (except for referer, authorization and username*) that contain sensitive non-encrypted information, and if there are, mask them too.
- Expect that the out of the box configuration is sufficient.
- All the masking definitions provided by Imperva are locked, you can only Enable/Disable them.

7. Once the data sample appears satisfactory, select **Send Collected Information** and click **Save**.

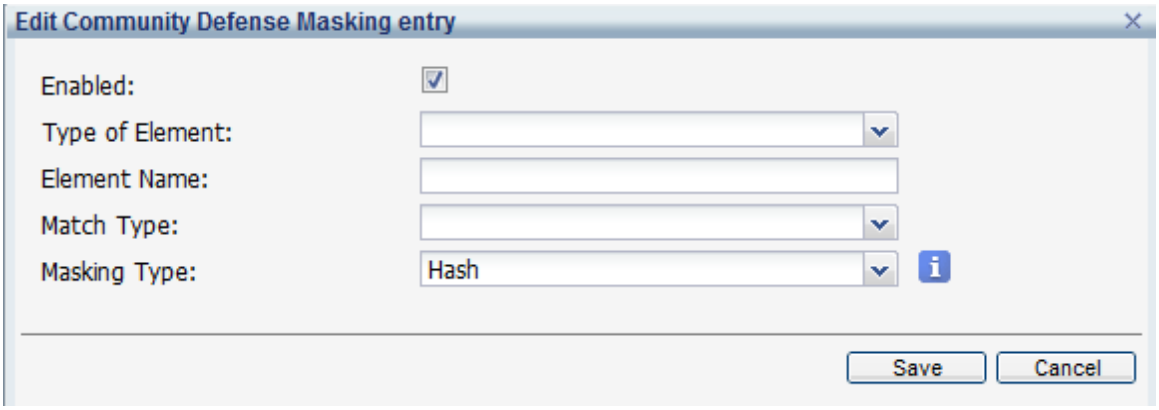
To edit masking definitions:

1. In the **Main** workspace, click **ThreatRadar. The ThreatRadar window appears.**
2. In the **Services** pane, select **Community Defense** and in the right pane, select the **Manage** tab.

3. In the **Masking Configuration** area, click **Edit**. The **Community Defense Masking Configuration** window appears, presenting the default masking definitions.



4. To add new a new masking definition, click **New**. The **Edit Community Defense Masking Entry** dialog box appears.



5. Set the Community Defense Masking parameters as shown in the table below.
6. Click **Save**. The **Edit Community Defense Masking Entry** dialog box closes.

Community Defense Masking Parameters

Option	Description
Enabled:	Select this checkbox to enable your masking definition and clear it to disable the definition. Default: Enabled.
Type of Element:	To which part of the HTTP Request or Event the masking definition applies. The following options are available: • Header • Parameter • Cookie • Event Part - Destination IP Address • Event Part - Destination Geo Location • Event Part - Destination Host Name • Event Part - Session ID • Event Part - Server Group
Element Name:	The name of the element that you want to mask, as follows: • If the type of element selected above is Header, Parameter or Cookie, enter here the name, or initial part of the name, of the header, parameter or cookie that you want to mask. • If the type is an Event Part, this field is disabled.
Match Type:	Enables searching your data for the items that you want to mask according to the Element Name. You can search for the **Full** name match or **Prefix** match. If you entered a prefix in the Element Name, select Prefix here. If you've entered the element's full name, select Full here. For example, to mask all parameters that start with the string "Pass", Enter "Pass" in the element name and select Prefix here. To mask only the parameter "Pass", but not the parameter "Passport", select "Pass" in the element name and "Full" here. **Note:** If the Type of Element is Event Part, this field is disabled.

Option	Description
Masking Type:	Enables defining how to mask the value of the matched field. The following options are available: • **Hash**: Hashing is a way to keep some irreversible information about the value which still allows Community Defense to correlate events with the same original value without knowing the actual values. This option enables masking sensitive data while Imperva knows the connection between the object and the event without access to the actual data. For example, in case your server group names contain sensitive information that you do not want to send to Community Defense. Hashing is consistent, which will cause the name "ServerGroup1" to always be hashed the same, for example to "ABC". "ServerGroup2" will always be hashed the same, for example to "XYZ". Thus, Community Defense will receive ABC as the server group name in all violations related to the server group ServerGroup1, and XYZ in all violations related to ServerGroup2, and it will be able to correlate the events according to the server group without actually being exposed to the names. • • **Hide first 12 characters**: Enables replacing the first 12 characters with asterisks. • • **Remove**: Enables removing of the entire string. Default: Hash.

Understanding Community Defense Summary

When the Community Defense service is selected in the ThreatRadar window, the Summary pane presents a summary of all the threats found by that service.

The Violations Distribution diagram provides a graphical representation of the number of violations detected by the Community Defense service as a part of the total number of violations detected by SecureSphere. All the violations with their details are displayed in the Violations window, see Managing Violations on page 99.

The following table describes the Community Defense service sharing status.

Community Defense Service Sharing Status

Statistics	Description
Violation information last sent at	The last time when the violation information was sent from your SecureSphere to the Imperva cloud.

The following table describes the Community Defense service statistics.

Community Defense Service Statistics

Statistics	Description
Violations detected over the last 24 hours	The number of violations that were detected by all the Community Defense service over the last 24 hours.
Violations detected over the last 7 days	The number of violations that were detected by all the Community Defense service over the last seven days.

Community Defense License

Community Defense service allows leveraging the existing Imperva customer base and its products into creating quality feeds of security-relevant information. This service incorporates customers' SecureSphere results into new feeds in ThreatRadar.

You can buy a Community Defense license that provides all the security feeds included in the Community Defense service without sending your violations to the Imperva cloud. In this case you will be able to use all the Community Defense feeds without sharing your data.



Note: You can purchase the Community Defense license only if you have the ThreatRadar Reputation Services license.

Remote File Inclusion

The Remote File Inclusion service provides signature protection against the Remote File Inclusion vulnerability. This signature is created by Imperva ADC using the information SecureSphere customers community share on the Imperva cloud.

Remote File Inclusion (RFI) is a type of vulnerability which allows an attacker to include a remote file, usually through a script on the web server. The vulnerability occurs due to the use of user-supplied input without proper validation, potentially exposing the site to the following security issues:

- executing unauthorized code on the web server
- executing unauthorized code on the client-side, such as JavaScript which can in turn lead to attacks such as cross site scripting (XSS)
- Denial of Service (DoS) attacks
- Data theft and/or manipulation

Remote File Inclusion Summary

When the Remote File Inclusion service is selected in the ThreatRadar window, the Summary pane presents a summary of all the threats found by that service.

The Data Retrieval Problems table provides a summary of problems that occurred during the update process. For example, if the predefined configuration could not be applied.

The Violations Distribution diagram provides a graphical representation of the number of violations detected by the Remote File Inclusion service as a part of the total number of violations detected by SecureSphere.

Managing the Remote File Inclusion Service

The Remote File Inclusion service is managed by a special dictionary and a signature based on this dictionary, all created by Imperva ADC. In addition you can generate reports to indicate the value provided by this service.

SQL Injection

The SQL Injection service is created on the basis of analysis Imperva ADC performs on the events received from SecureSphere customers that experienced SQL injection attack. The events were generated by the Web Service Correlated Validation Policy. One of the policy rules is SQL Injection rule, see SQL Injection on page 131.

SQL Injection Summary

When the SQL Injection service is selected in the ThreatRadar window, the Summary pane presents a summary of all the threats found by that service.

The Data Retrieval Problems table provides a summary of problems that occurred during the update process. For example, if the predefined configuration could not be applied.

The Violations Distribution diagram provides a graphical representation of the number of violations detected by the SQL Injection service as a part of the total number of violations detected by SecureSphere customers.

Managing the SQL Injection Service

The SQL Injection service is managed by feeds and policies. In addition you can generate reports to indicate the value provided by this service.

CHAPTER 6

Risk Management

Managing the risk inherent in the day to day operation of your network is of critical importance to maintaining a secure work environment and protecting your data. SecureSphere provides a set of tools that enables you to scan network assets for vulnerabilities, analyze risk using standardized scoring techniques, mitigate risk, and track mitigation of vulnerabilities to create an audit trail which can facilitate compliance with regulatory bodies.

This section reviews the various tools provided by SecureSphere to assist you in Managing Risk and includes the following topics:

Introduction to SecureSphere Risk Management 132

Scanning and Assessing Web Services 135

Analyzing Risk 143

Managing Vulnerabilities 150

Introduction to SecureSphere Risk Management

SecureSphere maintains the integrity of your resources by enabling you to import web vulnerabilities from third party tools.

Additionally, SecureSphere is equipped with a built-in Risk Analysis tool that displays the number of detected but as of yet unmitigated vulnerabilities, and displays risk by data type such as financial, credit card or SOX, or by network component. These tools are used at various stages and come together to provide a comprehensive solution to managing risk. The following figure illustrates the basic Risk Management workflow in SecureSphere.



Risk Management Process Overview

Risk management is cyclical process that involves four primary stages. The following table lists the various stages that are involved in this process.

Managing Risk Tasks Overview

	Action	Description	For more information, see...

	Action	Description	For more information, see...
2	**Scanning and Assessing**	SecureSphere integrates with external web scanners to test web servers for vulnerabilities. **Recommended Schedule**: Once a week. Before start of day on Monday.	Scanning and Assessing Web Services on page 135
3	**Analyzing (Risk)**	SecureSphere provides an intuitive, easy-to-use interface to examine risk to assets by network component (site)and data category (such as PCI, credit card, etc.). This enables you to conduct risk analysis and easily determine the current protection status of data and network components, vulnerability status, severity of unmitigated vulnerabilities and more. **Recommended Schedule**: Once a week, directly after scheduled assessments are run.	Analyzing Risk on page 143
5	**Managing (Vulnerabilities)**	Once vulnerabilities have been detected, they are managed from a central workbench, enabling you view details and conduct a number of tasks such as marking them with a priority, assigning them an owner, creating a task for tracking progress, and manually or automatically mitigating them. Furthermore you can track the progress of mitigation and generate regular reports for reporting purposes. **Recommended Schedule**: As often as required to manage vulnerabilities and mitigate risk following analysis.	Managing Vulnerabilities on page 150

Understanding the Risk Management Window

The Risk Management window provides a wide selection of options that enable you to configure assessment policies, conduct risk assessment and manage vulnerabilities.

To view the Risk Management window:

- In the **Main** workspace, click **Risk Management**. The **Risk Management** workspace appears displaying the **Risk Management** console by default.

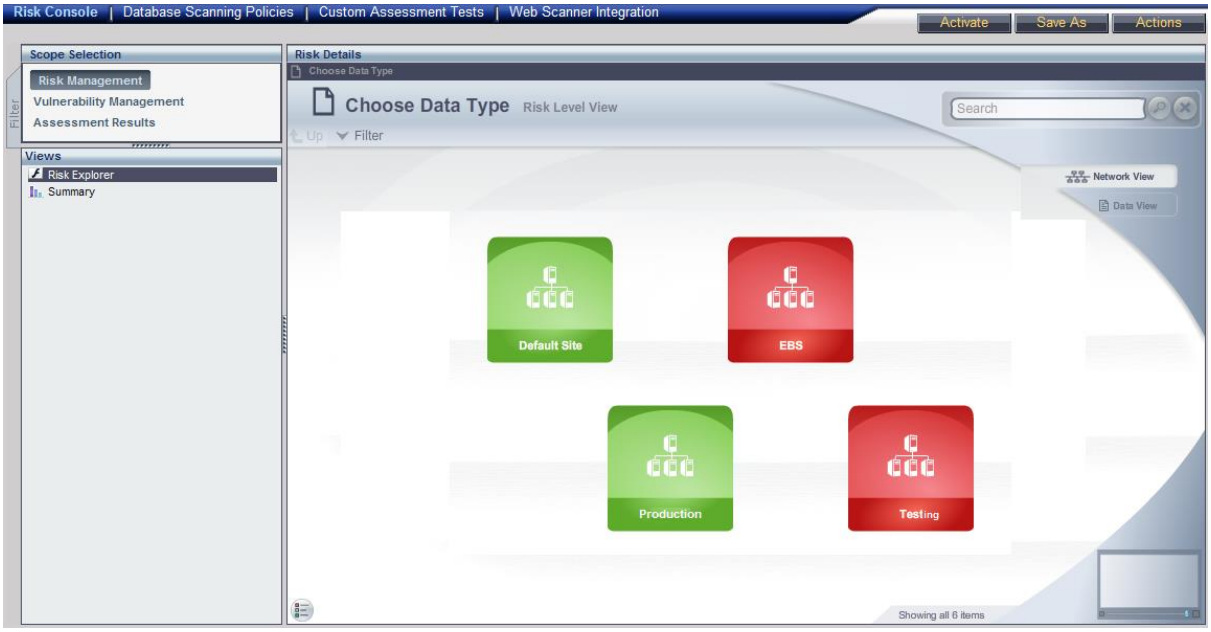


Figure 18: Risk Management Window

The Risk Management window consists of a number of options, each represented by links in the Risk Management navigation bar, and include:

- **Risk Console**: Enables you to view Risk based on scoring, manage vulnerabilities and view assessment results. For information on Risk Console options, see the table **Risk Console Views** below.
- **Web Scanner Integration**: Enables you to import a file from supported web scanners and have SecureSphere automatically mitigate web vulnerabilities that have been detected. For more information see Scanning and Assessing Web Services on page 135.

Risk Console Views

Risk Assessment Views	Description
Risk Management	Analyzes data types present in your network and attaches a CVSS score to assist you in understanding the current level of risk. For more information see Analyzing Risk on page 143.
Vulnerability Management	Enables you to manage vulnerabilities including assigning owner, setting priority, configuring a due date, and more. For more information see Managing Vulnerabilities on page 150.
Assessment Results	Enables you to track which assessments were run and which failed, and for what reason. For further information, see Analyzing Assessment Results

Scanning and Assessing Web Services

This section reviews the various aspects of scanning and assessing web servers for vulnerabilities in SecureSphere. It reviews the following subjects:

Web Service Vulnerabilities Overview 136
Integrating Web Scanners Task Overview 136
Conducting a Web Scan and Preparing Results 137
Creating and Configuring a Web Scanner Integration Policy 138
Running a Web Scanner Assessment Job 139
Configuring SecureSphere to Work with WhiteHat Sentinel 140
Configuring SecureSphere to Work with QualysGuard 142

Web Service Vulnerabilities Overview

Web Vulnerability Management is a primary component in SecureSphere Risk Management. It provides integration between SecureSphere and a number of leading web application vulnerability scanners, enabling the detection and mitigation of vulnerabilities detected in web applications by web vulnerability scanners. First, a scan is conducted with an external web scanner. Then, the results are imported into SecureSphere where they are used to automatically create security policies that mitigate vulnerabilities identified on your web servers. You can then manage, prioritize, track and mitigate these vulnerabilities, and conduct regular follow up. Making web scanning and assessment a regular part of your Risk Management routine helps to guarantee that your web servers are protected, and reduces the ability of malicious users to negatively impact your system.



Note: For a list of supported scanners, see Conducting a Web Scan and Preparing Results on page 137.

Once scan results are imported, SecureSphere automatically creates user-editable security policies that mitigate detected vulnerabilities. The mitigated vulnerabilities are saved in a database and can then be used to generate reports.

Scanner results can be imported into SecureSphere in the following ways:

- Locally, when the results are saved as a file on the local host. This is useful when scanning is ad-hoc or an internal process.
- Remotely, via HTTP or HTTPS, from a predefined location. This is useful when the scanner is provided as a remote service.

Integrating Web Scanners Task Overview

Integrating web scanners into the SecureSphere workflow is comprised of a number of standard steps. The table below lists the steps in mitigating vulnerabilities discovered by web scanners.

Integrating Web Scanners - Task Overview

	Action	Description	For more information, see...

	Action	Description	For more information, see...
1	**Run a Scan and Prepare Results**	Run a web scan using a supported 3rd party web scanner.	Conducting a Web Scan and Preparing Results on page 137
2	**Create a Scanner Integration Policy**	Import the results of your webscan and create an integration policy.	Creating and Configuring a Web Scanner Integration Policy on page 138
3	**Run Web Scanner Assessment Job**	Run the assessment job to mitigate all vulnerabilities discovered by your web scanner.	Running a Web Scanner Assessment Job on page 139

Conducting a Web Scan and Preparing Results

The first step in detecting and mitigating vulnerabilities discovered using a web scanner is to conduct a scan and remove false positives.

To conduct a web server scan and prepare results:

1. Scan your web servers using a supported web scanner.
2. In the scanner, review the scan results and remove false positives.
3. Export the list of the vulnerabilities that you want to import into the vulnerability workbench, manage and possibly mitigate to a convenient location. Supported file types are as follows:
   - **IBM AppScan**: `*.xml`
   - **HP WebInspect**: `*.scan`
   - **NTObjectives**: use the **Export to Imperva Xml** option.
   - **Cenzic**: use the **Export to Imperva Xml** option.
   - **WhiteHat**: a different procedure is required, see Configuring SecureSphere to Work with WhiteHat Sentinel on page 140.
   - **Qualys**: a different procedure is required, see Configuring SecureSphere to Work with QualysGuard on page 142.
4. Create and configure a scanner integration policy as described in Creating and Configuring a Web Scanner Integration Policy on page 138.



Notes:

- Vulnerabilities that have been mitigated through SecureSphere may show up as vulnerabilities during later scans even if the application is scanned through SecureSphere. This is because in most cases web scanners do not conduct attacks, only run small tests to probe for vulnerabilities. If a specific vulnerability has already been mitigated and later appears during a test, it can usually be disregarded. During an actual attack it will be detected and blocked by SecureSphere.
- When uploading a scanner result file to the MX, only the first 10K vulnerabilities are uploaded.

Creating and Configuring a Web Scanner Integration Policy

Once you have scanned your web servers and have exported the list of vulnerabilities to one of the supported file formats as described in Conducting a Web Scan and Preparing Results on page 137, you are ready to import that list and create SecureSphere policies to mitigate those vulnerabilities.

To create a scanner integration policy:

1. In SecureSphere, select Risk Management > Web Scanner Integration. The **Web Scanner Integration** window appears.
2. In the **Scanner Integration Policies** pane, click . The **Create New Policy** window appears.
3. Type a **Name** and **Description** for the new policy.
4. Optionally, select a **Followed Action** that will be conducted when the policy is run.
5. Click **Create**. A new policy is created and its details are displayed in the **Details** pane.
6. In the **Details** pane, define the parameters in the table below.
7. Click the Apply To tab and select the services to which you want to apply this policy. For more information, see Applying Security Policies to the Objects in the Sites Tree on page 319



Note:

- The web vulnerability security policies are applied to the web services that you select here and not to the services identified by scanners. For each service you can define one policy only.
- IPv6 addresses are not displayed in the **Apply To** tree.

8. Optionally, click the Scheduling tab to define custom scheduling. For detailed instructions on how to configure scheduling, see Configuring Action Sets and Followed Actions on page 75.
9. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Option	Description
Name	The name of the policy.
Description	A description of the policy.
Followed Action	Action you want to be conducted once the policy is run. Select an action set from the dropdown list. For more information on followed actions, see Working with Action Sets and Followed Actions on page 326.
Path to local report	Uploads the scan results saved on your local computer. Use this option for local files. To upload the results file, click **Upload File** and browse to locate the file.

Path to remote report	Uploads the scan results file from the web server on which it was saved using the HTTP protocol.
	For directions on how to use this field to pull vulnerabilities from WhiteHat Sentinel server, see Creating a Web Scanner Integration Policy Using Vulnerabilities from WhiteHat Sentinel on page 142.
	Note: Only the HTTP and HTTPs protocols are supported.
	Enter the URL that contains the IP address of the required web server.
	Use the following syntax:
	`protocol://username:password@host/full-path-of-file`
	For example, if the file is saved in `www.myhost.com/scanresults/last_scan`
	then in **Path to remote report**, enter:
	`http:johnbrown:securitypassword@www.myhost.com/scanresults/last_scan.xml`
	If the IP address of the required web server is 10.1.1.1 and the results are saved in `C:/scan_results/si_sample_vuln_report.xml`, then in **Path to remote report parameter**, enter:
	`http://10.1.1.1/scan_results/si_sample_vuln_report.xml`

Close Missing Vulnerabilities	Automatically closes vulnerabilities that were previously discovered using this policy but are no longer present (most likely due to having already been mitigated).
Use System Proxy Definitions	Enables you to import scan results through a proxy server. In order to use this feature you must also configure proxy settings in the **Admin** workspace under **System Definitions** > **External HTTP Settings**. For more information, consult with your SecureSphere administrator.

Running a Web Scanner Assessment Job

This procedure describes how to run a web scanner assessment job based on an existing policy. Running a web scanner assessment automatically mitigates vulnerabilities detected by your web scanner and imported into a web scanner party. For information on how to configure a web scanner policy, including guidelines for archiving and purging, see Creating and Configuring a Web Scanner Integration Policy on page 138.

To run a web service assessment job:

1. In SecureSphere, select Risk Management > Web Scanner Integration. The **Web Scanner Integration** window appears.
2. In the **Scanner Integration Policies** window, select the web scanner assessment policy you want to run. That job's details are displayed in the **Details** pane.
3. Click the **Action** button in the top right-hand corner of the screen, then click **Run Now**. The Assessment Job is run.

   Once the job has been completed SecureSphere reports vulnerabilities which can then be managed and mitigated. For more information on managing vulnerabilities see Managing Vulnerabilities on page 150.



Note: A number of other actions are available for web scanner assessments from the action button. These include **Run Now, Import** and **Upload File**.

Configuring SecureSphere to Work with WhiteHat Sentinel

The integration between Imperva and WhiteHat allows shared customers to fully realize the synergies between the Sentinel web vulnerability scanner service and SecureSphere Web Application Firewall. The integration includes the WAF's automatic ability to pull from the scanner the information about vulnerabilities and immediately mitigate these vulnerabilities by creating adequate custom security policies.

SecureSphere imports from the Sentinel server and mitigates the following vulnerability types:

- Cross Side Scripting (XSS)
- SQL Injection
- Directory Traversal
- Directory Indexing
- HTTP Response Splitting



Note: Other vulnerabilities reported by Sentinel are not automatically mitigated by SecureSphere.

SecureSphere and WhiteHat Integration Tasks Overview

#	Action	Description	For more information, see...
1	**Create a Web API Key.**	Create a Web API Key on your WhiteHat Sentinel account. This key is used as a shared secret that allows SecureSphere to authenticate itself to the WhiteHat server when it requests information about vulnerabilities.	Creating a Web API Key on WhiteHat Sentinel on page 141
2	**Create a request query.**	Create a request query for SecureSphere to download information about vulnerabilities.	Creating a Request Query on page 141

#	Action	Description	For more information, see...
3	**Configure the Web Vulnerability Management active module.**	On SecureSphere, create and configure the Web Vulnerability Scanner active module.	Creating a Web Scanner Integration Policy Using Vulnerabilities from WhiteHat Sentinel on page 142

Creating a Web API Key on WhiteHat Sentinel

To Create a Web API key:

1. Login to the Sentinel account.
2. Click the Account tab.
3. If the Web API Key field is blank, click **Generate Key**.
4. Copy and paste the Web API key into an API URL request.

For full documentation on the Web API Key, refer to WhiteHat documentation, as follows:

https://sentinel.whitehatsec.com/help/api.html#web_api_key

Creating a Request Query

SecureSphere uses a web request query to retrieve vulnerability information from WhiteHat. The query consists of a specific path, `https://sentinel.whitehatsec.com/api/vuln/`, followed by the set of parameters that are used to perform the following:

- Authenticate the SecureSphere to Sentinel server and identify the account from which vulnerabilities are to be pulled (Required).
- Define the format of the information retrieved about each of the vulnerabilities found (Required).
- Define a filter on the vulnerabilities to be retrieved, e.g., limiting the vulnerabilities retrieved to those found on a particular site, after a particular date, with a particular severity and so forth (Optional).

The tables below describe all the parameters required for the request query, and some of the optional parameters that can be used for filtering. For full information, refer to the WhiteHat Sentinel documentation.

Required Parameters and Their Values

Parameter name	Parameter description	Parameter value
`key`	The authentication parameter. It also identifies the account from which vulnerabilities are pulled.	The web API key that is generated on the Sentinel management application.
`display_attack_vectors`	Determines whether to display attack information in the report.	1
`display_param`	Determines whether to display the name of attacked parameter in the report.	1

EXAMPLE:

https://sentinel.whitehatsec.com/api/vuln/?key=d1234595-56bd-4077-9af2-a63d5c6e41b7&**display_param=1&**display_attack_vectors=1

This request pulls all the vulnerabilities found by the Sentinel Service on the customer's account.

Optional Parameters for Filtering

Parameter name	Parameter description	Parameter value	Comments
query_status	Determines the vulnerabilities to be returned based on the status of the vulnerability. Separate multiple status indications by a comma.	open/closed	
query_class	Determines the results to be returned based on the vulnerability class, using the short name for each class. Separate multiple classes by a comma. For the proper format of class names, see the Short Names of Vulnerability Classes for Queries table.	See WhiteHat documentation for the complete list.	
query_severity	Determines the vulnerabilities to be returned based on severity level. Separate multiple values with a comma.	1 [low], 2 [medium], 3 [high], 4 [critical], 5 [urgent]	https://sentinel.whitehatsec.com/api/vuln/?query_severity=10

EXAMPLE:

Supported open vulnerabilities with critical severity.

The following URL extracts all open vulnerabilities with critical severity that can be mitigated in SecureSphere:

https://sentinel.whitehatsec.com/api/vuln/?key=d1234595-56bd-4077-9af2-a63d5c6d41b7&display_param=1&display_attack_vectors=1&query_status=open&query_class=XSS,DirIndex,DirTrav,ResSplit,SQL &query_severity=5

For the complete list of optional parameters that can be used to filter vulnerabilities, consult WhiteHat documentation.

Creating a Web Scanner Integration Policy Using Vulnerabilities from WhiteHat Sentinel

To pull the information from the WhiteHat Sentinel server, use the Path to remote report field and enter the URL you created, as described in Creating a Request Query on page 141.

For the rest of the Web Vulnerability Management policy configuration procedures, refer to Creating and Configuring a Web Scanner Integration Policy on page 138.

Configuring SecureSphere to Work with QualysGuard

QualysGuard is a "Software as A Service" web scanner.

After a scan is performed, Qualys users must manually download the scan report from the Qualys site as XML file, and then import the scan report in a web scanner integration policy.

To import:

1. From the Qualys site, download the scan report as XML file and save the file.

2. In SecureSphere, in the **Main** workspace, click **Risk Management**.
3. Click **Web Scanner Integration**.
4. If you have not already done so, create a Scanner Integration Policy.
5. In the **Details** tab, under **Policy Parameters**, click **Upload File**.
6. Browse to the location of the scan report you downloaded from the Qualys site.

The remainder of the process is exactly the same as for other scanners.

Analyzing Risk

Data Risk Analysis is the process of managing the risk posed to your data. This is accomplished by viewing information regarding vulnerabilities that have not been mitigated as they relate to SecureSphere components such as Sites, Server Groups and Servers, and as related to data type such as credential or credit card data. SecureSphere provides you a dynamic Risk Explorer that utilizes color coding to provide a visual queue as to risk status, and enables you to zoom in on specific types of data and network components to gauge the risk to these components. It is additionally equipped with a Risk Summary view that summarizes risk details in easy to read tables.



Notes:

- The schedule for calculating risk score can be customized under **Admin** > **Maintenance** > **Risk Update**. By default, it is scheduled to update every morning at 5:00 AM. For more information see your SecureSphere administrator.
- Servers with IPv6 addresses are not displayed in the Network Risk Explorer.

The following subjects assist you in understanding how risk is calculated, navigating in the Risk Management module, and main tasks involved in risk management:

CVSS Risk Scoring 143
Analyzing Risk Workflow 144
Examining Risk with Risk Views 144
Analyzing Risk Details 148

CVSS Risk Scoring

SecureSphere uses the Common Vulnerability Scoring System (CVSS) to analyze the risk presented by the vulnerabilities discovered in your network and data, and subsequently assign a Risk score. CVSS is "an open framework for communicating the characteristics and impact of IT vulnerabilities." It is maintained by the National Institute of Standards and Technology, as part of the SCAP framework. Scoring vulnerabilities using CVSS provides an accurate model for measuring the risk inherent in discovered vulnerabilities, and subsequently in prioritizing them for mitigation.

The CVSS scores each vulnerability on a scale of 0 to 10 based on the effect the vulnerability has and the effort required to exploit it. The CVSS scoring function can be extended to include additional factors such as the type (value) of the information to the organization such as whether information may be sensitive (for example, credentials or credit card numbers) and available remediation options.



Note: When calculating risk, mitigated vulnerabilities are treated as if they are open. Only closed vulnerabilities are not factored into risk calculation.

After vulnerabilities have been discovered by SecureSphere, a risk score is applied to the various elements as displayed in the SecureSphere Risk Explorer using color coding. This provides a visual, easy to use interface to assist you in analyzing risk posed to your data and allowing you to drill down and view a breakdown of the vulnerabilities discovered.

For more information about CVSS, see http://nvd.nist.gov/cvss.cfm http://

Analyzing Risk Workflow

The Risk Explorer provides you with color coded data that assists you in gauging risk posed by vulnerabilities to your system. It enables you to get an overall view of the components in your network and the current risk they are exposed to. The following steps represent an example of how to work with the Risk Explorer to get an overview regarding risk posed to your system, and to drill down to see further details regarding that risk.

1. Select a risk view as described in Examining Risk with Risk Views on page 144.
2. Analyze your red zones. Red indicates that the object is at risk.
3. Hover over the at risk (red) object. The object zooms in and displays information regarding the object such as the number of objects contained in the parent object.
4. Single click the red object to drill down.
5. Repeat step 4 until you reach that asset's details window as described in Analyzing Risk Details on page 148. That object's details are displayed in both graphic and tabular format.



Note: Alternatively to view the details click the Information icon of an object.

6. Review the object's status details and vulnerability statistics as described in Analyzing Risk Details on page 148.

Examining Risk with Risk Views

SecureSphere provides a set of views that assist you in analyzing risk from various perspectives.

To analyze risk with risk views:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** appears.
2. In the **Scope Selection** pane of the **Risk Console**, click **Risk Management**. The **Risk Explorer** appears displaying the **Risk Explorer** view by default.



Figure 19: Risk Management Window

SecureSphere provides a number of views to assist you in examining risk. These include:

Main Risk Views

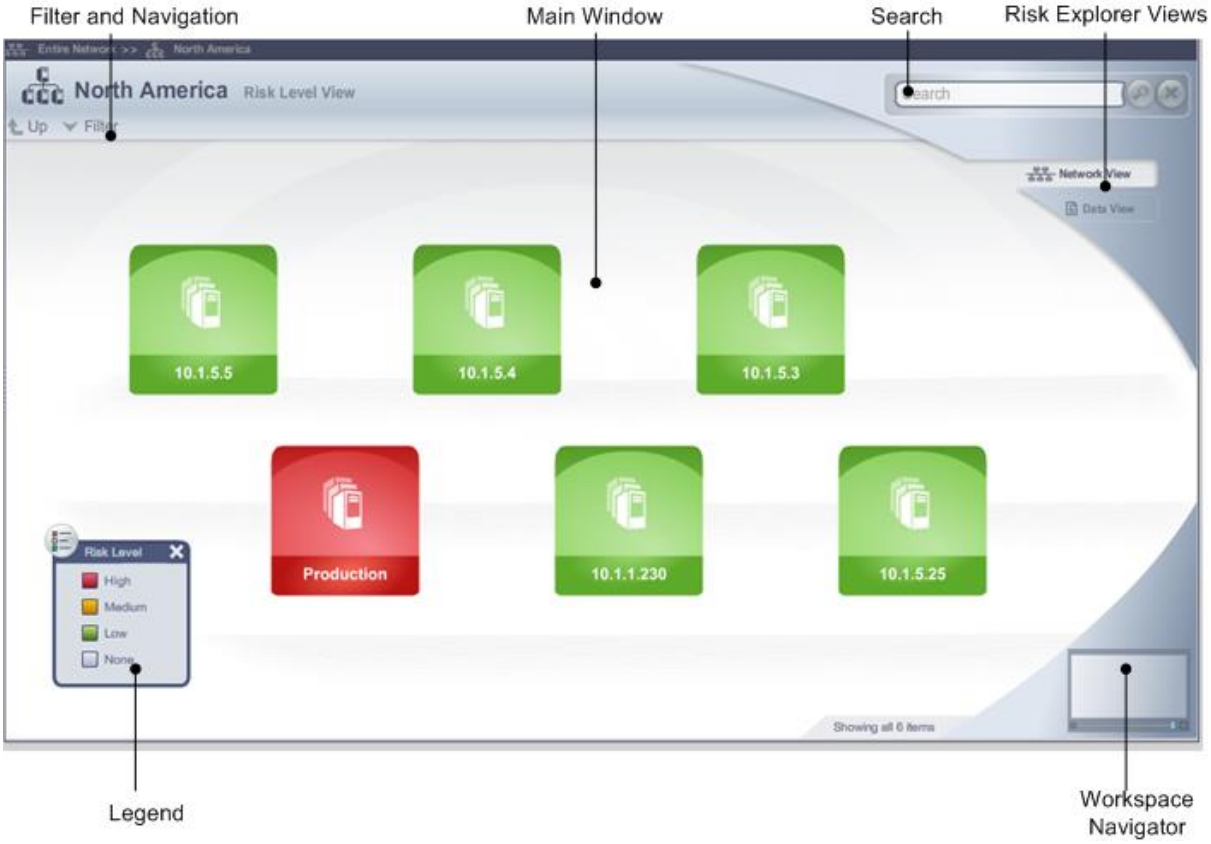
Name	Description
Risk Explorer	Displays details regarding risk scoring in the Risk Explorer using color coding to indicate the Risk Level of the different network object or data category. For more information on using the Risk Explorer see Understanding the Risk Explorer on page 145.
Risk Summary View	Provides information regarding risk in graphic and tabular format assisting in getting an overview of risk provided to different elements.

Understanding the Risk Explorer

Information displayed in the Risk Explorer is the result of assessment tests that have been conducted and the vulnerabilities and data that have been discovered. This information is based on vulnerabilities that have been detected on network assets, and aggregated by network topology and data type.

To view the Risk Explorer:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** appears.
2. In the **Scope Selection** pane of the **Risk Console**, click **Risk Management**. The **Risk Explorer** appears displaying the risk status of data and services. Risk is calculated based on vulnerabilities and sensitive data that have been discovered as a part of assessment tests.



The Risk Explorer provides a number of tools to assist you in selecting and displaying data. These include the items in the following table.

Risk Explorer Components

Tool	Description
Filter and Navigation	Enables you to configure a filter on which to display risk information based on properties such as risk level and protection coverage. Additionally enables you to move one level up. For instructions on configuring Risk views see Examining Risk with Risk Views on page 144.
Main Window	Displays the primary risk objects (data types and network nodes), lets you view risk at a glance and drill down to view details regarding the items contained in these objects. For more information see Managing System Events on page 107.
Legend	Displays a legend that reflects the selected filter. For example, if you filter data by Risk Level, the legend displays the various risk levels. Legend entries auto-hide after a short period of time and can be displayed by clicking the legend icon.
Search	Enables you to search for an item in the explorer by name. For information on using the search, see Conducting a Search in the Risk Explorer on page 148.
Views	Enables you to toggle between viewing risk by SecureSphere components (Network View) and data types (Data View). For more information on toggling between views see Toggling between Data and Network View in the Risk Explorer on page 146.
Workspace Navigator	Enables you to zoom in and out on the Risk Explorer by dragging the scrollbar. This can be helpful when your network contains a large number of items by making them smaller so they can all be viewed at once. You can alternatively use the mouse to zoom in and out on the work space by holding down the mouse button, and moving the mouse up or down.

Drilling Down and Navigating in the Risk Explorer

SecureSphere enables you to drill down on the various objects displayed in the Risk Explorer. By clicking an object you can zoom into the next level object or display vulnerability details. There are three ways to navigate among objects, each displaying a different type of data:

- **Hover:** Expands the object to display parameters regarding that object such as the server groups or services it contains.
- **Click**: Drills down to the next level object. For example, if you single click on a site, the Risk Explorer then displays the server groups that are part of that site.
- **Information Icon** : Displays details regarding the selected item. For example, if you click the information icon located on a server, details regarding that server's vulnerabilities is displayed, including the information that can be seen in Risk Details and Vulnerability Management Workflow on page 149.

  Note: To go to the previous level, click outside of the object in the workspace.

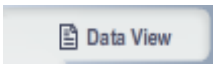
  For information on analyzing details, see Analyzing Risk Details on page 148.

Toggling between Data and Network View in the Risk Explorer

SecureSphere enables you to view vulnerability by both network asset and by data category. You can toggle between these views by selecting the desired view from the Risk Explorer.

To select an object for viewing:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** appears.
2. In the **Scope Selection** pane of the **Risk Console**, select **Risk Management**. The **Risk Explorer** appears.
3. In the **Risk Explorer**, select one of the following:

View	Description
Data View 	Displays risk data by category in the Risk Explorer. Available categories include Credentials, Financial, Personal, Account Numbers, Credit Cards and Personal Identification Numbers. Custom categories can also be added using the Data Categories Global Object. Enables you to drill down and view vulnerability details by data category. **Note:** When selecting Data View, all network objects that are not contained in the selected data category are filtered out.
Network View 	Displays the relevant SecureSphere component for the chosen level. For example, if you're at the top level, it displays all Sites configured in SecureSphere, and enables you to drill down to all server group in a site, all applications in a server group, and then drill down to details regarding a specific application.

The selected view is displayed.



Note: When viewing the contents of a server group, you can additionally toggle between servers, as represented by **IP addresses**, and services, as represented by available **service names**.

Filtering Data in the Risk Explorer

Filters in the Risk Explorer enable you to configure SecureSphere to display data based on the type of filter object you select. For example, you may only want to display items whose risk is medium and high, you can access the filter and deselect the none and low options, thereby resulting in the desired data being displayed.

To configure a filter:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** appears.
2. In the **Scope Selection** pane of the **Risk Console**, select **Risk Management**. The **Risk Explorer** appears.
3. In the **Risk Explorer**, click the **Filter** dropdown menu. Filter options are displayed, and include the items in the table below.
4. Once you have configured the desired parameter, click outside the filter. The Risk Explorer is updated to reflect the selected filter parameters.

Parameter	Description
Risk Level	Enables you to filter Risk by the following levels: None, Low, Medium and High.
Protection Coverage	Enables you to filter by coverage of SecureSphere gateways using the following Protection Levels **Full**: All IPs that have been discovered have been assigned to gateways for monitoring and protection. **Partial**: Some IPs that have been discovered are not assigned to gateways. **None**: No assets have been assigned to a SecureSphere gateway.

Conducting a Search in the Risk Explorer

The SecureSphere Risk Explorer enables you to search for a specific network or data component by name. This can be helpful when dealing with a large number of items, for example server groups, and you want to display a specific item.



Note: The search is conducted among the currently displayed objects. It does not search within objects, or objects located at a higher level.

To conduct a search:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** appears.
2. In the **Scope Selection** pane of the **Risk Console**, select **Risk Management**. The **Risk Explorer** appears.
3. In the **Risk Explorer**, navigate to the level in which you want to search. For example, if you want to search for a server group, single click the site that contains the server groups you want to search. The selected items are displayed.
4. In the **Search** pane, located in the upper right hand corner of the **Risk Explorer**, type the item you want to search for.
5. Press **Enter** or click the magnifying glass. Results are displayed.

Analyzing Risk Details

The Risk Details window displays details regarding the selected object and assists you in understanding the number of vulnerabilities contained by the selected object. The following information can help you in understanding how to work with risk details:

Understanding the Risk Details Window 148

Risk Details and Vulnerability Management Workflow 149

Understanding the Risk Details Window

The Risk Details window displays a variety of information regarding open vulnerabilities. Easy to read charts include information that includes vulnerabilities broken down by Open, Closed and Mitigated.

To drill down on risk details:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** appears.
2. In the **Scope Selection** pane of the **Risk Console**, select **Risk Management**. The **Risk Explorer** appears.

3. From the desired object in the **Risk Explorer**, click the Information icon. The **Risk Details** window is displayed.

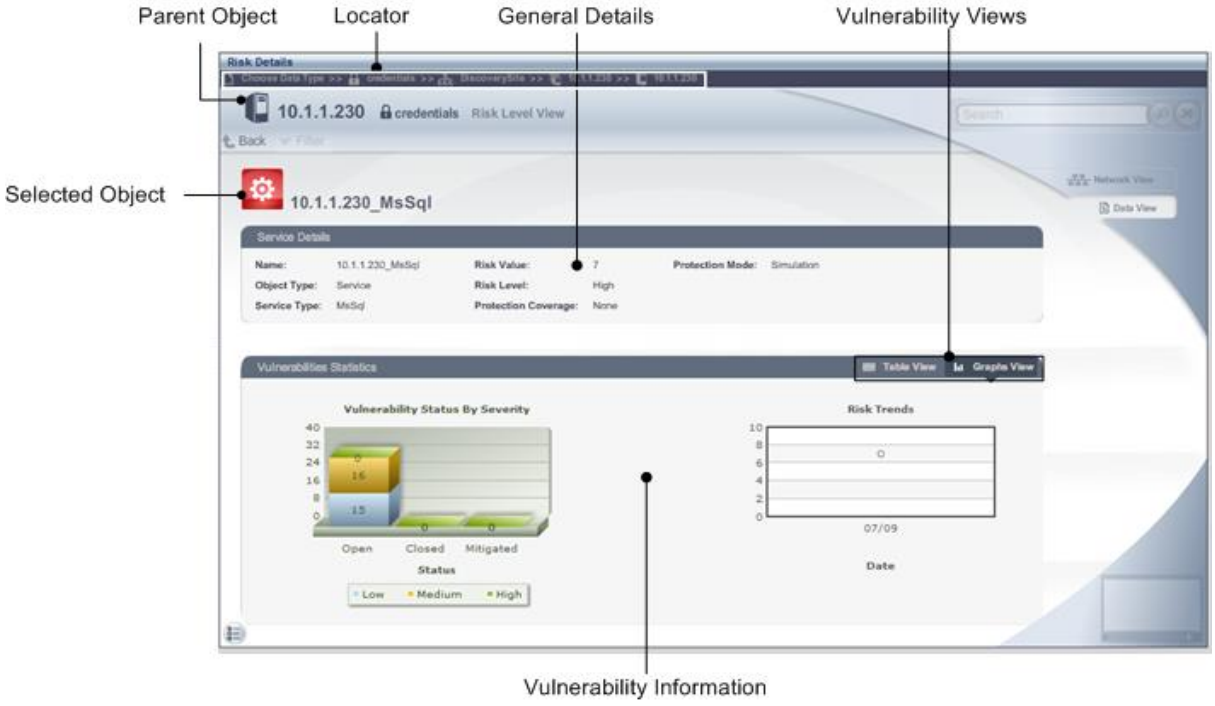


Figure 20: Risk Explorer Details

The Risk Details window provides a number of different indicators to help you in analyzing and understanding various aspects regarding the selected object. These include:

- **Parent Object**: The parent of the object currently displayed. For example, in the diagram above, the object name is 10.1.2.30_MySQL, and its parent Server 10.1.1.230.
- **Selected object**: The object whose details are currently being displayed.
- **Path**: Shows you the path to the current location. Each parent item in the SecureSphere hierarchy is displayed. You can also navigate by clicking the specific level you want to navigate to.
- **General Details**: Lists a variety of important details regarding the item including name, type of item, risk value and level.
- **Vulnerability Information**: Displays breakdowns of vulnerability status in graphic (default) or tabular format.

Risk Details and Vulnerability Management Workflow

Once you have reviewed high-level risk in the Risk Explorer and drilled down to display details, you can conduct the following tasks to review and manage vulnerabilities that may be open.

1. Drill down to the desire object as described in Drilling Down and Navigating in the Risk Explorer on page 146.
2. Review the Risk Level and Risk Value to determine what level of risk is posed to the selected item.
3. Review Vulnerability Statistics to determine the number of open vulnerabilities and to most common vulnerability types encountered.
4. Go to **Table** view to see details regarding these vulnerabilities.
5. With the desired data being displayed; from the Scope Selection pane, select Vulnerability Management, then under views, select Workbench. The Vulnerability Workbench appears displaying the vulnerabilities that represented in the Risk Explorer. For further information on managing vulnerabilities, see Managing Vulnerabilities on page 150.



Note: When navigating between the various Scope Selection options, information is always displayed regarding the specific data selected in the Risk Explorer. This means that once you drill down on data in the Risk Explorer, you can move to the Vulnerabilities window and the vulnerabilities for the selected data is automatically displayed.

6. Manage high-risk vulnerabilities as described in Managing Vulnerabilities on page 150.

Managing Vulnerabilities

After assessments have been run and vulnerabilities have been detected, you can manage them in the Vulnerability Management workbench. The Vulnerability Management workbench enables you to view details regarding detected vulnerabilities and set properties regarding these vulnerabilities such as status and priority. This section reviews the various aspects regarding managing vulnerabilities in SecureSphere including:

Analyzing Discovered Vulnerabilities 150
Vulnerability Management Workflow 151
Introduction to Common Vulnerabilities and Exposures 152
Viewing Discovered Vulnerabilities in Tabular View 153
Viewing Discovered Vulnerabilities in Graphical Views 154
Analyzing Vulnerability Details 156
Managing and Mitigating Vulnerabilities 159
Unmitigating a Blocked Vulnerability 164

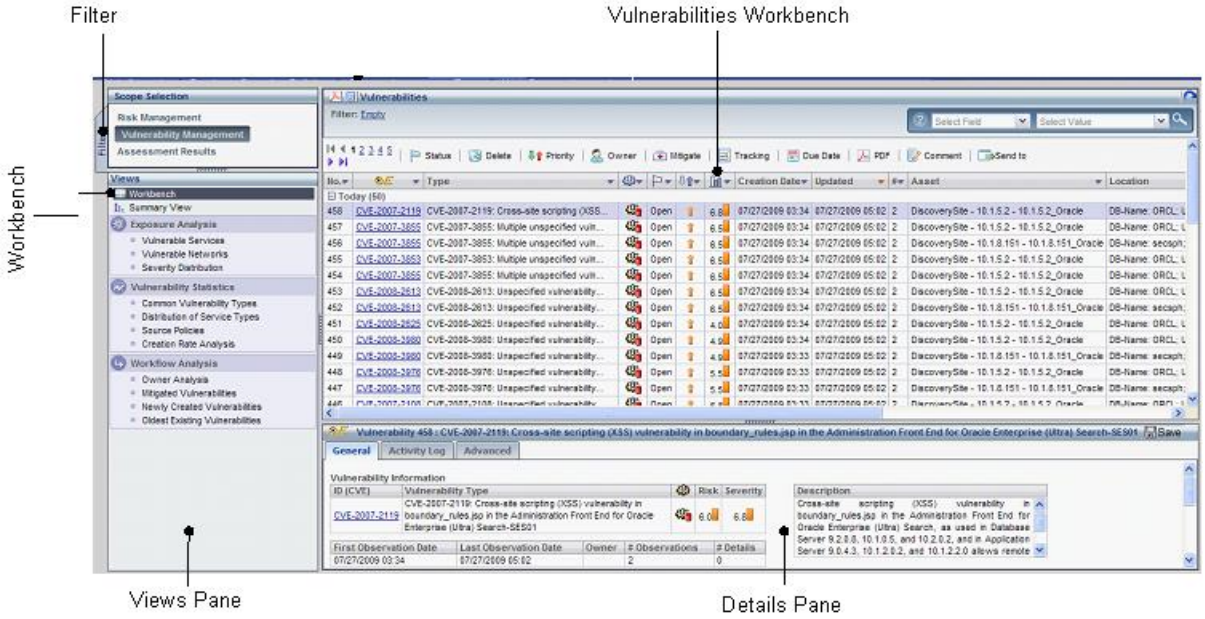
Analyzing Discovered Vulnerabilities

Analyzing discovered vulnerabilities involves viewing the results of assessments using the various vulnerability views, then taking action as required.

To view vulnerabilities discovered during assessment tests:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.

2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**. The **Vulnerability Management** window appears. The Vulnerabilities Workbench is displayed by default.



The Vulnerabilities workbench displays vulnerabilities discovered during assessment and their details in tabular format by default, and enables you to manage the discovered vulnerabilities. It additionally offers a number of graphical views with charts that assist in visually analyzing discovery results.

SecureSphere offers two primary ways of viewing discovered vulnerabilities:

- Viewing Discovered Vulnerabilities in Tabular View on page 153
- Viewing Discovered Vulnerabilities in Graphical Views on page 154

Vulnerability Management Workflow

Vulnerabilities are weaknesses that may leave your assets vulnerable to malicious attack or theft. Vulnerability details provide you an overview of information regarding each vulnerability.

The following table lists items that are used to manage vulnerabilities in the Vulnerability Workbench.

Managing Vulnerabilities - Task Overview

	Action	Description	For more information, see...

	Action	Description	For more information, see...
1	**Visually Scan Vulnerabilities**	Visually scan for vulnerabilities with high severity in the Vulnerabilities table.	Viewing Discovered Vulnerabilities in Tabular View on page 153
2	**Analyze Individual Vulnerabilities**	Examine vulnerability descriptions and other details and try to understand what sort of a threat the vulnerability poses.	Analyzing Vulnerability Details on page 156
3	**Set a Priority for the Vulnerability**	Set the vulnerability's priority to determine how urgent it is.	Setting Vulnerability Priority on page 160
4	**Assign Vulnerability to a User**	Assign the vulnerability to a user who is responsible for its management.	Assigning a Vulnerability an Owner on page 161
5	**Assign a Due Date to the Vulnerability**	Assign a due date for the vulnerability to have been manually mitigated.	Configuring Vulnerability Due Date on page 162
6	**Send the Vulnerability (Optional)**	Optionally send the vulnerability using an action interface.	Sending a Vulnerability on page 164

Introduction to Common Vulnerabilities and Exposures

Imperva Vulnerability Management integrates the use of Common Vulnerabilities and Exposures (CVE), an international organization that maintains an up-to-date dictionary of publicly known information security vulnerabilities and exposures. The use of CVE in SecureSphere provides you with an industry standard classification of the latest vulnerabilities regularly updated, a description of these detected vulnerabilities, and the knowledge required to mitigate them.



Note: For information about how to work with the CVE knowledge base, see Working with the CVE Knowledge Base on page 152.

According to CVE:

- An information security "**vulnerability**" is a mistake in software that can be directly used by a hacker to gain access to a system or network. CVE considers a mistake a vulnerability if it allows an attacker to use it to violate a reasonable security policy for that system (this excludes excluding entirely "open" security policies in which all users are trusted, or where there is no consideration of risk to the system).
- An information security "**exposure**" is a system configuration issue or a mistake in software that allows access to information or capabilities that can be used by a hacker as a stepping-stone into a system or network. CVE considers a configuration issue or a mistake an exposure if it does not directly allow compromise but could be an important component of a successful attack, and is a violation of a reasonable security policy.

For more information regarding CVE, visit their website at `http://cve.mitre.org/`.

Working with the CVE Knowledge Base

When Vulnerabilities are reported, they are displayed in SecureSphere with a link that enables you to view detailed information regarding the vulnerability. This information can be accessed from both the Tabular view one line summary of a vulnerability and from the details of a specific vulnerability.

Viewing Discovered Vulnerabilities in Tabular View

The vulnerabilities table provides a summary of information regarding vulnerabilities detected by SecureSphere. The vulnerabilities table is customizable and can display a range of basic information.

To analyze vulnerability details

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**, then click Workbench. The **Vulnerability Management** window appears.

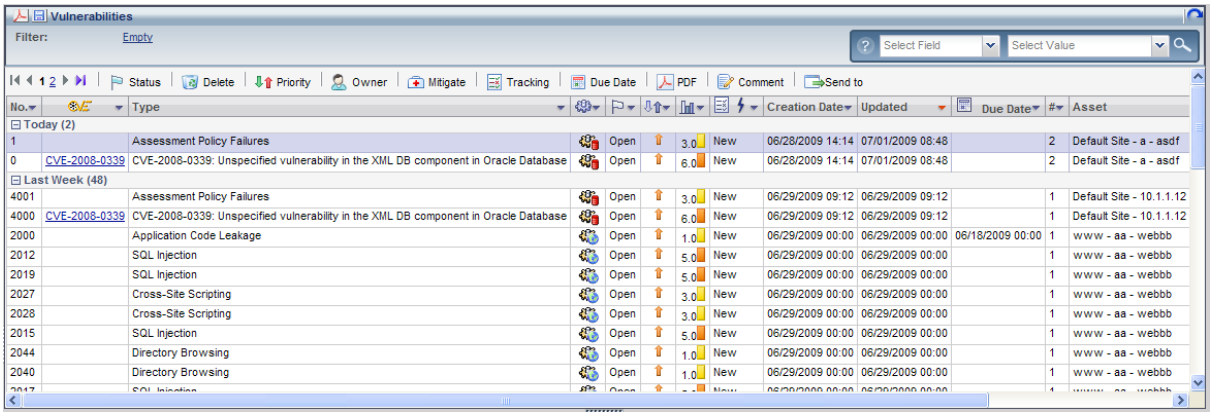


Figure 21: Vulnerability Workbench

The following table reviews columns available in the Vulnerabilities Workbench.

Note: To sort columns in a table, click  on the desired column heading. The arrow direction indicates whether the sort order is ascending or descending based on the selected column.

Vulnerabilities Table Contents

Column Heading	Description
No.	A sequential number assigned to vulnerabilities as they are discovered.
CVE	ID number as determined by the Common Vulnerabilities and Exposures (CVE) Standard. This item is a hyperlink. To view a description of the CVE, click the link.
Type	Name of the vulnerability type associated with the CVE ID.
Service Type	Type of service on which the vulnerability is located.
Status	Status of the vulnerability. Automatically assigned as opened until otherwise modified. Possible statuses include Open, Closed, or Excluded.
Priority	Priority of the vulnerability. Manually assigned by clicking the priority button in the vulnerability toolbar. Possible priorities include Low, Medium and High. Default: Medium.
Severity	Severity of the vulnerability. Automatically assigned by SecureSphere using CVE scoring. Possible severities include Low, Medium and High.
Tracking	Whether the vulnerability is new or assigned, been accepted or rejected.
Creation Date	Date the vulnerability was initially detected.
Updated	Date some aspect of the vulnerability was last updated, for example status or tracking.
Due Date	Date vulnerability management is due to be completed.
#	Number of times this vulnerability was observed. Identical vulnerabilities in SecureSphere are aggregated into a single vulnerability in the Vulnerabilities window. For more information regarding Vulnerability aggregation, see Vulnerability Aggregation on page 154.
Asset	The server on which the vulnerability was discovered.

Vulnerability Aggregation

Vulnerability aggregation is the process in which vulnerabilities of the same service, policy, vulnerability type and key discovered in different assessment scans are aggregated into a single vulnerability in the Vulnerabilities window. This assists with easy management, grouping together identical vulnerabilities that may have taken place in different scans.

Viewing Discovered Vulnerabilities in Graphical Views

In addition to Tabular view, SecureSphere enables you to view discovered vulnerabilities in intuitive, easy to read charts that represents the vulnerabilities discovered in your network. This can assist you in examining vulnerabilities by various properties such as all high severity vulnerabilities, or all vulnerabilities assigned to a specific owner.

To view discovered vulnerabilities in graphical format:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**. The **Vulnerability Management** window appears.

3. From the **Views** pane, select the desired view. For example, Vulnerabilities by Severity. That view is displayed.

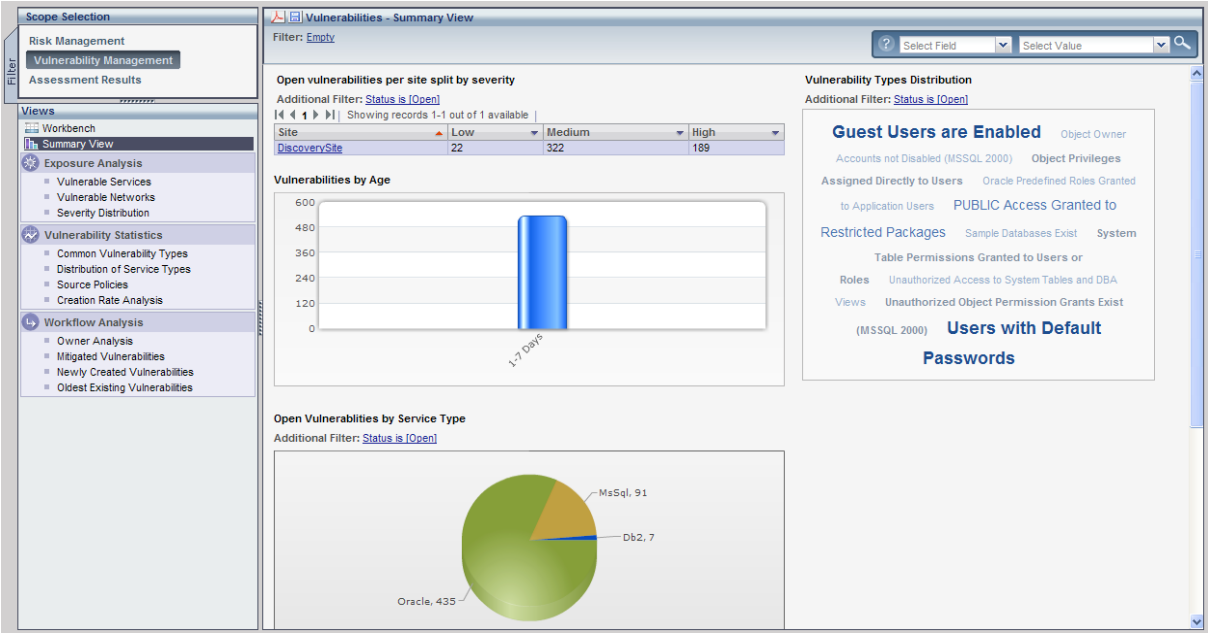


Figure 22: Vulnerabilities - Summary View

SecureSphere has a number of vulnerability views that show data in comprehensible format. You can additionally filter items or conduct actions by clicking on a dynamic link then selecting the desired option.

Views display data in graphic format using tables and charts. The following table list the views that are available.

Vulnerability Views

Name	Description
Summary View	Provides an overview of open vulnerabilities and assists in getting an overall picture of the status of your system. It shows a list of vulnerabilities by age, a tag cloud showing most common vulnerabilities by size of the tag, and open vulnerabilities by service type. Select a category and begin filtering.
Vulnerable Services	Displays various information regarding vulnerable services including by IP, severity, priority and Server Group.
Vulnerable Networks	Displays vulnerabilities in a number of charts by subnet including by status, severity, vulnerability type and service type.
Vulnerabilities by Severity	Displays vulnerabilities in a number of charts by severity including by service type, status and priority.
Common Vulnerability Types	Displays vulnerabilities by the most common vulnerability types in a number of charts including by service type, by status and by service name.
Distribution of Service Types	Displays vulnerabilities by service types in a number of charts including by severity, by status and by vulnerability type.
Source Policies	Displays vulnerabilities by policies in a number of charts including by severity, by vulnerability type and by vulnerability age.
Creation Rate Analysis	Displays vulnerabilities by creation rate in a number of charts including by service type, by severity and by vulnerability type.
Owner Analysis	Displays vulnerabilities by assigned owner in a number of charts including by severity, status, and vulnerability type.
Mitigated Vulnerabilities	Displays mitigated vulnerabilities in a number of charts including by service type, severity, and vulnerability type.
Newly Created Vulnerabilities	Displays newly created vulnerabilities in a number of charts by severity including by status, by service type, by owner and by vulnerability type.
Oldest Existing Vulnerabilities	Displays oldest existing vulnerabilities in a number of charts by severity including by status, by service type, by owner and by vulnerability type.

Analyzing Vulnerability Details

Vulnerabilities that are displayed in the vulnerability workbench need to be analyzed. Once you have an understanding of what the vulnerability entails, you can set a number of properties such as status and priority, assign an owner, mitigate the vulnerability by blocking it or conducting manual mitigation, and more.

The Vulnerabilities Details window comes with a number of tabs that provides information that allows you to understand various aspects of a vulnerability. These include the items in the following table.

Vulnerability Workbench Tabs

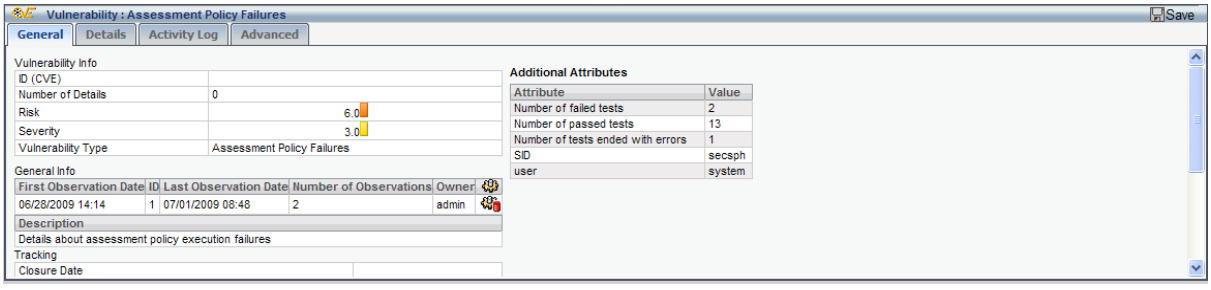
Tab Name	Description
Overview	Contains a variety of information on the vulnerability that can be used to diagnose the vulnerability and subsequently manually mitigate it. For more information see Analyzing General Vulnerability Details on page 157.
Regulations	Lists regulations that are applicable to the test including the name of the regulation, the section number in the regulation, and what specifically was tested to comply with the regulation.
Recommended Fix	Provides a recommendation on how to remedy the vulnerability so it cannot be exploited.
Technical Details	Lists the specific objects in the test that were part of an assessment policy that failed the test and resulted in the vulnerability.
Activity Log	Displays all changes made to a vulnerability including its status, comments that were added, etc. For more information see Viewing the Activity Log on page 159.
Advanced	Shows the number of times this vulnerability has been detected on the asset listed, and enables you to assign keywords to the vulnerability for use with filtering and reporting. Additionally, if a vulnerability has been mitigated displays the name of the mitigation policy.

Analyzing General Vulnerability Details

General Vulnerability details provide a variety of information about a vulnerability that includes basic information such as CVE number, risk and severity level. It additionally displays other attributes such as a number of times a test has passed or failed, and the user name that was involved with the vulnerability. The Description field can be key in understanding how to manually mitigate a vulnerability.

To view general vulnerability details:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**. The **Vulnerability Management** window appears.
3. In the **Views** pane of the **Vulnerability Results** window, click **Workbench**. The workbench appears displaying all vulnerabilities discovered in the system.
4. In the **Vulnerabilities** pane of the workbench, select the vulnerability whose details you want to view. The selected vulnerability's details are displayed in the **Details** pane below.



The Vulnerability Details pane displays a number of informative details regarding the scheduled vulnerability. The key to understanding the vulnerability is found in the description.

Once you've done this, you can assign the vulnerability an owner, configure a due date, and more. The owner is then responsible for taking the required steps to manage the vulnerability, and mitigate it if necessary. The following table lists the various items displayed in the vulnerabilities details.

Vulnerabilities Details

Name	Description
ID (CVE)	Identification number as assigned by the Common Vulnerabilities and Exposures Organization identifiers. For more information, see Working with the CVE Knowledge Base on page 152.
Vuln. Type	The number and name of the Vulnerability type as defined by CVE.
[Service]	The type of service on which the vulnerability has been discovered.
Status	Status of the vulnerability as manually configured in the Vulnerability Workbench.
Tracking	The assignment status configured for the vulnerability. This property is set by a user when selecting a vulnerability in the Vulnerability Workbench and clicking Tracking. For more information, see Setting Vulnerability Tracking Status on page 162.
Priority	The priority assigned to the vulnerability. This property is set by a user when selecting a vulnerability in the Vulnerability Workbench and clicking Priority. For more information see Setting Vulnerability Priority on page 160.
Severity	The severity of the vulnerability. This property is automatically determined as part of ADC content.
#	The number of times this vulnerability was observed as related to the key used to identify the vulnerability.
Creation Date	The date the vulnerability was detected and recorded in SecureSphere.
Updated	The last time a change was made to one of the properties of the vulnerability such as tracking, priority, owner, and due date, or a new vulnerability was added to an existing aggregation.
Asset	The server on which the vulnerability was discovered.
Owner	The owner assigned to the vulnerability. This property is set by a user when selecting a vulnerability in the Vulnerability workbench and clicking Owner. For more information see Assigning a Vulnerability an Owner on page 161.
Due Date	The due date assigned to the vulnerability. This property is set by a user when selecting a vulnerability in the Vulnerability workbench and clicking Due Date. For more information see Configuring Vulnerability Due Date on page 162.
Closure Date	The date the vulnerability was marked as closed. This property is set by a user when selecting a vulnerability in the Vulnerability workbench and selecting the closed option from Status. For more information see Changing Vulnerability Status on page 160.
Description	Description of the vulnerability as provided by Imperva ADC, typically based on information taken from CVE.
Vulnerability Aggregated By	The key used as the basis of the vulnerability aggregation.
Vulnerability Information	Additional information that may be accumulated as part of the assessment.

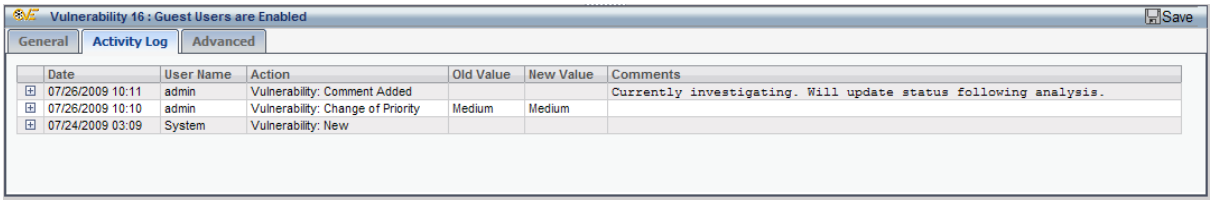
Name	Description
Detail Script Output	Displays the output of details scripts that were run as part of an assessment test. Much like the vulnerability itself, this information is aggregated. So if the vulnerability is an aggregated vulnerability (more than one), this table may display the results of multiple tests.

Viewing the Activity Log

The activity log displays all activity that has been conducted on a vulnerability. It adds a new line for each time an action is taken. For example, if a user configures a due date, this item will be added to the activity log. This provides an audit trail that can be followed to understand what actions have been taken on the vulnerability and at what time.

To view the activity log:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**, then click Workbench. The **Vulnerability Management** window appears.
3. In the **Vulnerabilities** window, select the vulnerability whose activity you want to view. The selected vulnerability's details appear in the **Details** pane at the bottom of the window.
4. In the **Details** pane, click **Activity Log**. The activity log for the selected vulnerability is displayed.



Note: Comments that may have been added when an action was taken may be truncated. To expand the activity entry and see the entire comment, click expand [+] next to the selected entry.

Managing and Mitigating Vulnerabilities

Once vulnerabilities have been detected, they are displayed in the Vulnerabilities Workbench. The vulnerabilities workbench enables you to analyze the various vulnerabilities that have been detected, to set a priority and status, assign an owner, mitigate vulnerabilities and much more.

Note: When working with vulnerabilities in the Vulnerability Workbench, you can select multiple Vulnerabilities and mark them all at once by using **Shift + Click** and **CTRL + Click**.

The following actions can be conducted on vulnerabilities that have been detected in SecureSphere:

Changing Vulnerability Status 160
Setting Vulnerability Priority 160
Assigning a Vulnerability an Owner 161
Marking a Vulnerability as Mitigated 161
Setting Vulnerability Tracking Status 162

Configuring Vulnerability Due Date 162
Generating a PDF Listing Vulnerabilities 163
Adding a Comment to a Vulnerability 163
Sending a Vulnerability 164

Changing Vulnerability Status

When a vulnerability is initially created by SecureSphere it is set to Open. After this point, you can change the vulnerability's status to assist in tracking and determining which vulnerabilities need to be dealt with.

To change the status of a vulnerability:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**. The **Vulnerability Management** window appears.
3. In the **Views** pane of the **Vulnerability Results** window, click **Workbench**. The workbench appears displaying all vulnerabilities discovered in the system.
4. In the **Vulnerabilities** pane of the workbench, select the vulnerability whose status you want to change. The selected vulnerability is highlighted.
5. In the **Vulnerabilities** toolbar, click Status. The **Status** dialog box appears.
6. Select the desired status. Options include:
   - **Open** (default): Sets the status of the vulnerability to open. This is the default status when a vulnerability is created.
   - **Closed**: Sets the status of the vulnerability to closed, meaning the vulnerability has been managed.
   - **Excluded**: Sets the status of the vulnerability to excluded. Meaning the next time an assessment is run, it will not report the vulnerability to SecureSphere.
7. Type a comment if desired.
8. Click **OK**. The priority has been set and is updated in the **Vulnerabilities** pane.

Setting Vulnerability Priority

This procedure describes how to manually assign priority to an open vulnerability. This assists in managing vulnerabilities, determining the order in which they should be managed, and verifying that important vulnerabilities are not left unattended.

To set priority of a vulnerability:

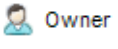
1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**. The **Vulnerability Management** window appears.
3. In the **Views** pane of the **Vulnerability Results** window, click **Workbench**. The workbench appears displaying all vulnerabilities discovered in the system.
4. In the **Vulnerabilities** pane of the workbench, select the vulnerability whose priority you want to set. The selected vulnerability is highlighted.
5. In the **Vulnerabilities** pane toolbar, click Priority. The **Priority** dialog box opens.
6. Select the desired priority. Options include:
   - **High**: Vulnerability is high risk and should be addressed and managed immediately.
   - **Medium**: Vulnerability is of indeterminate risk, requires further examination to understand the risk it poses to the system, and should be addressed as soon as possible.
   - **Low**: Vulnerability is low risk and may not require additional action.

7. Type a comment if desired.
8. Click **OK**. The priority has been set and is updated in the Vulnerabilities pane.

Assigning a Vulnerability an Owner

This procedure describes how to assign an owner to vulnerabilities. This assists in delegating vulnerabilities for review and mitigation.

To assign an owner to a vulnerability:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**. The **Vulnerability Management** window appears.
3. In the **Views** pane of the **Vulnerability Results** window, click **Workbench**. The workbench appears displaying all vulnerabilities discovered in the system.
4. In the **Vulnerabilities** pane of the workbench, select the vulnerability you want to assign an owner. The selected vulnerability is highlighted.
5. In the **Vulnerabilities** pane toolbar, click 

. The **Owner** dialog box opens.
6. In the Owner dialog box, select a desired user. The list of available users are taken from users configured in SecureSphere.
7. Enter a comment if desired.
8. Click **OK**. The owner has been assigned and is updated in the **Vulnerabilities** pane.

Marking a Vulnerability as Mitigated

This procedure describes how to mark a vulnerability as mitigated. You have two options for mitigating:

- **Block**: Mitigates vulnerability by blocking communication.
- **None**: Marks the vulnerability but takes no concrete action.

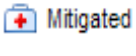
As a general note, not all vulnerabilities require mitigation. It is up to you to determine if mitigative action needs to be taken, as gauged from the description of the vulnerability, and what kind of mitigative action is required. Mitigative action can be assigning an immediate action of Blocked, or taking manual action by, for example, applying a missing service pack or patch.

Manually Mitigating Vulnerabilities: In order to understand what needs to be done to manually mitigate a vulnerability, you need to review the vulnerability **Description**, located in the Vulnerability **General** tab. You should use the information in this location to determine the required course of action for mitigating these vulnerabilities.



Note: Mitigating a vulnerability means creating a policy which then takes action such as blocking the vulnerability. It is only possible to block vulnerabilities in a deployment which includes gateways. The DAS product does not include a gateway, so it is not possible to mitigate vulnerabilities in a DAS-only deployment.

To mark a vulnerability as mitigated:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**. The **Vulnerability Management** window appears.
3. In the **Views** pane of the **Vulnerability Results** window, click **Workbench**. The workbench appears displaying all vulnerabilities discovered in the system.
4. In the **Vulnerabilities** pane of the workbench, select the vulnerability you want to mark as mitigated. The selected vulnerability is highlighted.
5. In the **Vulnerabilities** pane toolbar, click 

. The **Mitigate** dialog box opens.
6. Select the desired option, as follows:

- **SecureSphere Immediate Action**: Allows you to have SecureSphere mitigate the vulnerability by blocking communication. Blocking the vulnerability creates a security policy which blocks the connection. In order to unblock the connection this policy must be manually modified. For instructions on unblocking the connection, see Unmitigating a Blocked Vulnerability on page 164.



Note: Only vulnerabilities marked as (Public Access) in the file name can blocked. This attribute can be seen in the name of the assessment test.

- **Manual**: Indicates that the vulnerability has been manually mitigated. For example, you have applied a patch update. When selecting this option always add a comment indicating the way the vulnerability was manually mitigated.

7. Click **OK**. The mitigation status has been set and is updated in the **Vulnerabilities** pane.

Setting Vulnerability Tracking Status

This procedure describes how to set a vulnerability tracking status. This assists you in the day-to-day management of open vulnerabilities, informing you if the vulnerability has been assigned.

To set the tracking status of a vulnerability:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**. The **Vulnerability Management** window appears.
3. In the **Views** pane of the **Vulnerability Results** window, click **Workbench**. The workbench appears displaying all vulnerabilities discovered in the system.
4. In the **Vulnerabilities** pane of the workbench, select the vulnerability whose tracking status you want to set. The selected vulnerability is highlighted.
5. In the **Vulnerabilities** pane toolbar, click  Tracking . The **Tracking** dialog box opens.
6. Select the desired tracking option, as follows:
   - **New** (default): Sets the tracking status of the vulnerability to new. This is the default tracking status when a vulnerability is created.
   - **In Progress**: Configures the vulnerability as in progress, meaning the vulnerability is being manually investigated and managed by an owner.
   - **Accepted**: Marks the vulnerability as accepted.
   - **Rejected**: Marks the vulnerability as rejected.
7. Type a comment if desired.
8. Click **OK**. The tracking status has been set and is updated in the **Vulnerabilities** pane.

Configuring Vulnerability Due Date

This procedure describes how to assign a due date for a vulnerability. Assigning a due date enables you to assign a date by when the vulnerability should be closed. This date then appears in the vulnerability details.

To assign a due date to a vulnerability:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**. The **Vulnerability Management** window appears.
3. In the **Views** pane of the **Vulnerability Results** window, click **Workbench**. The workbench appears displaying all vulnerabilities discovered in the system.
4. In the **Vulnerabilities** pane of the workbench, select the vulnerability whose due date you want to assign. The selected vulnerability is highlighted.

5. In the **Vulnerabilities** pane toolbar, click **Due Date**. The **Due Date** dialog box opens. You have two options
   - Type a date. Date format is: **MM/DD/YYYY.**
   - Click the calendar button and select a date from the calendar.
6. Type a comment if desired.
7. Click **OK**. The the vulnerability has been assigned a due date.

Generating a PDF Listing Vulnerabilities

This procedure describes how to generate a PDF including details of a specific vulnerability. This PDF can then be distributed as desired.

To generate a PDF with vulnerability information:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**. The **Vulnerability Management** window appears.
3. In the **Views** pane of the **Vulnerability Results** window, click **Workbench**. The workbench appears displaying all vulnerabilities discovered in the system.
4. In the **Vulnerabilities** pane of the workbench, select the vulnerability you want to generate in PDF format. The selected vulnerability is highlighted.
5. In the **Vulnerabilities** pane toolbar, click **PDF**. SecureSphere begins generating the PDF and displays a status bar.
6. When generation is complete, click **Download Report**. The PDF report opens. You can now save the PDF, e-mail it to other users, etc.



Notes:

- You can save vulnerability as a report by clicking the Save as Report icon in the top left-hand corner of the vulnerabilities workbench.
- You can save all vulnerabilities in the Vulnerability workbench in a single PDF by clicking the PDF icon in the top left-hand corner of the vulnerabilities workbench.

-

Adding a Comment to a Vulnerability

This procedure describes how to add a comment to a vulnerability. In addition to adding a comment for each action you take on a vulnerability, you can add a comment that is disconnected from any action. This can assist you in providing general information to other users for any reason.

To add a comment to a vulnerability:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**. The **Vulnerability Management** window appears.
3. In the **Views** pane of the **Vulnerability Results** window, click **Workbench**. The workbench appears displaying all vulnerabilities discovered in the system.
4. In the **Vulnerabilities** pane of the workbench, select the vulnerability you want to add a comment to. The selected vulnerability is highlighted.
5. In the **Vulnerabilities** pane toolbar, click  Comment. The **Add Comment** dialog box appears.
6. Type a comment.
7. Click **OK**. The comment is added to the vulnerability. It can be viewed in the Activity Log. For information on viewing the Activity Log, see Viewing the Activity Log on page 159.

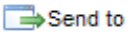
Sending a Vulnerability

This procedure describes how to send a vulnerability to any system configured as part of an action interface. This includes by e-mail, as a syslog message, and more. In order to send a vulnerability you must have a pre configured Vulnerabilities based action interface by which to send it. Supported action interfaces for sending vulnerabilities include e-mail, syslog, and remedy. For more information see Working with Action Sets and Followed Actions on page 326.



Note: A number of vulnerabilities can be configured with a Sent to action at once by using CTRL + Shift to select the vulnerabilities you want to send. This sends a separate item for each vulnerability. So if, for example, you are using a send e-mail action interface, three separate e-mails would be sent.

To send a vulnerability:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**. The **Vulnerability Management** window appears.
3. In the **Views** pane of the **Vulnerability Results** window, click **Workbench**. The workbench appears displaying all vulnerabilities discovered in the system.
4. In the **Vulnerabilities** pane of the workbench, select the vulnerability you want to send. The selected vulnerability is highlighted.
5. In the **Vulnerabilities** pane toolbar, click 

. The **Send to** dialog box appears.
6. Select the desired action interface to use for sending the vulnerability.
7. Type a comment if desired.
8. Click **OK**. The vulnerability is sent through the selected action interface.

Unmitigating a Blocked Vulnerability

When choosing to mitigate a vulnerability by using the Block option, SecureSphere automatically creates a policy which determine how to identify potentially risky traffic and block. In order to unblock traffic that had been blocked using mitigation, you need to find the policy that was created and delete it.

To unmitigate a vulnerability:

1. In the **Main** workspace, click **Risk Management > Risk Console**. The **Risk Console** is displayed.
2. In the **Scope Selection** pane of the **Risk Console**, select **Vulnerability Management**. The **Vulnerability Management** window appears.
3. In the **Views** pane of the **Vulnerability Results** window, click **Workbench**. The workbench appears displaying all vulnerabilities discovered in the system.
4. In the **Vulnerabilities** pane of the workbench, select the vulnerability you want to unmitigate. The selected vulnerability is highlighted.
5. In the **Details** pane below, click the **Advanced Tab**,
6. Click on the mitigated policy, you are taken to the policy in the Policies window.
7. From the **Policies** pane in the **Policies**, click  **Delete**. You are asked if you want to delete the policy.
8. Click **Yes**. The policy is deleted and the vulnerability is no longer mitigated (blocked).



Note: You can also unblock traffic mitigated by a mitigation policy removing the policy from the site tree, under the relevant service's applied policies tab.

CHAPTER 7

Security Policies

This section describes how to configure and use SecureSphere security policies.

SecureSphere comes with a set of predefined policies that protect against known attacks and threats. You can edit the default SecureSphere policies and configure custom policies.

This section includes the following topics:

Introducing Security Policies 166
Security Policy Main Tasks 175
Network Security Policies 184
Securing Web Transport 189
HTTP Protocol Signature Policies 197
Web Service Custom Policies 198
Web Service Correlated Validation Policy 203
Securing Web Applications 206

Introducing Security Policies

A **SecureSphere security policy** is a set of definitions that characterize security violations and actions that SecureSphere must take in response to them.

SecureSphere contains predefined policies that provide protection against the majority of known attacks and threats. You can use the default policies, modify them or create user-defined policies. The following subjects provide information regarding security policies:

Providing Multi Level Protection Using Security Policies 167
Security Policy Types 167

Correlated Attack Validation (CAV) Used in Correlation Policies 170
Intrusion Prevention Used in Signature Policies 170
Security Policy Structure 171
Directly Configuring Policies from the Security Policies Window 171
Using Default Security Policies 172
Security Policy Exceptions 173
Enabling Response Page in Violations 175

Providing Multi Level Protection Using Security Policies

To ensure complete application security, SecureSphere policies provide your system with multi level protection that correlates with the SecureSphere object hierarchy, as follows:

Server Groups > Services > Applications

Policies are enabled by applying them to objects in SecureSphere hierarchy. The security engine includes several security layers called **security policy types**, mostly corresponding to the OSI model. At each level, the security engine can block traffic immediately or continue monitoring specific IPs, application users and sessions for later blocking if necessary. Each policy type is bound to a specific object level, which also illustrates how **security policy types** are correlated with the SecureSphere object levels, as follows:

- The lower level of the network security layer prevents network based attacks and is bound to the Server Group level.
- The protocol and basic applicative functionality layer, such as protocol compliance and application attacks signatures, is bound to the Service level.
- The higher protection layers, such as application profile and correlated attack validations, are bound to the Application level.



Note: For an overview of security policy types, see Security Policy Types on page 167.

The figure below shows how policy types are bound to the object levels.

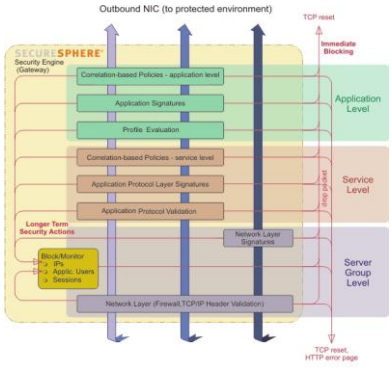


Figure 23: Security Policies and SecureSphere Object Levels

Security Policy Types

The policies that are applied to the object are divided into the 2 following groups:

- **Basic Security Policy**: Enables defining a single policy of each policy type.
- **Additional Security Policy**: Enables defining multiple policies of the same type or different types and applying them to the same object.

The following tables provide a short summary of the policy types.

Server Group Level Network Policy Types

Type	Description
Firewall Policy	Defines the services is used to access the protected servers, see Firewall Policy on page 184.
Network Protocol Validation	Validates a proper use of the TCP/IP protocol according to the RFC standard, preventing attacks that are based on TCP/IP protocol vulnerabilities, see Network Protocol Validation on page 186.
Stream Signature	Identifies well known attacks for generic services, see Stream Signature on page 188.

Service Level Web Security Policy Types

Type	Description
HTTP Protocol Validation	Validates a proper use of the HTTP protocol according to the RFC standard, see HTTP Protocol Validation Policy on page 189.
Web Service Correlated Validation	Protects against multi-stage web application attacks, see Web Service Correlated Validation Policy on page 203.
Cookie Signing Validation	When SecureSphere operates in the Kernel Reverse Proxy mode, this policy enables cookie signing and verification, see Cookie Signing Validation Policy on page 195.
HTTP Protocol Signatures	Protects against known protocol attacks using signatures, see HTTP Protocol Signature Policies on page 197.
Web Service Custom	Enables creating user-defined web service level policies based on various combinations of the match criteria, see Web Service Custom Policies on page 198.

Application Level Web Security Policy Types

Type	Description
Anti Scraping	Identified scraping events in which someone is attempting to "scrape" data from web pages. For more information, see Anti Scraping Policy on page 208.
Bot Mitigation	Identifies bot attacks meant to cause Denial of Service and other automated attacks. For more information see Bot Mitigation Policy on page 207.
Fraud Prevention Services	Works together with third party vendors to determine if devices connecting to the network are compromised. Then determine policies for handling problematic traffic. For more information see ThreatRadar Fraud Prevention Services on page 206.
Web Profile	Alerts on and prevents the application user behavior that deviates from the Application Profile, as defined in SecureSphere, see Web Profile and Web Profile Policy on page 208.
Web Worm	Provides protection for zero day web worms based on Imperva unique technology, see Web Worm Policy on page 209.
Web Application Signatures	Protects against known application attacks using signatures, see Web Application Signatures Policy on page 209.
Web Application Custom	Enables creating of user-defined web application level policies based on various combinations of the match criteria, see Web Application Custom Policies on page 210.

Type	Description
OCSP Protocol Validation	Verifies the revocation status of X.509 digital certificates. This assists web infrastructure including servers and Internet browsers in determining if a web site certificate is valid, has been revoked, or if its status is unknown. For more information, see OCSP Protocol Validation Policies on page 212.

Correlated Attack Validation (CAV) Used in Correlation Policies

Imperva's correlation policies are based on the unique Correlated Attack Validation (CAV) technology, which addresses complex attacks that are ambiguous in nature. Correlated Attack Validation (CAV) examines multiple pieces of information at the network, protocol and application levels over time to distinguish between attacks and valid user traffic. By basing decisions on multiple observations rather than a single event, CAV delivers a highly accurate and completely automated defense system, achieving overall accuracy that cannot be matched by several standalone data security products.

The following policy types are based on the CAV technology:

- Web Service Correlated Validation, see Web Service Correlated Validation Policy on page 203.
- Web Service Custom, see Web Service Custom Policies on page 198.
- Web Application Custom, see Web Application Custom Policies on page 210.

Correlated Attack Validation effectively blocks multi-stage application attacks. These attacks cannot be accurately prevented through simple signature matches because they consist of several events or multiple components within a single request or query.

For example, HTTP request smuggling combines a Web application attack with two redundant HTTP request headers. Vulnerable Web servers and proxy servers receive the request and process the content indicated by the first header. Malicious content in the remainder of the HTTP request can poison a Web proxy cache or include a cross-site scripting or session hijacking attack.

Intrusion Prevention Used in Signature Policies

SecureSphere provides signature protection against known vulnerabilities in commercial infrastructure software such as Apache, IIS, Oracle, and so on, see Working with Signatures on page 307 The content provided by the signature detection mechanism is based on the research done by Imperva's Application Defense Center (ADC) and a combination of other resources such as Snort™, CVE and so on. Using this content, you can quickly isolate the most reliable signature dictionaries for your specific environment. Signatures are continuously and automatically updated on demand or even manually if needed.

The SecureSphere signature database is hosted on the Imperva web site and is generally updated on weekly basis. The SecureSphere Management Server can be configured to automatically connect to the Imperva web site and download the latest updates. If your SecureSphere management server has no Internet connection, you can perform this process manually by downloading the latest signature database from the Imperva web site and then uploading it to the SecureSphere management server using the GUI.

To simplify the use of the signature database, SecureSphere includes the concept of Signatures Dictionaries. A dictionary is a collection of signatures generated by applying a filter on the SecureSphere signature database. SecureSphere instantly and automatically generates the dictionary for you. You can define as many dictionaries as you like. When a new signature is added to the SecureSphere signature database, it is automatically added to all relevant dictionaries.

SecureSphere comes with a predefined set of dictionaries. These dictionaries were defined by the Imperva ADC, providing most customers with a quick start on the product without having to define dictionaries.

You can select whether or not to use each dictionary with each one of the protected server groups/services/applications. For example, you can select to use a specific dictionary only with specific server groups. When a certain dictionary is selected for a specific domain level, SecureSphere detects the signatures in the dictionary if they appear in a communication to the protected level.

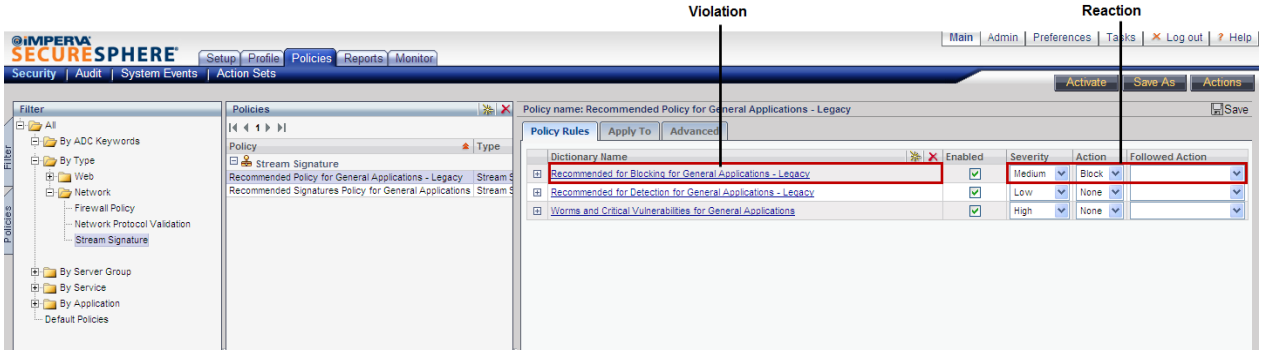
The following policy types are based on signature protection:

- HTTP Protocol Signatures, see HTTP Protocol Signature Policies on page 197.
- Web Application Signatures, see Web Application Signatures Policy on page 209.

Security Policy Structure

Most of the security policies consist of **rules** that are building blocks of that policy. Each rule includes 2 main parts, a description of the **violation** against which the rule protects and a definition of the **reaction** to that violation. By that way, a security policy is a set of rules that are grouped together based on security feature that they represent.

The figure below shows the Policies window with the Rules pane.



Directly Configuring Policies from the Security Policies Window

SecureSphere enables you to configure a number of parameters regarding security policies directly from the main security policy window by selecting one or more policies and using right-click functionality. This eliminates the need to go into a policy and edit its individual configuration, and enables you to select a number of policies and configure them with a specific parameter all at the same item.

To directly configure policies from the Security Policies window:

1. In the **Main** workspace, select **Policies > Security Policies**. The **Security Policies** window appears.
2. Configure a filter to display only those policies that are relevant. For more information on filtering policies, see Filtering Policies on page 181.
3. Select each of the policies you want to apply the filter to by clicking on them once.
   - To select a series of policies, click the first policy in the series, then shift + click the last policy of the series.
   - To select a number of policies not in a series, ctrl + click each policy you want to select.
4. Once you have selected all of the desired policies, right-click on one of the selected policies the right-click shortcut menu opens. You have the options in the table below.
5. Click the desired action. The selected policies are configured and the changes are saved.

Option	Description
Set Followed Action	Directly applies an existing followed action to the selected policies. For more information see Directly Assigning Followed Actions from the Security Policy Window on page 341.
Set Rules Action	Sets rules of the policy to **Block** traffic when policies are violated, or alternatively set the rules action to **None**.
Set Rules Severity	Sets the severity of the rules of the selected policies. Options include Severity No Alert, Informative, Low, Medium and High. **Note:** When setting the severity to No Alert, followed actions are removed.
Enable Rules	Enables or disables ALL rules in the selected policies.
Display Web Response Page	Determines whether or not to include the web response page shown to users in alerts for the selected policies. For more information, see Enabling Response Page in Violations on page 175. **Note:** This option is only available with web based policies.

Using Default Security Policies

SecureSphere contains default policies that you can adjust to your security needs.



Note: Policies are defined globally in SecureSphere, so changing policy definitions for one object affects all the other objects that this policy is applied to. In case you need to change a policy, consider cloning the policy and adjust it specifically as required.

Out of the box, SecureSphere provides you with a set of default policies that are based on best practices. To meet your system requirements you can perform one of the following:

- Adjust the built in policy rules using exceptions, disabling and enabling specific rules, changing the severity, action and followed actions, see Security Policy Exceptions on page 173.
- Clone a policy.
- Create custom policies.

The default policies are automatically applied to any new domain object added to the system. This means that when you start building your site, adding server groups, services and applications, SecureSphere automatically applies default policies to each one of these object levels, providing this object with protection against various security threats. For example, when adding a new server group, the network layer policies are automatically applied to this object.

If you change a default policy, the template of this policy remains the same. So when you are creating a new policy from scratch, this policy is generated directly from the unchanged template. If you are creating a new policy using a modified policy, then the changes that you performed appear in the new policy as well.

You can modify the default and the user defined policies in the following locations in SecureSphere:

- The **Security** window in the Policies workspace: You must be aware of all the aspects of the effect that such change might cause to the objects to which these policies are applied to, see Modifying Policies in the Security Window on page 176.
- The **Applied Policies** tab in the Sites window: You can perform only minor changes to the violated policy and the **Applied To** tab is read only, see Tuning Security Policies in the Setup Window on page 180.
- The **Alerts** window in the **Monitoring** workspace: You can perform only minor changes to the violated policy and the **Applied To** tab is read only, see Modifying Security Policies in the Monitoring Window on page 180.

Security Policy Exceptions

You can define exception for some policies when a specific part of the application does not comply with the security policy in your organization. Using exceptions you can adjust the default SecureSphere policies to meet your requirements and to prevent false positives by specifying exceptions that include conditions under which an alert is not created.

For example, in some cases there are acceptable protocol violations for which you do not want the system to generate an event such as specific URLs to/from which long strings are commonly transmitted, or certain methods that are employed but do not fully comply with the standards.

Exceptions can be defined only for the policies that contain violations that might cause false positives and not for all policies. You can define exceptions in the following ways:

- In the **Violations** window, if you see a violation that is actually a false positive, you can define it as exception in one click, see Taking Action on Alerts and Violations on page 98.
- In the **Security** window you can define exceptions manually for specific security policies.

To define an exception in the Security window:

1. In the **Main** workspace, select **Policies > Security**. The **Security** window appears.
2. In the **Security** window, select the policy for which you want to define an exception and then select **Advanced**. The **Advanced** tab appears.
3. In the **Exceptions** pane, click . A new line appears in the **Exceptions** table.
4. From the drop-down list, select the **rule/signature** for which you want to define the exception.
5. Add a **comment** that explains the logic behind the exception, why the exception was added. Comments can also be added when adding an exception from an alert.
6. Click **Save**. The **Empty** link appears in the **Exception Summary** column.
7. Click **Empty**. The **Edit Exception** window appears.
8. In the **Edit Exception** window, set the exception using match criteria mechanisms and click **Save**. The **Edit Exception** window closes and the summary information appears in the **Exception Summary** column.



Note: The match criteria used to define exceptions is also used in custom policies. For match criteria descriptions, refer to Custom Policies Match Criteria on page 600.

XFF Forwarded Addresses As Source IP Address Exceptions

If, in the policy Exceptions pane, you define a web client's IP address as the **Source IP Address** match criteria as described above, the policy will not ignore connections from that client's IP address if there is a proxy between the client and the SecureSphere Gateway. This is true even if **Identify real client IP according to HTTP forwarding header** in the HTTP service's **Operation** tab is checked.

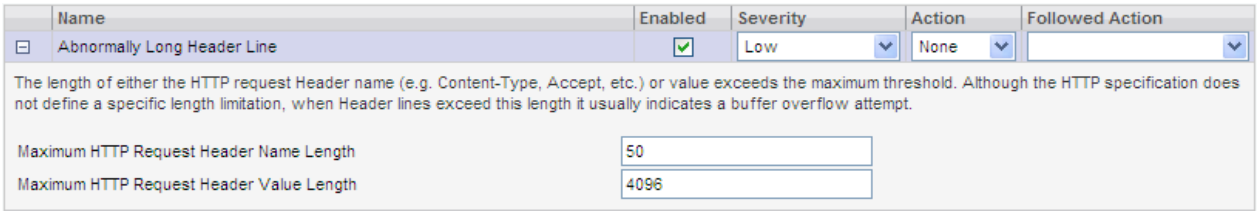
The workaround is as follows:

1. Create a new policy.

   If the policy you want to modify is an ADC policy which cannot be modified, then clone it, that is, create another policy based on (**Use Existing**) the ADC policy.
2. In the new policy, add **Source IP Addresses** to the Match Criteria (**Match Criteria** tab).
3. Set **Operations** to **Exclude All**.
4. Select the IP Group to exclude (ignore) under **IP Groups**, or specify a single IP address under **User Defined**.
5. Click the right pointing arrow to move your selection to the **Selected** box.
6. Save the policy.
7. Apply the policy to the service.

EXAMPLE: A POLICY EXCEPTION EXAMPLE

The Abnormally Long Header Line rule, see the following figure, is included in the Web Protocol Policy rules. If the length of either the HTTP request Header name or value exceeds the maximum threshold, then the policy is violated, as it indicates a buffer overflow attempt.



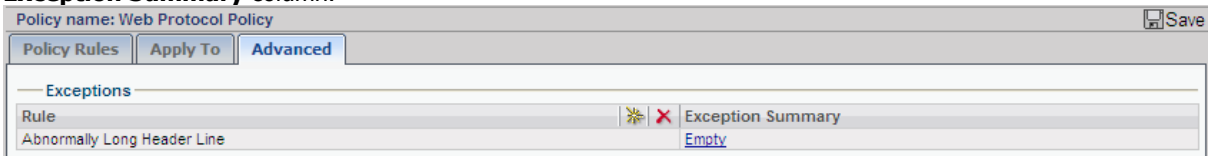
If the violation only occurs in the following specific location in the application: /products/productsline.asp., it is clearly a false positive as the value is correct. This value is just a little bit longer than the others in the application. In this case the options are as follows:

- Enable this rule, meaning that each time it happens, alerts and violations appear in the Monitoring workspace.
- Tune the **HTTP Request Header Value Length** parameter and the **HTTP Request Header Name Length** parameter to accept values that are greater than the default values, which applies across the whole service, meaning that SecureSphere will allow longer or shorter header values and names in all the HTTP requests.
- Disable the policy rule **Abnormally Long Header Line** so that it does not apply any more to the service, meaning that this rule will not be applied to any of the HTTP request headers and the real attack will not be prevented.
- Create an exception directly from the violation or manually in the **Security** window, meaning that only in the request headers from that specific URL SecureSphere will not send alerts and will not block the traffic each time the abnormally long names or values are detected.

Enabling this rule and using it as is will cause a lot of false positives. Tuning the rule to accept greater values or disabling the rule will seriously compromise the security level. The most accurate results, which will prevent the false positives and cause low level security compromise, can be achieved by creating an exception.

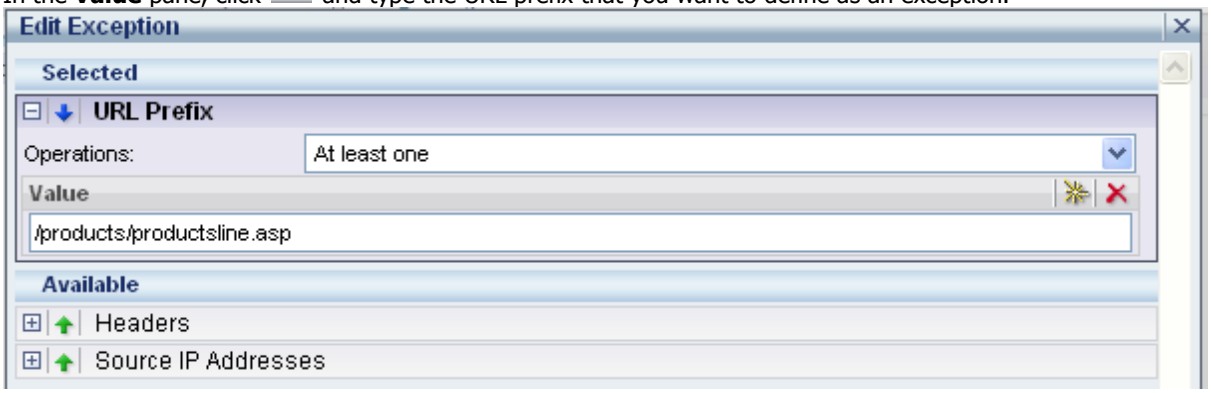
To define an exception to the Abnormally Long Header Line rule using the URL prefix:

1. In the Main workspace, select Policies > Security. The Security window appears.
2. In the **Filter** pane, select **Web > Service Level > HTTP Protocol Validation**. **Web Protocol Policy** appears in the **Policies** pane.
3. In the Policy Rules tab, select **Abnormally Long Header Line** and click **Advanced**. The **Advanced** tab appears.
4. In the Exceptions pane, click . A new line appears in the Exceptions table.
5. From the drop-down list, select **Abnormally Long Header Line** and click Save. The **Empty** link appears in the **Exception Summary** column.

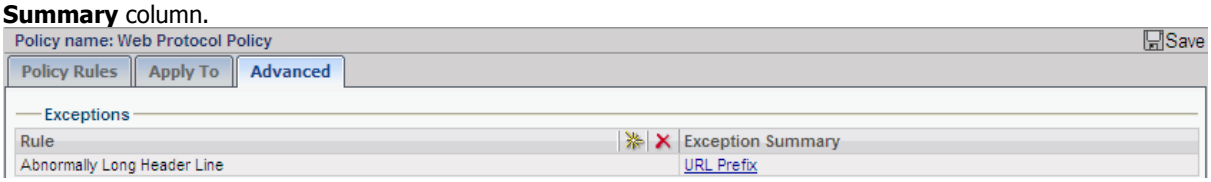


6. Click **Empty**. The **Edit Exception** window appears.
7. In the **Edit Exception** window, click **+** next to the **URL Prefix**. The **URL Prefix** pane appears.
8. From the **Operations** drop-down list, select **At Least One**.

9. In the **Value** pane, click  and type the URL prefix that you want to define as an exception.



10. To apply the URL Prefix exceptions to this rule, click on the green arrow. The URL Prefix is now in the Selected pane.
11. Click **Save**. The **Edit Exception** window closes and the summary information appears in the **Exception Summary** column.



You can fine tune this exception to be based not only on URL, but also on specific header or even source IP.

Enabling Response Page in Violations

The ability to view the response the user received from a web server provides the administrator with additional capabilities such as capturing a response that contains sensitive data or a violation. This enable tracking data leaked from the company. The information is recorded by SecureSphere and it appears on the **Response** page in the **Monitoring** window. For more information, see Managing Blocked Sources on page 105.

To enable the Response Page tab:

1. In the Main workspace, select Policies > Security.
2. In the Security window, select the relevant policy and click Advanced.
3. To enable displaying the **Response Page** tab in the **Violations** window, select Display Response Page in Violations.



Note:

- Response pages can also be configured directly from the policy window by right-clicking the desired web policies, then choosing **Display Web Response Page** > **Yes**. For more information on this functionality, see Directly Configuring Policies from the Security Policies Window on page 171.
- The setting is also applied static files.

Security Policy Main Tasks

The following table presents the overview of tasks performed with security policies.

Security Policy Main Tasks

Task Overview	See
When adding a new object in the Sites window (Main > Setup > Sites), check the default policies that SecureSphere automatically applies to this object and edit them if needed using the Applied Policies tab in the Sites window. This task is recommended for advanced users only.	Tuning Security Policies in the Setup Window on page 180
Review the default policies and adjust them to the security needs of your organization. This task is recommended for advanced users only. The best recommended way of editing policies by unexperienced users: Create a new policy on the basis of an existing one, edit it and then apply it to the objects that you want.	Modifying Policies in the Security Window on page 176
Create user-defined security policies. This task is recommended for advanced users only.	Adding New Security Policies on page 179
Applying policies: Policies are activated only when they are applied to the objects that you want to protect.	Applying Policies on page 318
Modifying policies in the Monitoring window.	Modifying Security Policies in the Monitoring Window on page 180

Modifying Policies in the Security Window

You can modify policies at any time. You can modify some default and user-defined policies. For more details about the default policies see Using Default Security Policies on page 172.

Each policy contains a set of rules, so modifying the policy means enabling/disabling/adding/removing/cloning/editing rules. You can remove only user-defined rules from the policy. Default rules can be only disabled and not removed.

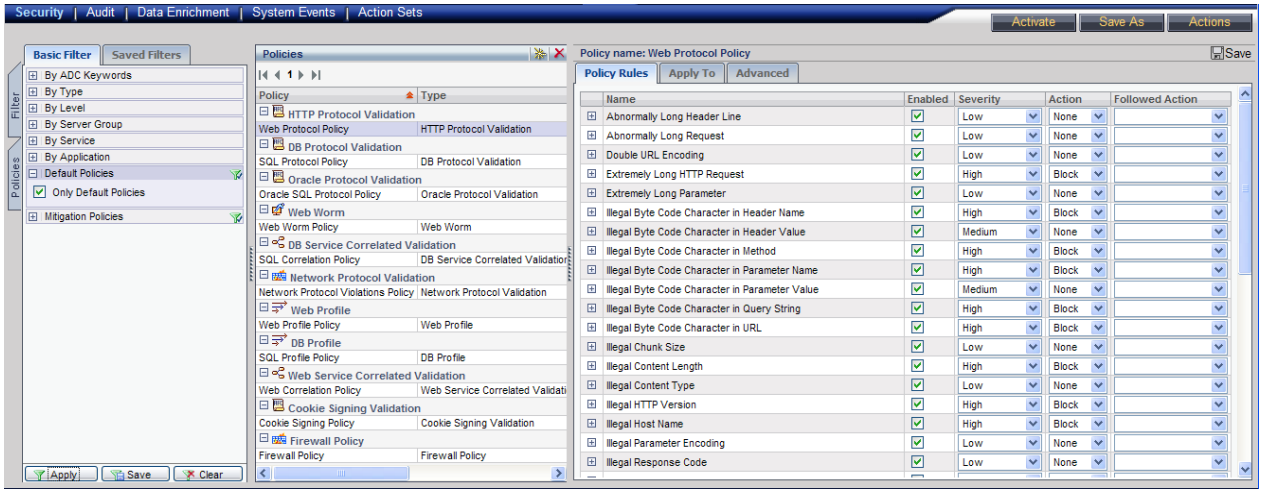


Figure 24: Security Policies Window

The Security window contains the following parts:

- **Filter pane**: Enables you to select the category of the policies that you want to view/modify in the Security window. The filtering categories are defined according to different methods of policy grouping, see Filtering Policies on page 181.
- **Policies pane**: Presents the policies belong to the category or the object selected in the Filter pane. The policies are displayed in this pane according to their types, the type name appears on the top and under the type name appear all the policies that belong to that type, see Security Policy Types on page 167.
- **Policy details pane**: Presents the name of the policy at the top of the pane and enables you to perform the following:
  - View and modify policy rules, see Modifying Security Policies on page 177.
  - Apply policies, see Security Policy Exceptions on page 173.
- Set the Advanced policy parameters, see Security Policy Exceptions on page 173
- Enabling Response Page in Violations and disable the learning process in the profile, see Disabling Automatic Web Profiling on page 278.

Modifying Security Policies

You can modify security policies at any time. The following table presents security policy rule parameters and explains how to tune them.

Security Policy Rule Parameters

Component	Description
Name	The definition of the violation against which this rule protects.
Enabled	Enables the rule, meaning that this rule participates in protection provided by the policy. If **Enabled** is not selected, the rule is not applied.
Severity	The following severity levels can be configured for each rule: • **No Alert**: No alert is generated for the violation by this policy. Select **No Alert** if there is another policy which does alert on this violation, or on some variation of it, and you do not want multiple alerts for the same violation. For example, if you want to issue an alert when this violation occurs 5 times in one hour, define No Alert in the existing policy and define a custom policy with the Violation match criteria. Use this option with care, because if you do not define another policy to issue alerts for this violation, the violation may go undetected. **Note:** For policies with No Alert severity, Followed Action can not be configured, event's information is not reported and it does not participate in the learning process. • **Informative**: Define in case of a very low risk, or when it is not a real attack, but an event was reported and it provides additional information that can be used for tuning purposes. Informative violations are not displayed in the dashboard. • **Low** • **Medium** • **High**
Action	The immediate action that is taken when there is a match on the policy rule: • **None**: SecureSphere takes no action, and only generates violations and alerts, reporting about the security event. • **Block**: SecureSphere blocks the operation that caused this violation and generates violations and alerts, reporting about the security event. **Default**: None.
Followed Action	The action that is taken by SecureSphere in addition to the action defined by the Action parameter. The additional action is taken from the pre-configured list of actions defined in Action Sets, see Working with Action Sets and Followed Actions on page 326. The Followed Action is taken even when the Action is set to None. Example: once the violation is blocked, you want SecureSphere to send email to different people in the organization, notifying them about blocking of this specific violation.
Match Criteria	Select a set of criteria. These are related by a boolean logical AND, so that only if all of their criteria match, the rule is triggered. See Filter Criteria and Report Data Fields on page 574.

To modify policies in the Security window:

1. In the **Main** workspace, select **Policies > Security**. The **Security** window appears.
2. In the **Security** window, select the policy that you want to modify, see Filtering Policies on page 181. The **Policy Rules** tab presents all the rules related to the selected policy.
3. To modify the rules, see the explanations in table Security Policy Rule Parameters above.
4. To modify the Apply To options, see Applying Policies on page 318.
5. To modify the Exception definitions, see Security Policy Exceptions on page 173.
6. To enable the Response page, in the Advanced tab, select Display response page in alerts. For more information about the Response page, see Enabling Response Page in Violations on page 175.

7. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Adding New Security Policies

In addition to the policies provided by Imperva, you can create user-defined security policies. This capability takes advantage of SecureSphere object-based structure that enables you to adjust the policy mechanism to your specific needs and requirements according to the object levels (Server Group, Service, Application).

Policies provided by Imperva cover the majority of typical configurations. Sometimes for some elements in your system you need to provide protection using strict rules and industry standards, while in other cases you might want to define more flexible policies.

Example: You have two main applications in your organization. Application A is well defined and works according to the standard. Application B is an internally developed application that was developed 10 years ago by a programmer who left the company shortly after that. Application B is still widely used by the company employees, but it does not behave exactly according to the industry standard.

To adjust policies to different security needs and protection levels, you can do one of the following:

- To clone the default SecureSphere policies and adjust them to your requirements, see Cloning/Creating New Policies on page 179.
- To create a policy from scratch, see Cloning/Creating New Policies on page 179.
- To add a new custom policy, see Web Service Custom Policies on page 198 or Web Application Custom Policies on page 210

Cloning/Creating New Policies

You can create user-defined policies that are bound to the object levels and are divided into the following policy type categories:

- **Web Service**: Enables cloning/creating policies using the Web Service policy types, see Securing Web Transport on page 189.
- **Web Application**: Enables cloning/creating policies using the Web Application policy types, see Securing Web Applications on page 206.

To create a user-defined policy:

1. In the Main workspace, select Policies > Security. The Security window appears.
2. In the Security window, click  and select the policy type category based on which you want to create the policy from the drop-down list. The **Create New Policy** dialog box appears.
3. In the Name text box, type the user-defined name.



Note: It is recommended to give a meaningful name and description to the new policy, as the name and the description appear in alerts.

4. To create a new policy, select **From scratch** and select the policy type from the **Type** drop-down box.
5. To clone a policy, select the policy that you want to use for cloning from the **Use existing** drop-down list.
6. Click **Create**.
7. The **Create New Policy** dialog box closes and the new policy name appears in the **Policies** pane. Now you can define the policy rules according to the policy type definitions.

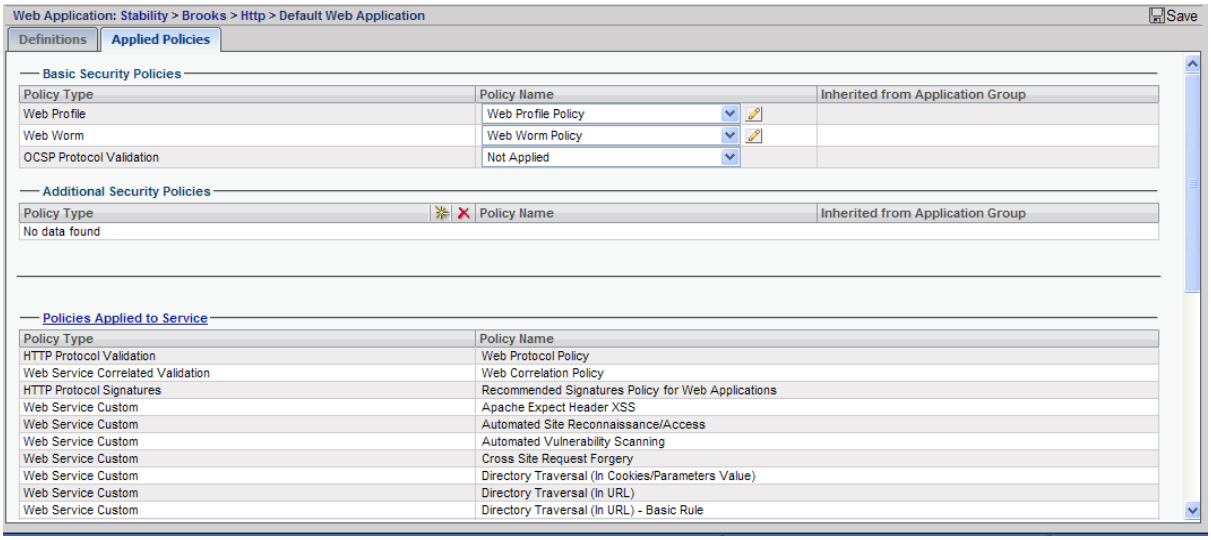
Tuning Security Policies in the Setup Window

You can tune the default policies that are automatically applied to every domain object (server group, service, or application) in the **Setup** window. Each domain level in the site tree is associated with a different set of default policies. For each object the Applied Policies tab displays the policies that are applied to this domain level and the policies that are applied to the levels above the object.

The policies that are applied to the object are divided into the 2 groups, basic policies and advanced policies, see Security Policy Types on page 167.

To tune a policy:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. From the **Sites Tree** pane, select the object to which you want to apply policies, see Creating a New Site on page 50, and click **Applied Policies** in the right pane. The **Applied Policies** tab appears.



3. To edit a policy, click  next to the policy that you want to edit. The **Policy Details** window appears.
4. In the **Policy Details** window, you can modify the following settings:
   - **Policy Rules tab**: Modify policy rules as explained in Security Policy Rule Parameters under table in Modifying Security Policies on page 177.
   - **Applied To tab**: View the read-only presentation of applied policies, applying of policies is done in the Security window, see Applying Policies on page 318.
   - **Advanced tab**: Add/edit policy exceptions. For more information, see Security Policy Exceptions on page 173, Enabling Response Page in Violations on page 175 and disable the learning process in the profile, see Disabling Automatic Web Profiling on page 278.
5. In the **Policy Details** window, click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, Activating Settings on page 75. The changes affect all the objects that the policy is applied to.

Modifying Security Policies in the Monitoring Window

When reviewing violations in the Monitoring window, if you see that obviously there is a problem in policy rule definitions, you can modify security policy rules directly in the Alerts window. For example, if you see an aggregated alert and you know that this is actually a false positive, you need to change the definitions of the policy that identifies violations.

To modify a policy in the Alerts window:

1. In the **Main** workspace, select **Monitoring > Alerts**. The **Alerts** window appears.

2. In the **Alerts** table, select the alert for which you want to change the policy definitions. The **Details** pane for the selected alert appears.
3. In the **Details** pane, click . The **Policy Details** window appears.
4. In the **Policy Details** window, you can modify the following settings:
   - **Policy Rules tab**: Modify policy rules as explained in the table named Security Policy Rule Parameters in Modifying Security Policies on page 177.
   - **Applied To tab**: View the read-only presentation of applied policies, applying of policies is done in the Security window, see Applying Policies on page 318.
   - **Advanced tab**: Add/edit policy exceptions. For more information, see Security Policy Exceptions on page 173, Enabling Response Page in Violations on page 175 and disable the learning process in the profile, see Disabling Automatic Web Profiling on page 278.
5. In the **Policy Details** window, click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, Activating Settings on page 75.

Filtering Policies

SecureSphere includes a filter that assists you in filtering policies in the policies window.

Note: The dynamic filter described here is available with Security policies. For all other policy types a simpler version of the filter is available.

To filter policies:

1. In the **Main** workspace, select **Policies**. The **Policies** window appears.
2. In the **Filters** pane, expand a category and enable the options you want to filter by.

   Note: You can enable multiple filter options from different categories. This displays results for only those items that have been enabled.

   The **Policies** pane displays all policies that belong to the selected categories.

To enable ADC keywords filtering categories:

1. In the **Admin** workspace, select **System Definitions > Keywords Settings**. The **Keywords Settings** pane appears on the right.
2. In the **Keywords Settings** pane, select the categories according to which you want to filter the policies in the **Security** window.

   Note: The user defined keywords are available for reports only.

You can use various filters to present policies in the **Security Policies** window, as presented in the following table.

Security Policy Filters

Filter Categories	Description
By ADC Keywords	Enables filtering policies according to the predefined categories to present the following industry standards: EBS, HPIAA, PCI, PeopleSoft, SAP, SOX. All the categories are disabled by default and in order to use them you need first to enable them in the **Admin** workspace.
By Type	Presents policies according to policy types, see Security Policy Types on page 167. All the policy types are divided to the following categories, web and network policies. In web categories the policy types are organized according to the domain object levels, service level and application level.
By Server Group	Presents all the server groups in the alphabetical order and enables you to view policies applied to the selected server group.
By Service	Presents all the services in the alphabetical order and enables you to view policies applied to the selected service.
By Application	Presents all the applications in the alphabetical order and enables you to view policies applied to the selected application.
Default Policies	Presents all default policies available in SecureSphere. For more information, see Using Default Security Policies on page 172.
Mitigation Policies	Displays mitigation policies that were created to mitigate a vulnerability. Over time SecureSphere can create a large number of mitigation policies. The show mitigation policy filter enables you to decide whether or not to display them. **Note:** Enabling both mitigation filter options, or neither option leads to all policies being displayed (both mitigation and non-mitigation policies). For more information on mitigation, see Marking a Vulnerability as Mitigated on page 161.

Legacy Policies

In the policies that use the ADC content some of the dictionaries include signatures from the other dictionaries which in certain cases when all applied might result in duplicated alerts. To avoid overlapping, each signature is included only once. In order to support backward compatibility, the dictionaries that are created earlier than the build 6274 receive a new name and the word *legacy* is added to its name.

For example, there are policies that have almost the same name, but include different signatures: Recommended Policy for General Applications and Recommended Policy for General Applications – legacy. The word *legacy* represents the policy that contains the old dictionaries.

Adding Comments to Security Policies

If **Enable comments for policy changes** in **Admin** > **System Definitions** > **Policy Settings** is on, you can add comments to a Security Policy.

Whenever you save a Security Policy, you can enter a comment describing, for example, what you did.

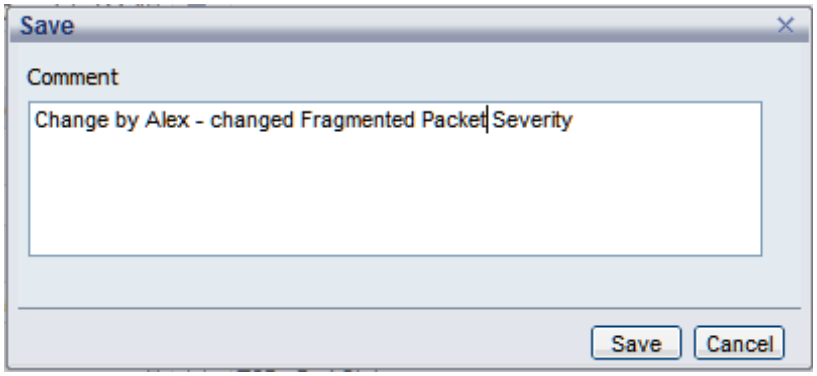


Figure 25: Security Policy Comment

If you click **Save**, the Security Policy is saved, and the comment is saved as a system event and in the log.

if you click **Cancel**, the Security Policy is not saved.

Note:

- Comments are not attributes of a Security Policy but are system events, and if system events are purged, the comments are no longer be visible.
- Comments are available only for Security Policies and not for other SecureSphere policies.

You can see the comment in the Security Policy's **Activity Log** tab, which is displayed only if **Enable comments for policy changes** is on.

You can also see the comment in the system event monitor (**Main** > **Monitor** > **System Events**).

You can filter reports in which the comments are present by the comment field.

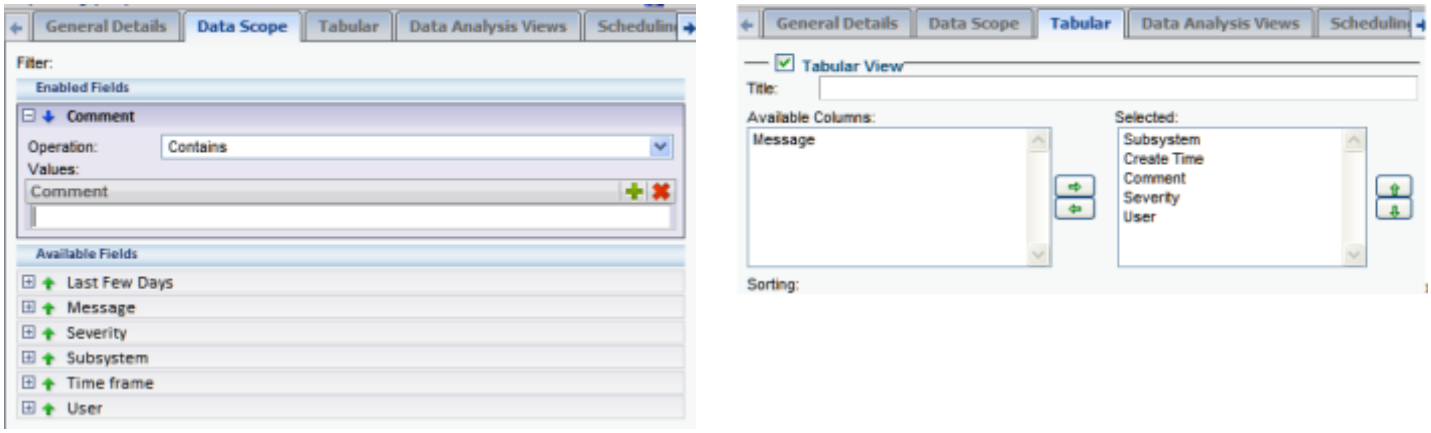


Figure 26: Filtering Reports by Comments

Network Security Policies

SecureSphere's network layer policies provide network access control of traffic flowing into and out of protected network application segments. These policies prevent unauthorized access to critical servers from unauthorized internal users using protocols such as Telnet, PCAnywhere, FTP, SMTPL. Firewall policies also play a critical role in a comprehensive worm defense strategy by preventing the spread of worms from internal user desktops to protected network application segments via non-essential ports.

Network security policies are applied on the server group level. These policies are basic policies, meaning that only one policy of each policy type can be applied to a server group. The following topics review subjects involving Network Security Policies:

Firewall Policy 184
Network Protocol Validation 186
Stream Signature 188

Firewall Policy

The Firewall policy defines what the permitted services in your network are and protects against worms that might take advantage of open ports in your system. For example, a worm can use an open port to attack applications. The Firewall policy enables you to define ports and source IPs that are allowed to send requests to your system, as follows:

- Traffic from/to the server - Internal/external server connections.
- To/from what IP address/port the server can send/receive in case it is infected.

Network firewall policies are defined for a server group, and can be configured for server groups that use inline gateways. When SecureSphere is in the sniffing mode, alerts are generated, but the connection is not blocked. For each protected server or a group of protected servers you can define a different firewall policy that meets the security requirements of that particular group. A Firewall policy is a basic policy, which means that only one policy can be applied to a server group. You can configure a few policies and apply only one of them.

For each protected server group you can define a different policy for internal and external traffic based on the white/black list principle, as follows:

- **White list**: The policy states only what is allowed and all the other traffic must be blocked.
- **Black list**: The policy states that everything is allowed except for the specific rules, meaning that only the traffic that must be blocked is defined in the policy.

For example, you can define a white list for the external traffic and a black list for the internal traffic.

The following table presents Firewall policy rule parameters.

Firewall Policy Rule Parameters

Parameter	Description
Name	The definition of the violation against which this rule protects.
Enabled	Enables the rule, meaning that this rule participates in protection provided by the policy. **Default**: Disabled.
Severity	The following severity levels can be configured for each rule: • **No Alert**: No alert is generated for the violation by this policy. Select **No Alert** if there is another policy which does alert on this violation, or on some variation of it, and you do not want multiple alerts for the same violation. **Note:** For policies with No Alert severity, Followed Action can not be configured, event's information is not reported and it cannot participate in the learning process. • **Informative**: Define in case of a very low risk, or when it is not a real attack, but an event was reported and it provides additional information that can be used for tuning purposes. Informative violations are not displayed in the dashboard. • **Low** • **Medium** • **High** **Default**: Informative.
Action	The immediate action that is taken when there is a match on the policy rule: • **None**: SecureSphere takes no action, and only generates violations and alerts, reporting about the security event. • **Block**: SecureSphere blocks the traffic that caused this violation and generates violations and alerts, reporting about the security event. **Default**: Block.
Followed Action	The action that is taken by SecureSphere in addition to the action defined by the Action parameter. The additional action is taken from the pre-configured list of actions defined in Action Sets, see Working with Action Sets and Followed Actions on page 326. The Followed Action is taken even when the Action is set to None. **Example**: once the violation is blocked, you want SecureSphere to send email to different people in the organization, notifying them about blocking of this specific violation.
Alert Name (read-only)	The name of the alert that appears in the Alerts table when this policy is violated. For example: Unauthorized Access to Service.
Always allow applicative service ports	Enables overriding of any choices made in the internal/external section, meaning that if this option is Enabled, service definitions come before policy definitions. For example, if an HTTPS service is mapped to port 8083 on a server group, then any traffic is allowed to that port. **Default**: Enabled.
Already established connections should be blocked	When this option is enabled, if the gateway goes up, or the server group IPs were just defined/added, the connections that were established before that are blocked. **Note:** If Already established connections should be blocked and Always allow applicative service ports are selected, there is a potential conflict. In such cases, the **Always allow applicative service ports** is dominant. **Default**: Disabled.

Parameter	Description
Internal/External	Firewall rules are divided into Internal - traffic designated to the server group, and External - traffic that originates from the server group. Enables to select how the Firewall policy is defined for each traffic direction: • **Permit**: The policy is defined as a white list, meaning that only the protocols that appear in the table are allowed to enter/exit your system and all the others must be blocked. • **Block**: The policy is defined as a black list, meaning that only the protocols that appear in the table are blocked. **Default**: Block.
Protocol	The service that is blocked/permitted in the Internal/External traffic. Each Firewall policy is a combination of service, such as FTP, SMTP, and a source or destination IP address group. The protocols are mapped into ports in the global ports group.
Source	The IP group that you want to protect using this Firewall policy. The segments are defined in Global Objects and are used globally in SecureSphere, see Configuring IP Groups on page 347.

Network Protocol Validation

The Network Protocol Validation policy ensures a proper use of the TCP/ IP or UDP protocol, preventing hackers from misusing the protocol or taking advantage of its known vulnerabilities. In addition to validating against the RFC, there are some additional predefined rules that are based on Imperva ADC research as well as best practice, for example the rule SSL Untraceable Connection.

The SSL Untraceable Connection rule is also included under this policy to alert you in case SecureSphere cannot monitor the SSL traffic.

For example, an illegal TCP flag such as TCP SYN with FIN is immediately identified by the Network Protocol Validation policy.

The following table shows the rules included in the Network Protocol Validation policy.

Network Protocol Validation Policy Rules

Rule	Description
Bad IP Option Length	The specified option length does not match the standard for that option or the header length. **Defaults**: Disabled; Severity - Medium; Action - None.
Bad IP Option Padding	Non zero option was found after the end of the IP options list. **Defaults**: Disabled; Severity - Medium; Action - None.
Bad Source IP Address	The IP source address is 0.0.0.0 or 255.255.255.255. **Defaults**: Disabled; Severity - Medium; Action - None.
Bad TCP Option Length	The specified option length does not match the standard for that option or the header length. **Defaults**: Disabled; Severity - Medium; Action - None.
Bad TCP Option Padding	Non zero option found after the end of the TCP options list. **Defaults**: Disabled; Severity - Medium; Action - None.
Fragmented Packet	Enables you to block all IP fragments. Although fragmentation could be allowed on some parts of the network, in many cases where SecureSphere is located (in front of the DB/web), packets must not be fragmented. **Defaults**: Disabled; Severity - Informative; Action - None.
ICMP - LAND Attack	The source IP address is identical to the destination IP address. Suspected Land attack. **Defaults**: Disabled; Severity - Medium; Action - None.
IP - Bad Fragment	Bad Fragment: Multiple or different last fragment, or corrupted overlapping fragments. **Defaults**: Disabled; Severity - Medium; Action - None.
IP - Bad Total Length	The total length of the IP packet does not match the Ethernet frame size. **Defaults**: Disabled; Severity - Medium; Action - None.
IP - Micro-Fragment	The packet does not include the complete IP header, suspected microfragement attack. **Defaults**: Disabled; Severity - Medium; Action - None.
IP - Too Small TTL	A small TTL means that attacker manipulated the data, so when the TTL of the IP packet is lower than a configured threshold, it indicates an attack. Define the **Minimum Allowed TTL** parameter. Default: 2. The value can be defined between 0-255. **Defaults**: Disabled; Severity - Medium; Action - None.
Illegal TCP Option	Non compliant Packet: Misused TCP option, for instance SACK permitted IP option in non SYN packet. **Defaults**: Disabled; Severity - Informative; Action - None.
Invalid IP Flags	The reserved flag bit is not zero, or no fragment flag is set but more fragments or offset are not zero. **Defaults**: Disabled; Severity - Informative; Action - None.
Modified TCP Retransmission Data (client to server only)	Retransmission with altered content. Only the out of order packets are checked. **Defaults**: Disabled; Severity - Medium; Action - None.
SSL Untraceable Connection	SecureSphere cannot monitor the traffic in this SSL connection, as it could not extract the required SSL session keys. This issue can emerge in the following cases: • If the SSL connection was initiated before the SecureSphere boot process was completed. • If SecureSphere has no access to the server's private key used to establish this session. • If the SSL cipher method used for this session is not supported by SecureSphere.

Rule	Description
	Defaults: Enabled; Severity - Informative; Action - None.
TCP - Bad State	Invoked when the sequence of TCP packets does not match the standard TCP state machine, which happens due to incorrect network configuration, for example when only one traffic direction is seen. **Defaults**: Disabled; Severity - Medium; Action - None.
TCP - LAND Attack	The source IP address is identical to the IP destination address and the source port is identical to the destination port. Suspected Land attack. **Defaults**: Disabled; Severity - Medium; Action - None.
TCP - Micro-Fragment	The packet does not include the complete TCP header. **Defaults**: Disabled; Severity - Medium; Action - None.
TCP - Non Zero Reserved Flags	Non compliant Packet: Reserved flags in the TCP header are not zero. **Defaults**: Disabled; Severity - Informative; Action - None.
TCP - SYN with Data	Non compliant SYN packet with non zero data length. **Defaults**: Disabled; Severity - Informative; Action - None.
TCP - SYN with FIN	Non compliant packet with both SYN flag and FIN flag. **Defaults**: Disabled; Severity - Informative; Action - None.
TCP - SYN with Non Zero ACK Sequence	Non compliant Packet: ACK sequence in a SYN packet is not zero. **Defaults**: Disabled; Severity - Informative; Action - None.
TCP - SYN with RST (not SYN ACK)	Non compliant Packet with both SYN flag and RST flag. **Defaults**: Disabled; Severity - Informative; Action - None.
TCP Unexpected SYN	SYN packet sent along an established connection. **Defaults**: Disabled; Severity - Informative; Action - None.
UDP - Micro-Fragment	Non compliant packet: The packet does not include the complete UDP header. Suspected microfragment attack. **Defaults**: Disabled; Severity - Medium; Action - None.
Unknown IP Option	Unknown IP option: Option type is not standard. **Defaults**: Disabled; Severity - Informative; Action - None.
Unknown TCP Option	TCP header options: TCP option type is not standard. **Defaults**: Disabled; Severity - Informative; Action - None.

Stream Signature

The Stream Signature policy protects against service attacks, for, example, FTP command overflow attempt. The protection is based on the use of general signatures.

The Stream Signature policy is a default policy provided by ADC. The policies include dictionaries that contain signatures intended for detection and blocking of worms and vulnerabilities that attack services. The policy is violated if the signature analyzer identifies a signature pattern in an event.

These policies are applied by default when a server group is added to a site.

The **Recommended Policy for General Applications - Legacy Rules**, see Legacy Policies on page 182. The **Recommended Signatures Policy for General Application Rules** policy includes the following dictionaries:

- Recommended for Blocking for General Applications (**Defaults**: Enabled; Severity - High; Action - Block.)
- Recommended for Detection for General Applications (**Defaults**: Enabled; Severity - Low; Action - None.)
- Worms and Critical Vulnerabilities for General Applications (**Defaults**: Enabled; Severity - High; Action - Block.)

Securing Web Transport

The service level policies protect at the service protocol layer. SecureSphere protocol compliance ensures that the use of HTTP protocol meets the RFC standard. By ensuring that the usage of the HTTP protocol meets standards, protocol compliance prevents attacks on both known and unknown vulnerabilities in commercial web server implementations.

Imperva ADC has conducted comprehensive research and collected a group of protocol violations that usually indicate attack attempts. You can enable or disable each of these violations for each group of protected servers.



Note: Web security policies support user names of up to 48 characters.

You can configure the actions to be taken for all protocol violations defined in the system. The following policy types are available for securing web transport in SecureSphere:

HTTP Protocol Validation Policy 189
Cookie Signing Validation Policy 195

HTTP Protocol Validation Policy

HTTP protocol validation prevents protocol exploits including buffer overflow, malicious encoding, HTTP smuggling, and illegal server operations. Imperva provides you with a default policy that enables strict adherence to RFC standards, while allowing minor variations for specific applications.

Before the validation, SecureSphere normalizes the URL to provide a common basis for validation. This includes decoding, conversion to UTF8, conversion to lower case (if service is configured as case insensitive), directory normalization and so on.

The HTTP protocol validation policy enables to protect your system against the evasion techniques, such as encoding attempts. It is very important to stop the evasion technique on the service level. If the malicious traffic is not blocked on the service level, then the application level SecureSphere policies are not able to provide protection, as the protection is evaded.

The following table presents the HTTP Protocol Validation Policy rules.

Web Protocol Policy Rules

Rule	Description
Abnormally Long Header Line	The length of either the HTTP request Header name (for example Content-Type, Accept, and so on) or value exceeds the maximum threshold. Although the HTTP specification does not define a specific length limitation, when Header lines exceed this length it usually indicates a buffer overflow attempt. You can tune the maximum threshold for the following parameters: • **Maximum HTTP Request Header Name Length**: If the length of the HTTP Request Header Name exceeds the predefined value, SecureSphere identifies it as a violation. **Default**: 50. • **Maximum HTTP Request Header Value Length**: If the length of the HTTP Request Header Value exceeds the predefined value, SecureSphere identifies it as a violation. **Default**: 4096. **Defaults**: Enabled; Severity - Low; Action - None.
Abnormally Long Request	The length of one of the following parts of an HTTP request exceeds the maximum threshold: HTTP Method, URL, Query String, HTTP Version. Although the HTTP specification does not define a specific length limitation, a URL that exceeds this length usually indicates a buffer overflow attempt or a script injection attempt. Violation of this length limitation may also indicate other types of attacks, such as partial validation by servers (the web server validates only the first part of a long string while the operating system or other backend service processes the entire string). You can tune the maximum threshold for the following parameters: • **Maximum HTTP Method Length** **Default**: 15. • **Maximum Query String Length** **Default**: 4096. • **Maximum URL Length** **Default**: 500. • **Maximum HTTP Version Length** **Default**: 8. **Defaults**: Enabled; Severity - Low; Action - None.
Double URL Encoding	URL encoding is a standard based format for embedding non-printable, or special characters within HTTP request fields (such as the URL). The character is represented by a percentile symbol ('%') followed by the hexadecimal representation of its value (e.g. the TAB character whose ASCII value is 9 can be represented as '%09'). Double URL encoding is an evasion technique used by attackers to bypass access control, authorization and detection mechanisms applying URL encoding multiple times to the attack URL. For example, a '/' character in a directory traversal attack would be encoded as %252F which is the result of applying URL encoding twice. **Defaults**: Enabled; Severity - Low; Action - None.

Rule	Description
Extremely Long Parameter	The length of a parameter exceeds the maximum defined length (default is 4096 characters). Although the HTTP specification does not define a specific length limitation, a parameter that exceeds this length usually indicates a buffer overflow attempt. Violation of this length limitation may also indicate other types of attacks, such as partial validation by servers (the web server validates only the first part of a long string while the operating system or other backend service processes the entire string). This violation applies to HTTP requests using POST methods only. You can define the maximum threshold for the following parameters: • Maximum URL Parameter Name Length • Maximum URL Parameter Value Length • Maximum SOAP Element Name Length • Maximum SOAP Element Value Length **Defaults**: Enabled; Severity - Low; Action - None.
Illegal Byte Code Character in Header Name	An HTTP header name contains characters banned by HTTP standard. These are usually non-printable ASCII characters or other special characters (e.g. '<', '>', etc.). Each header name is checked individually against the set of characters allowed by the RFC (number=2616). **Defaults**: Enabled; Severity - High; Action - Block.
Illegal Byte Code Character in Method	An HTTP method contains characters banned by HTTP standard. These are usually non-printable ASCII characters or other special characters(e.g. '<', '>', etc.). Each HTTP method is checked individually against the set of characters allowed by the RFC (number=2616). **Defaults**: Enabled; Severity - High; Action - Block.
Illegal Byte Code Character in Parameter Name	A parameter name in the request's content contains characters banned by HTTP standard. These are usually non-printable ASCII characters or other special characters(e.g. '<', '>', etc.). Each parameters name is checked individually against the set of characters allowed by the RFC (number=2616). **Defaults**: Enabled; Severity - High; Action - Block.
Illegal Byte Code Character in Parameter Value	A parameter value in the request's content contains characters banned by HTTP standard. These are usually non-printable ASCII characters or other special characters(e.g. '<', '>', etc.). Each parameters value is checked individually against the set of characters allowed by the RFC (number=2616). **Defaults**: Enabled; Severity - Medium; Action - None.
Illegal Byte Code Character in Query String	A URL query string contains characters banned by HTTP standard. These are usually non-printable ASCII characters or other special characters(e.g. '<', '>', etc.). Each URL query string is checked individually against the set of characters allowed by the RFC (number=2616). **Defaults**: Enabled; Severity - High; Action - Block.
Illegal Byte Code Character in URL	A URL contains characters banned by HTTP standard. These are usually non-printable ASCII characters or other special characters(e.g. '<', '>', etc.). Each ULR is checked individually against the set of characters allowed by the RFC (number=2616). **Defaults**: Enabled; Severity - High; Action - Block.
Illegal Chunk Size	Illegal chunk size was detected. Chunk size is expected to be hexa-numeric type. This violation indicates that a chunk size value other than hexadecimal was detected. **Defaults**: Enabled; Severity - Low; Action - None.
Illegal Content	The Content-Length header value on a request or response is not a numeric value. **Defaults**: Enabled; Severity - High; Action - Block.

Rule	Description
Length	
Illegal Content Type	The structure of the content type is not in the format of type/subtype and hence is illegal. **Defaults**: Enabled; Severity - Low; Action - None.
Illegal HTTP Version	Each HTTP request must contain the version of the protocol used by the client, in the form of HTTP/v where v is replaced by the actual version number (one of 0.9, 1.0, 1.1). This anomaly indicates a malformed request not sent from a normal browser. **Defaults**: Enabled; Severity - High; Action - Block.
Illegal Host Name	The host name field within an HTTP request contains characters banned by HTTP standard. These are usually non-printable ASCII characters or other special characters(e.g. '<', '>', etc.). The host name is checked against the set of characters allowed by the RFC (number=2616). **Defaults**: Enabled; Severity - High; Action - Block.
Illegal Parameter Encoding	URL encoding is a standard based format for embedding non-printable, or special characters within HTTP request fields (such as the URL). The character is represented by the percent symbol ('%') followed by the hexadecimal representation of its value (e.g. the TAB character whose ASCII value is 9 can be represented as '%09'). Illegal encoding occurs when the percent symbol is not followed by a hexadecimal value (for instance, '%XY'). If such illegal encoding occurs in the parameters of the request, this violation is generated. **Defaults**: Enabled; Severity - Low; Action - None.
Illegal Response Code	An HTTP response does not include a legal HTTP response code. The HTTP standard specifies that code should be a 3 digit number. A non-compliant code indicates an abnormal response which is usually the consequence of a severe failure. **Defaults**: Enabled; Severity - Low; Action - None.
Illegal URL Path Encoding	URL encoding is a standard based format for embedding non-printable, or special characters within HTTP request fields (such as the URL). The character is represented by a percentile symbol ('%') followed by the hexadecimal representation of its value (e.g. the TAB character whose ASCII value is 9 can be represented as '%09'). Illegal encoding within the URL is a technique used by malicious individuals both for evasion and attack by embedding ill-crafted sequences preceded by the percentile symbol. These sequences (e.g., %vv, %t0, etc.) are banned by the standard and are usually decoded in an unexpected manner by different web servers. **Defaults**: Enabled; Severity - Low; Action - None.
Malformed HTTP Header Line	An HTTP Header field is comprised of a name (e.g. Content-type, Content-length, etc.) and a value separated by a Colon (':') character. A malformed header line does not include the Colon character making it impossible to correctly parse the content of the line. This is an indication of an attack on the server's parsing mechanism. **Defaults**: Enabled; Severity - Low; Action - None.
Malformed SOAP Message	If the HTTP request includes an XML/SOAP message which cannot be parsed, SecureSphere invokes this violation. **Defaults**: Enabled; Severity - Medium; Action - None.
Malformed URL	The URL in a request must begin with a '/' character and may include a protocol and host prefix (e.g. http://myserver.com/default.asp). Omission of the '/' character or a protocol prefix other than http are usually an indication of a malicious attempt to tunnel other protocols (e.g. SMTP) using the web server. **Defaults**: Enabled; Severity - Low; Action - None.

Rule	Description
NULL Character in Header Name	The use of a NULL (0 valued) character within the name or value of an HTTP request Header field is banned by the standard. An attacker embeds NULL characters within the Header field in order to evade detection mechanisms. **Defaults**: Enabled; Severity - Low; Action - None.
NULL Character in Method	The use of a NULL (0 valued) character within an HTTP method is banned by the standard. An attacker embeds NULL characters in order to evade detection mechanisms. **Defaults**: Enabled; Severity - Low; Action - None.
NULL Character in Parameter Name	The use of a NULL (0 valued) character within an HTTP parameter name is banned by the standard. An attacker embeds NULL characters within the parameter name in order to evade detection mechanisms. **Defaults**: Enabled; Severity - Low; Action - None.
NULL Character in Parameter Value	The use of a NULL (0 valued) character within an HTTP parameter value is banned by the standard. An attacker embeds NULL characters within the value name in order to evade detection mechanisms. **Defaults**: Enabled; Severity - Low; Action - None.
NULL Character in Query String	The use of a NULL (0 valued) character within a query string is a violation of the standard.An attacker embeds NULL characters in order to evade detection mechanisms. **Defaults**: Enabled; Severity - Low; Action - None.
NULL Character in URL	The use of a NULL (0 valued) character within a URL is a violation of the standard.An attacker embeds NULL characters in order to evade detection mechanisms. **Defaults**: Enabled; Severity - Low; Action - None.
Redundant HTTP Headers	An HTTP request contains protocol information and processing hints in header fields. Some of the fields have a crucial role in the interpretation of the message by the server and a significant effect on the processing of the request by the server. Those fields are required to appear only once in any request. The "Redundant HTTP Headers" anomaly is invoked if an HTTP request contains multiple instances of such headers. The default list of headers that are affected by this anomaly includes the "Content-length" header field (which affects the number of bytes from the network stream that are associated with the message), the "Content-type" header field (which affects the parsing of the message content), the "Host" header fields and others. **Defaults**: Enabled; Severity - Medium; Action - None.
Redundant UTF-8 Encoding	UTF-8 is a popular character encoding scheme for representing Unicode characters in using variable length byte sequences. Promiscuous interpretation of UTF-8 strings by web servers results in the translation of multiple sequences into the same ASCII character (e.g. '/' or '.'). This technique known as "Redundant UTF-8 Encoding" is used by attackers to evade access control, authorization and detection mechanisms. **Defaults**: Enabled; Severity - High; Action - Block.
Slow HTTP From a Single Source	HTTP connections from a single source send data at very low rates, indicating an attempt to bring down the web server by exhausting its resources. **Number of Slow Connections**: The maximum number of slow connections allowed from a single source, default - 50. **Defaults**: Enabled; Severity - High; Action - None. You can define additional parameters relating to slow HTTP/S connections in the Advanced tab of the HTTP Protocol Validation Policy " Advanced HTTP Protocol Validation Policy" on page 195.

Rule	Description
Slow HTTP From Multiple Sources	HTTP connections from multiple sources send data at very low rates to the same web service, indicating an attempt to bring down the web server by exhausting its resources. **Number of Slow Connections**: The maximum number of slow connections allowed from multiple sources, default - 100. **Defaults**: Enabled; Severity - High; Action - None. You can define additional parameters relating to slow HTTP/S connections in the Advanced tab of the HTTP Protocol Validation Policy " Advanced HTTP Protocol Validation Policy" on page 195.
Slow HTTPS From a Single Source	HTTPS connections from a single source send data at very low rates, indicating an attempt to bring down the web server by exhausting its resources. **Number of Slow Connections**: The maximum number of slow connections allowed from a single source, default - 50. **Defaults**: Enabled; Severity - High; Action - None. You can define additional parameters relating to slow HTTP/S connections in the Advanced tab of the HTTP Protocol Validation Policy " Advanced HTTP Protocol Validation Policy" on page 195.
Slow HTTPS From Multiple Sources	HTTPS connections from multiple sources send data at very low rates to the same web service, indicating an attempt to bring down the web server by exhausting its resources. **Number of Slow Connections**: The maximum number of slow connections allowed from multiple sources, default - 100. **Defaults**: Enabled; Severity - High; Action - None. You can define additional parameters relating to slow HTTP/S connections in the Advanced tab of the HTTP Protocol Validation Policy " Advanced HTTP Protocol Validation Policy" on page 195.
Too Many Cookies in a Request	SecureSphere invokes this anomaly if an HTTP request includes too many cookies. Allowed Number of Requests: default - 100. **Defaults**: Disabled; Severity - Informative; Action - None.
Too Many Headers per Request	An HTTP request contains too many Header fields. This is an indication of a buffer overflow attack or an attempt to evade detection mechanisms. Allowed Number of Requests: default - 25. **Defaults**: Enabled; Severity - Low; Action - None.
Too Many Headers per Response	An HTTP reply contains too many Header fields. This is an indication of information leakage (e.g. credit card numbers, user identifiers, etc.) through the Header fields rather than through the response body. Allowed Number of Responses: default - 20. **Defaults**: Enabled; Severity - Medium; Action - None.
Too Many URL Parameters	The number of parameters in the URL exceeds the allowed threshold. Note that this violation only applies to HTTP requests that use the POST methods. HTTP requests that use the GET method will not invoke this violation. Allowed Number of Requests: default - 1000. **Defaults**: Enabled; Severity - High; Action - Block.
URL is Above Root Directory	URL points to a file residing outside of the Web server's root directory. This usually indicates a directory traversal attack in which the attacker tries to access files outside of the root directory. In some poorly written applications some embedded links may point outside of the Web server's root directory. **Defaults**: Enabled; Severity - High; Action - Block.

Rule	Description
Unauthorized Request Content Type	This anomaly occurs for a variety of HTTP requests in which the content-type cannot be correctly established according to the RFC, or in which the content-type is identified as an unauthorized type. **Defaults**: Enabled; Severity - Medium; Action - None.
Unknown HTTP Request Method	Each HTTP request must start with a field called HTTP method or HTTP verb (e.g. GET or POST). The list of methods is defined by various standards. Some of the methods have a potential dangerous effect on the web server and hence only a partial list of the methods should be allowed by most web servers. This anomaly detects a method that is not one of the defaults, indicating an attempt to execute a dangerous and illegal operation on the web server. **Defaults**: Enabled; Severity - High; Action - Block.

Advanced HTTP Protocol Validation Policy

In the **Advanced** tab of an HTTP Protocol Violation Policy, you can specify additional parameters of the policy.

Community Defense Settings

Allow sending event information to Community Defense - If enabled, SecureSphere will send event information from this site to the Imperva Community Defense database.

Exceptions

In this section you can specify the conditions when the policy will not be applied.

Response Page Settings

Display response page in alerts - If enabled, the response will be displayed in the alert.

Slow HTTP/S Attack

In this section you can define the parameters that identify a slow HTTP/S attack.

Number of Bytes Per Second - SecureSphere will identify a connection as a slow HTTP/S attack if its bytes per second rate is below **Number of Bytes Per Second**.

Lead Time (Seconds) - The number of seconds after the connection starts at which SecureSphere will begin measuring the bytes/second.

Resend Violation Every (In Seconds) - SecureSphere will resend violations only after **Resend Violation Every** seconds have passed since the last violation was sent.

Cookie Signing Validation Policy

You can specify that SecureSphere signs and/or encrypts cookies, and applies a Cookie Signing Validation policy to verify that cookies are signed and/or verified. For more information about cookies, refer to Cookie Protection in Web Profiles on page 252. The following subjects provide information regarding cookies and their management through policies:

- Cookie Signing on page 195
- Cookie Encryption on page 196
- Enabling and Disabling Cookie Signing and Encryption on page 196

Cookie Signing

When cookie signing is enabled, SecureSphere attaches a signature to cookies when they are sent to the client and verifies that the signature is present and correct when the cookie returns from the client. Signing cookies protects against cookie tampering and cookie injection.

Cookie Encryption

When cookie encryption is enabled, SecureSphere both signs cookies and encrypts them. Cookie encryption enhances protection against cookie tampering and cookie injection, and also protects sensitive data which may be present in the cookie.

Enabling and Disabling Cookie Signing and Encryption

A Cookie Validation Policy is applied at the service level. When you apply a Cookie Validation Policy to a service, the cookies defined in the policy are automatically signed. The policy specifies what actions to take in the event of an incorrect or missing cookie signature.



Note: Applying a Cookie Validation Policy to a service implicitly enables cookie signing for the service. In contrast, cookie encryption must be explicitly enabled in the Cookie Validation Policy.

To enable or disable cookie signing and encryption:

1. In the **Main** workspace, select **Policies** > **Security** and under **Web**, select **Service Level** > **Cookie Signing Validation**.
2. In the middle pane, select a policy.
3. In the right pane, in the **Policy Rules** tab, specify the parameters as given in the following table.

Cookie Policy Signing Rules Parameter

Parameter	Description
Incorrect cookie signature	If enabled, SecureSphere signs the cookies on their way to the client and verifies signatures in cookies returning from the client to the server. If the signature is invalid, SecureSphere generates an alert with the specified **Severity Level** and takes the **Action** and **Followed Action** as specified. For more information, see Modifying Policies in the Security Window on page 176. **Defaults**: Disabled; Severity - High; Action - Block.
Missing cookie signature	If enabled, SecureSphere verifies that cookie signatures are present in cookies returning from the client to the server. If the signature is not present, SecureSphere generates an alert with the specified **Severity Level** and takes the **Action** and **Followed Action** as specified. **Default**: Disabled; Severity - Low; Action - None.
Encrypt cookie value	If enabled, cookies will be encrypted and decrypted by the SecureSphere gateway. **Default**: Disabled.
Action for non blocked connections with incorrect or missing cookie signatures or missing encryption	This can happen if a cookie was set before SecureSphere was deployed. Choose one of the following options: • **Remove** the cookie and let the request pass. • **Redirect** the request to the specified URL. • None **Default**: **None**.
Include/Exclude	Define the match criteria for applying the policy. Select **Include** to specify a list of cookies to which to apply this policy. If only some cookies need to be protected, list them here. Select **Exclude** to specify a list of cookies to which the policy is not applied. The policy is applied to all cookies except the cookies under **Exclude**. Enable **Prefix** if the cookies have dynamic names, for example, the Microsoft IIS ASPSESSION cookie, which usually consist of a fixed prefix followed by a dynamic suffix such as ASPSESSIONIDFFEDSSE. For these cookies, enter only the fixed prefix as the cookie name. Matching is performed against the prefix only, and the dynamic part of the cookie's name is ignored.



Note: SecureSphere signs cookies, based on the match criteria, for all services to which a Cookie Signing Validation Policy is applied.

HTTP Protocol Signature Policies

A **Signature** is a method of attack detection based on pattern match that can be used only against known attacks with stable data, see Intrusion Prevention Used in Signature Policies on page 170. The HTTP Protocol Signature policies are the way to apply Signature dictionaries. Each policy defines how the dictionary is used, activated and also how to react in case an attack is detected.

The **Recommended Policy for Web Applications - Legacy**, as explained in Legacy Policies on page 182. The **Recommended Signatures Policy for Web Applications** policy is used to protect web applications on the service level. This policy includes the following signature dictionaries:

- MSSQL Data Retrieval with Implicit Conversion Errors (**Defaults**: Enabled, Severity - No Alert, Action - None)

- Recommended for Blocking for Web Applications (**Defaults**: Enabled, Severity - High, Action - Block)
- Recommended for Detection for Web Applications (**Defaults**: Enabled, Severity - Low, Action - None)
- Worms and Critical Vulnerabilities for Web Applications (**Defaults**: Enabled, Severity - High, Action - Block)

Web Service Custom Policies

Web Service Custom policies provide you with a flexible engine that correlates between the server group and service protection levels and which is based on CAV technology, see Correlated Attack Validation (CAV) Used in Correlation Policies on page 170. This is done by creating user-defined policies based on different combinations of the match criteria. Correlation policies can be manually configured or are available as predefined rules that cover certain well-known scenarios.

For example, using custom correlation policies you can limit access to specific URLs and directories based on the source IP address. You can also restrict the permitted HTTP headers, user agents (browsers), and so on. You can define as many custom correlation policies as you want. It is recommended to perform as many operations as possible using predefined profile/protocol violation policies and use custom correlation policies only for operations that are otherwise impossible to perform.



Note: SecureSphere does not support matching of white characters used in Asian languages according to UTF8 in web responses. The response content with Asian languages can be matched by SecureSphere only if the application itself uses UTF8.

Web Custom Policies Match Criteria on page 601 presents match criteria used in the web custom policies.

EXAMPLE: AUTOMATED VULNERABILITY SCANNING

The Automated Vulnerability Scanning policy is one of the predefined web custom policies that is used to detects and prevent scanning attempts. Understanding of how this policy is defined using web custom match criteria can help you to understand how to define web custom policies.

If SecureSphere identifies a number of violations in the predefined period of time that is higher than the predefined number, it indicates that there is a scanning attempt.

To detect and prevent scanning attempts:

1. In the main window, select **Policies > Security**, in the Policies pane, click  and select WEB Service. The **Create New Policy** dialog box appears.
2. Type the name of the new policy, from the **Type** drop-down box, select WEB Service Custom and click Create.
3. In the **Match Criteria** tab, click on the green arrow next to **Violations**.
4. Expand the **Violations** match criteria and select the violations on which you want to get alerts.
5. To define match over time, click on the green arrow next to **Occurrence**.
6. Expand the **Occurrence** match criteria and define the following parameters:
7. Define the reaction settings, see Modifying Policies in the Security Window on page 176.
8. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Option	Description
Occurred more than	Set the maximum number of the violations that you allow in your web site.
Within	Set the period of time in seconds in which the violations are permitted.
In a context of a single	The number of occurrences within the defined period of time must be a part of the same session. Select **Originating Session**.

EXAMPLE: BLOCKING TRAFFIC FROM A SPECIFIC GEOGRAPHIC LOCATION

Malicious attacks can come from any location. Though every day concerted attacks whether for profit are often launched from a specific geographic locations. You can use the SecureSphere Source GeoLocation criteria to configure a custom Security Policy to block undesired traffic originating from a specific geographic location based on country name.



Note: You must have an active ThreatRadar Reputation Services license to use the GeoLocation feature and configure the below settings.

To configure a policy based on geographic location:

1. In the main window, select **Policies > Security**, in the Policies pane, click  and select Web Service. The **Create New Policy** dialog box appears.
2. Type the name of the new policy, from the **Type** drop-down box, select Web Service Custom and click Create.
3. Click on the green arrow next to **Source GeoLocation**.
4. Expand the **Source GeoLocation** match criteria. Then under **Action**, select **Block**.
5. Add China and Venezuela to the **Selected** pane.
6. Click **Save**. Settings are saved. Any traffic that emanates from China or Venezuela will be blocked.

EXAMPLE: TUNING THE HTTP PROTOCOL VALIDATION POLICY

Sometimes the custom policies mechanism can be used not only to define new custom policies, but also to tune the existing policies. For example, you do not want to block your incoming traffic or to get alerts when the Abnormally Long Request violation is detected, which is a part of the HTTP Protocol Validation policy. It is important for you to get alerts, but only during the working hours. In addition, you do not want to get alerts on the traffic that arrives from the development team.

To define the custom policy:

1. In the main window, select **Policies > Security**, in the Policies pane, click  and select WEB Service. The **Create New Policy** dialog box appears.
2. Type the name of the new policy, from the **Type** drop-down box, select WEB Service Custom and click Create.
3. In the **Match Criteria** tab, click on the green arrow next to **Violations**.
4. Expand the **Violations** match criteria and select **Abnormally Long Request**.
5. To define match over time, click on the green arrow next to **Occurrence**.
6. Expand the **Occurrence** match criteria and define the parameters in the table **Occurrence Parameters** below.
7. Select **Time of Day**, expand it and highlight the days and hours in which you want to get alerts.
8. Define the reaction settings, see Modifying Policies in the Security Window on page 176.
9. To avoid applying this policy on the development team, select **Source IP Addresses** and set the parameters in **Source IP Address Parameters** below.

10. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Occurrence Parameters

Option	Description
Occurred more than	Set the maximum number of violations that you want to allow within the predefined period of time.
Within	Set the period of time in seconds in which the violations are permitted.
In a context of a single	The number of occurrences within the defined period of time must be a part of the same session. Select **Originating Session**.

Source IP Address Parameters

Option	Description
Operations	Select **Exclude All**.
IP Groups	Select the predefines IP group that contains IP addresses of the development team, see Configuring IP Groups on page 347.

The following table presents the predefined policies that can help you to protect your web services against specific attacks and threats.

Predefined Web Service Custom Policies

Predefined Policy	Description
Anonymous Proxies Detection	Detects connections coming from the anonymous proxy. The proxy is located on the Internet and it hides the real IP address. ADC maintains the list of such proxies, so that it can compare the incoming IP address to the list of anonymous proxies, which might be a potential attacker. **Defaults**: Enabled; Severity - High; Action - None.
Anti Google Hacking	Prevents Google crawling attach on your site that might lead to sensitive data leakage. The following predefined policies are available: • **Anti Google Hacking - 1**: Provides wider coverage. • **Anti Google Hacking - 2**: Provides more accurate protection, as it contains a single rule with many match criteria to make it more accurate. **Defaults**: Enabled; Severity - High; Action - Block.
Apache Expect Header XSS	Prevents Cross Site Scripting based on Apache Expect Header. **Defaults**: Enabled; Severity - High; Action - Block.
Automated Site Reconnaissance/Access	Identifies attackers that use automated tools, which scan the web site and look for vulnerabilities in order to plan future attacks. **Defaults**: Enabled; Severity - High; Action - None.
Automated Vulnerability Scanning	Identifies attackers that use automated tools, which scan the web site and look for vulnerabilities in order to attack this site. **Defaults**: Enabled; Severity - High; Action - None.
Cross Site Request Forgery	Protects against the Cross Site Request Forgery attack. **Defaults**: Enabled; Severity - High; Action - None.
Data Leakage	The following policies protect the following credit cards against data leakage: • Data Leakage - American Express Credit Card Numbers (**Defaults**: Enabled; Severity - High; Action - Block). • Data Leakage - Application Source Code: Identifies technical data leakage in the error codes (**Defaults**: Enabled; Severity - High; Action - Block). • Data Leakage - Developer Comments: Prevents exposure of the HTML comments written by the developer that built the site. Using the information about how the site is built, the hackers might easily attack the site (**Defaults**: Enabled; Severity - High; Action - None). • Data Leakage - Diner's Club / Carte Blanche Credit Card Numbers (**Defaults**: Enabled; Severity - High; Action - Block). • Data Leakage - Discover Credit Card Numbers (**Defaults**: Enabled; Severity - High; Action - Block). • Data Leakage - JCB Credit Card Numbers (**Defaults**: Enabled; Severity - High; Action - Block). • Data Leakage - MasterCard Credit Card Numbers (**Defaults**: Enabled; Severity - High; Action - Block). • Data Leakage - U.S Social Security Number (**Defaults**: Enabled; Severity - High; Action - Block). • Data Leakage - Visa, Long Credit Card Numbers (**Defaults**: Enabled; Severity - High; Action - Block). • Data Leakage - Visa, Short Credit Card Numbers (**Defaults**: Enabled; Severity - High; Action - Block). • Data Leakage - enRoute Credit Card Numbers (**Defaults**: Enabled; Severity - High; Action - Block).

Predefined Policy	Description
Directory Browsing Detection	Prevents the situation when the users are trying to play with the site structure in the URLs and get to various places on the site including the locations where they are not supposed to be. SecureSphere identified the server respond that indicates that the user is in the problematic location. **Defaults**: Enabled; Severity - High; Action - Block.
Directory Traversal (In Cookies/Parameters Value)	Identifies the Directory Traversal attack in the following locations: • cookie/parameter if there is a file name that indicates the sensitive location in cookie/parameter (**Defaults**: Enabled; Severity - High; Action - Block). • In URL (**Defaults**: Enabled; Severity - High; Action - Block). • In URL - Basic Rule (**Defaults**: Enabled; Severity - Medium; Action - None).
File Download Injection	Prevents the attacker from injecting the Content-Disposition HTTP header. Content-Disposition allows web applications to set presentational information for the content sent. The header can be used to determine whether the content is treated as an attachment or as an in-line content that is automatically displayed when the message is viewed. By injecting the Content-Disposition header the attacker can trigger a file download at the user with injected content as an attachment or content that is handled by the default action defined by the user, while the content appears to be as originating from the trusted host. **Defaults**: Enabled; Severity - High; Action - Block.
Fullwidth/Halfwidth Unicode Decoding	Protects against attackers that use the uncode decoding method, which is the method that is used to attack web servers. If an HTTP content scanning system fails to properly scan full-width/half-width Unicode encoded traffic, malicious HTTP traffic may bypass content scanning systems. **Defaults**: Enabled; Severity - High; Action - Block.
HTTP Response Splitting Vulnerability	Identifies and blocks response splitting, which is a method the attacker uses to control the response sent to the client. **Defaults**: Enabled; Severity - High; Action - Block.
IE Discussion Bar-Access to Internal Information	Provides protection to the vulnerabilities in the Internet Authentication Service (IAS) components. **Defaults**: Enabled; Severity - Low; Action - None.
MSSQL Data Retrieval with Conversion Errors	Defines how to use servers behind MSSQL, which might cause conversion errors and protects against data leakage that uses vulnerabilities of MSSQL. **Defaults**: Enabled; Severity - High; Action - Block.
Malformed HTTP Attack (Non compatible HTTP Results Error code)	Correlates between the HTTP protocol validation rules and the error code retuned from the server, if it happened more than the threshold from the same IP. **Defaults**: Enabled; Severity - High; Action - None.
OS Command Injection	Protects in the cases when the attacker inserts OS commands using web requests. **Defaults**: Enabled; Severity - High; Action - Block.
Plain Vanilla Scanner Detection	Prevents obvious scanning attempts done by the scanners that are not trying to hide. **Defaults**: Enabled; Severity - High; Action - Block.
Privacy Violation - Credit Card Number Insertion	Indicates that there is an insertion of a credit card number to the Parameter value, which might violate the privacy policy of the web application. **Defaults**: Enabled; Severity - High; Action - Block.

Predefined Policy	Description
Privacy Violation - Credit Card Number Insertion by Internal IP Address	Indicates that an insertion of a credit card number to the Parameter value is performed by an internal IP. This might violate the privacy policy of the web application. **Defaults**: Enabled; Severity - High; Action - Block.
Privacy Violation - Credit Card Number Insertion by non Internal IP Address	Indicates that an insertion of a credit card number to the Parameter value is performed by the external IP. This might violate the privacy policy of the web application. **Defaults**: Enabled; Severity - High; Action - Block.
Sensitive Error Messages Leakage	Alerts on error messages in responses. This might indicate web server/database failure. **Defaults**: Enabled; Severity - High; Action - Block.
Suspected parameter tampering - Deprecated	**Defaults**: Enabled; Severity - High; Action - None.
Suspicious Response Code	Identifies suspicious response code. It is used to create a violation for activity that should be excluded from the profile. **Defaults**: Enabled; Severity - Medium; Action - None.
Unsuccessful Directory Browsing	Prevents the case in which users get an error message and using this message they can identify the web server. Alerts when Directory Browsing occurs. **Defaults**: Enabled; Severity - Low; Action - None.
WEB-FRONTPAGE	The following predefined policies are available in order to identify well known web front page vulnerabilities: • **WEB-FRONTPAGE- Access to Internal Information** (**Defaults**: Enabled; Severity - Low; Action - None). • **WEB-FRONTPAGE- External Access to Internal Information** (**Defaults**: Enabled; Severity - Low; Action - None). • **WEB-FRONTPAGE-Access to Sensitive Internal Information** (**Defaults**: Enabled; Severity - High; Action - Block).
Webdav Method Detection	Identifies attacks based on the WebDAV method, which means attacking the sites that use the WebDAV protocol that allows users to collaboratively edit and manage files on remote web servers. **Defaults**: Enabled; Severity - Medium; Action - None.
eMail Hoarding	Protects against automatic agents that search sites to create databases of spam e-mail lists. **Defaults**: Enabled; Severity - High; Action - Block.

Web Service Correlated Validation Policy

The protection provided by the Web Service Correlated Validation policy is based on Correlated Attack Validation (CAV) technology, see Correlated Attack Validation (CAV) Used in Correlation Policies on page 170.

The following table presents the rules included in the Web Service Correlated Validation policy.

Web Service Correlated Validation Rules

Rule	Description
Cross-site scripting	Cross-site scripting ('XSS' or 'CSS') is an attack that takes advantage of a Web site vulnerability in which the site displays content that includes un-sanitized user-provided data. To accurately detect cross-site scripting attacks the product must know where and when to look for that signature. Most cross-site scripting attacks occur either with error pages or with parameter values. Therefore the product needs to look for cross-site scripting signatures either within parameter values or within requests that return error messages. To look for signatures in parameters values the product must parse the URL correctly and retrieve the value part and then search for the signature on the value while overcoming encoding issues. To look for signatures in pages that return error messages the product needs to know that the specific URL returned an error code. Intrusion Detection and Prevention Systems which are not Web application oriented simply do not implement these very advanced capabilities. The detection process includes checking of the URLs and parameters and it contains the following steps: • **Normalization**: Checking the input values in the incoming HTTP traffic and deleting additional comments and symbols that may be used as evasion techniques. • **Detection**: Correlation between various detection methods, as follows: • Checking for application profile violations. • Match on characters. • Match on the JavaScript keywords. • Match on regular expressions. **Defaults**: Enabled; Severity - High; Action - Block.
Forceful Browsing	Using forceful browsing, attackers may gain access to restricted parts in the Web server directory. This kind of attack occurs when the attacker "forces" a URL by accessing it directly instead of following links. A client is violating the normal flow of an application. This alert indicates that a user is manually typing a URL in the browser in order to bypass the normal flow between different pages in an application. The aim of an attacker could be an unexposed file on the server or unauthenticated access to sensitive parts of an application. **Defaults**: Disabled; Severity - Medium; Action - None.

Rule	Description
SQL injection	SQL injection is a technique used to take advantage of non-validated input vulnerabilities to pass SQL commands through a Web application for execution by a backend database. To protect against SQL injection SecureSphere combines various techniques. Application layer signatures are immediately effective against the common attack vectors. In addition to the signatures, SecureSphere observes the SQL communication and builds a profile consisting of all allowed SQL queries. Whenever an SQL injection attack occurs, SecureSphere can detect the unauthorized query sent to the database. SecureSphere can also correlate anomalies on the SQL stream with anomalies on the HTTP stream to accurately detect SQL injection attacks and automatically build and enforce security polices. Imperva has developed a scoring-based algorithm to augment the immediate protection offered without adversely affecting performance. This algorithm is a defense layer that comes just before (below) the profile layer of the security engine. To deal with evasion technics and to avoid false positives, SecureSphere performs deep analysis of the string and prevents all the threats before it arrives to the database server. The detection process includes checking of the parameters and it contains the following steps: • **Normalization**: Checking the SQL commands in the incoming HTTP traffic and deleting additional comments and symbols that may be used as evasion techniques. • **Detection**: Correlation between various detection methods, as follows: • Checking for application profile violations. • Match on characters. • Match on keywords defined in stored procedures, see Configuring Stored Procedure Support. • Match on regular expressions, in case the score results are beyond the predefined threshold value. **Defaults**: Enabled; Severity - High; Action - Block.
Session Attribute Changes	SecureSphere detects session attribute changes by looking for mismatch between the original request for application and other requests within the framework of the same session. This is achieved by comparing the parameters in the first request to the parameters in other requests to see if all the requests are sent by the same user. You can select the exact location checked by SecureSphere, as follows: • Accept Encoding: Checks the Accept Encoding request header. • Accept Language: Checks the Accept Language request header. • Enforce Empty Accept Encoding • Enforce Empty Accept Language • Ignore Static Files: Ignores the static files during the search process. For more information about the static files, see • Source IP • User Agent **Defaults**: Disabled; Severity - Informative; Action - None.
Too Many of the Same Response Code	SecureSphere counts the number of HTTP responses with a 500, 404, 403, 400, 304, 302 and 200 response codes. The time period during which SecureSphere counts the responses is 5 minutes. If the average number of responses for any of these sources exceeds the policy limit, SecureSphere generates this violation. **Defaults**: Disabled; Severity - Low; Action - None.

Securing Web Applications

SecureSphere web application security policies are applied to the application level traffic and provide content-based protection. The protection model includes application level signatures, dynamic profile, web worm protection and application level web custom policies. The following policy types are available for securing web applications in SecureSphere:

- ThreatRadar Fraud Prevention Services on page 206
- Anti-Automation Policies on page 206
- Web Profile and Web Profile Policy on page 208
- Web Worm Policy on page 209
- Web Application Signatures Policy on page 209
- Web Application Custom Policies on page 210
- OCSP Protocol Validation Policies on page 212

ThreatRadar Fraud Prevention Services

ThreatRadar Fraud Prevention Services is a platform used together with third party vendors to prevent fraud and reduce risk to your network and data. Fraud Prevention Services integrates with your organization's web applications protect them without requiring code changes to these applications. Once a user's status has been determined, it is stored and can be used at any time during the session to impose security policies. For example, in a banking application, Trusteer Pinpoint can be used upon login to determine if a visitor's device is compromised with malware, and the results can be used to determine how to handle the user if they try to conduct sensitive transactions, such as a wire transfer.



Fraud Prevention features are only supported on 64 bit SecureSphere platforms.

How it Works

SecureSphere coordinates with a Fraud Prevention Service provider. When a user tries to connect to your company's network or access a website (for example, an online banking login page), SecureSphere instructs the Fraud Prevention Service to conduct its tests using Java script and cookies. Fraud Prevention Services then informs SecureSphere as to the status of the session trying to connect to your network. You can optionally configure a security custom policy, which then tells SecureSphere how to handle user requests based on this status.

The status of the Fraud Prevention Service scan appears in the violation details in the Monitoring window if the user violated a security policy of any type.

To view the list of the available service providers, refer to the ThreatRadar Dashboard.

Anti-Automation Policies

Anti-Automation policies detect automated attacks against web servers and then mitigate these attacks by either by blocking access to the servers or alerting about them. While each policy protects against a specific threat, applying a combination of these policies can provide an increased level of protection.

The various policies that are available for protecting against automated attacks include:

- **Bot Mitigation**: Identifies bot attacks meant to cause Denial of Service or other automated attacks. For more information see Bot Mitigation Policy on page 207
- **Anti Scraping**: Identifies attempts to "scrape" information from your web site, often used to gain business intelligence and pricing information. For more information see Anti Scraping Policy on page 208

Bot Mitigation Policy

Automated robots, or bots as they are called, are used for both regular operation such as crawling web sites for new information and for malicious purposes such as disrupting the operation of a website through Denial of Service, and harvesting e-mail addresses for spammers.

The Bot Mitigation Policy uses guidelines to determine if the origin of a session is a valid user or an automated bot. If the traffic meets the guidelines that determine its an automated bot whose purpose may be malicious, then the policy is activated.

Two items need to be supported by end user browsers otherwise they will not be allowed to access web sites protected by Bot Mitigation policies.

- JavaScript
- Cookies

Without these capabilities, end users receive a message indicating that these capabilities are mandatory to view the requested page. The message displayed is configurable and enables you to customize it with contact details for your organization.



Notes:

- Bot Mitigation policies offer one action, block.
- A Bot Mitigation policy block action does not support long or short IPs.
- This policy has no effect when the SecureSphere gateway is configured in Sniffing mode or the SecureSphere Server Group's operation mode is set to Simulation.
- The Bot Mitigation policy by default is configured to work on .htm and .html file types. If you want it to work on other pages, it is recommended that you configure the policy by removing the File Types Criteria and using the URLs on which you would like to have the policy activated in the URL Prefix criteria.

EXAMPLE: BOT MITIGATION EXAMPLE

The objective of configuring a Bot Mitigation policy is to determine guidelines that will at the same time allow authentic traffic to pass unhindered but also identify suspicious traffic and block it. For this purpose you use the Bot Mitigation policy.

To protect against a bot attack:

1. In the **Main** workspace, select **Policies > Security**, in the **Policies** pane, click . The **Create New Policy** dialog box appears.
2. Type the name of the new policy, from the **Type** drop-down box, select Bot Mitigation and click Create.
3. In the **Match Criteria** tab, select a **Severity**, choose a **Followed Action** if desired.
4. In the **Bot Mitigation Settings**, modify the message text as required. It is recommended that you provide an actual phone number that customers can call in order to best mitigate situations in which valid customers are prevented from accessing the system and can obtain support.
5. In the **Match Criteria** tab, expand the **Source IP Addresses** match criteria and select IP groups you want to be excluded from monitoring as part of the Bot Mitigation policy. Typically this would include all the major search engines whose operation include using crawlers on your website for legitimate reasons. You'd configure the Operation to be set as Exclude.
6. In the **Match Criteria** tab, expand the **File Types** match criteria, then add rows for the different types of pages you want to run the policy on. For example, if your web site uses .htm and .html policies, then add a row for each of these items and type ".htm" and ".html".
7. In the **Match Criteria** tab, expand the **URL Prefix** match criteria, then add rows for the specific pages you want to protect. For example, you may want to add the URL of your company's login page.
8. Click the **Apply to** tab and apply the policy to the relevant infrastructure.
9. Click **Save**. The policy has been saved and applied to the selected locations.

Anti Scraping Policy

Scraping is in simple terms the extracting of information from a web site. This information can then be used for a variety of purposes, such as business intelligence. For example, a large retailer offers information on its pricing online. The company has noticed that its competition is offering slightly lower rates that are highly correlated to their price. Further investigation reveals that the competition is "scraping" their web application and extracting all pricing information. Another example might be job web sites that are pay per use. This information is used in the competition pricing strategy and works for their advantage.

In addition to standard policy parameters including Action, Followed Action and Severity, the Anti Scraping policy uses a number of additional parameters that can be configured, and which correlate with each other to determine when scraping is taking place. These items include:

- **Context**: of the connection. Whether a single IP Address, Session or User
- **Period of time**: in seconds, and
- **Number of URLs**: unique URLs which are being accessed within this context and time frame

  If all of these thresholds are met, the policy is violated and an alert is generated.

EXAMPLE: ANTISCRAPING EXAMPLE

The objective of configuring an Anti Scraping policy is to determine what constitutes a scraping event. For this purpose you use the Anti Scraping policy.

To protect against scraping:

1. In the **Main** workspace, select **Policies > Security**, in the **Policies** pane, click . The **Create New Policy** dialog box appears.
2. Type the name of the new policy, from the **Type** drop-down box, select Anti Scraping and click Create.
3. In the **Match Criteria** tab, select an **Action**, either Block or None, select a **Severity**, choose a **Followed Action** if desired, and select the **Enabled** checkbox.
4. In the **Anti Scraping Detection Settings**, review the parameters to see if they match your requirements, or modify them as needed. When all three conditions are met, SecureSphere defines it as a scraping attack.
   - Number of Unique URLs: Determines the number of unique URLs that are being accessed on your web server within the context and period of time configured.
   - Context in which URLs are detected: Determines the context environment in which URLs are looked at. Options include IP address, a specific User, or a specific session.
   - **Period (in seconds)**: Determines the period of time in which the above two parameters need to be encountered (unique URLs and context).
5. In the **Match Criteria** tab, expand the **Source IP Addresses** match criteria and select IP groups you want to be excluded from monitoring as part of the Anti Scraping policy. Typically this would include all the major search engines whose operation include using crawlers on your website for legitimate reasons. You'd configure the Operation to be set as Exclude.

   Note: Alternatively you can use the Match Criteria (URL Prefix) to protect or exclude specific URLs.
6. Click the **Apply to** tab and apply the policy to the relevant infrastructure.
7. Click **Save**. The policy has been saved and applied to the selected locations.

Web Profile and Web Profile Policy

A Profile contains application profile definitions learned by SecureSphere. By comparing HTTP requests/responses to the profile, SecureSphere can detect abnormal behavior and prevent malicious activity with pinpoint precision.

Each Web application has a single Web profile and profile policy associated with it, which is applied only to the profile URLs and cookies in the Protect mode.

For more information on Profile policy, refer to Web Profile Policy on page 275.

Web Worm Policy

A Web worm policy blocks Web access to the standard Web application folders, which are different for each Web server, such as Apache, IIS and so on, on the assumption that legitimate Web users have no valid need to access these folders.

A web worm utilizes a web server vulnerability to spread to a large amount of web servers in a short period of time. Some web worms utilize well-known vulnerabilities. These worms can be easily stopped using the signature layer, as a signature for that attack probably already exists. The real problem is with worms that utilize unknown vulnerabilities, i.e., vulnerabilities that were not published prior to the worm outbreak and thus no signature exists at the time of outbreak. The SecureSphere web worms protection layer was implemented for this type of worms. The web worms mechanism relies on SecureSphere's ability to build a profile of allowed URLs on each web server.

The assumption is that web worms spread by sending a URL. The vulnerabilities used by the worm should exist on a large number of web servers, for example all IIS 6 servers, in order for the worm to spread massively. Thus the worm must use a URL which exists on many web servers. Only default URLs (URLs which exist by default when you install a web server or a common application on a web server) meet that criteria.

Using its learning and profiling capabilities, SecureSphere automatically learns the names of all default files which are actually being accessed by users on the protected web server. A pre-configured list of directories (e.g. /, /scripts/, /cgi-bin/) instructs SecureSphere where default files are usually located. Once the profile is ready, SecureSphere blocks any attempt to access a non-profiled URL on a default directory. This way, if for example a new worm uses an unknown vulnerability by sending an HTTP request for a URL to /scripts/page.aspx and this URL is not part of the learned profile, SecureSphere blocks the request, thus blocking the worm itself.

Before blocking, SecureSphere verifies that the request has no legitimate host value and no legitimate session ID.

The rate of false positives (i.e., blocking legitimate URLs which are part of the application) is very low as this feature only works on default directories, which are rarely used by web applications to store files.

The Web Worm policy is applied to web applications by default. Each web application can have exactly one Web Worm policy associated with it.

The rule is enabled by default and its default action is Block and default severity is High.

Susceptible Directory Group

The standard Web application folders protected by SecureSphere are called **susceptible directories**. The protection is based on learning of their normal behavior and then finding the deviations from it.

These directories are usually stored in the same folders on each operating system, for example Apache default directories.

You can define a **Susceptible Directory Group** in **Global Objects**, see Configuring Susceptible Directories on page 426, then select this group in the Web Worm policy and apply it to the particular applications.

Web Application Signatures Policy

The Web Application Signatures policy enables you to apply content-based signatures to the application level, see Intrusion Prevention Used in Signature Policies on page 170.

There are no default policies of that type provided by SecureSphere. You can set user-defined policies, see Cloning/Creating New Policies on page 179.

Web Application Custom Policies

Web application custom policies provide you with the ability to combine all the protection levels and to define very accurate and high level protection, see Correlated Attack Validation (CAV) Used in Correlation Policies on page 170. These policies are defined using flexible Match Criteria mechanism, which is also used on the service level.

EXAMPLE: PREVENTING BRUTE FORCE ATTACK ON APPLICATIVE AUTHENTICATION

This example outlines a correlation rule that detects and blocks brute force attempts to attack the applicative authentication mechanism. This is done by detecting many login failures in a short period of time from the same Source IP or session.

To protect against the Brute Force attack:

1. In the main window, select **Policies > Security**, in the Policies pane, click  and select WEB Application. The **Create New Policy** dialog box appears.
2. Type the name of the new policy, from the **Type** drop-down box, select WEB Application Custom and click Create. For example, type Brute force attack on Applicative Authentication.
3. In the **Match Criteria** tab, click on the green arrow next to **Authentication Result**.
4. To perform match on the number of the legitimate logins according to the profile definitions, expand the **Authentication Result** match criteria and select **failed** from the drop-down box.
5. To define match over time, click on the green arrow next to **Occurrence**.
6. Expand the **Occurrence** match criteria and define the parameters in the table **Brute Force Occurrence Parameters** below.
7. Define the reaction settings, see Modifying Policies in the Security Window on page 176.
8. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Brute Force Occurrence Parameters

Option	Description
Occurred more than	Set the maximum number of login attempts that is acceptable in your web site.
Within	Set the period of time in seconds in which the login attempts are considered to be legitimate attempts. For example, if you know that normally you have not more than 5 attempts per user with in 2 minutes, then you set 5 and 120.
In a context of a single	The number of occurrences within the defined period of time must be a part of the same session. Select **Originating Session/Source IP**.

EXAMPLE: PREVENTING UNAUTHORIZED USER ACCESS TO CERTAIN SITE LOCATIONS

You can define a policy that block traffic destined to particular parts of your web site, so that only privileged users can get there.

To prevent unauthorized user access:

1. In the main window, select **Policies > Security**, in the Policies pane, click  and select WEB Application. The **Create New Policy** dialog box appears.
2. Type the name of the new policy, from the **Type** drop-down box, select WEB Application Custom and click Create.
3. In the **Match Criteria** tab, click on the green arrow next to **Authenticated Session**, expand it and select **No**.
4. Click on the green arrow next to **URL Prefix**.

5. Expand the URL Prefix match criteria and set the following parameters:
   - **Operation**: Select **At least one**.
   - **Value**: Click  and type `/data`.
6. From the **Action** drop-down box, select **Block**.
7. From the **Severity** drop-down box, select **High**.
8. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

EXAMPLE: PREVENTING DATA LEAKAGE IN WEB RESPONSES

You can configure policies that prevent data leakage and sensitive data exposure. First you need to set the dictionary groups and then you can configure the policies. During policy configuration, you need to define match criteria with the exact location where you want to look for the predefined patterns. This optimizes the search process and saves system resources, as the search is applied only to the relevant traffic or relevant part in the web response.

To prevent data leakage in web responses:

1. Define a **Sensitive Data Dictionary** group in **Global Objects**, see Configuring Sensitive Data Protection on page 350.
2. In the main window, select **Policies > Security**, in the Policies pane, click  and select WEB Application. The **Create New Policy** dialog box appears.
3. Type the name of the new policy, from the **Type** drop-down box, select WEB Application Custom and click Create.
4. In the **Match Criteria** tab, click on the green arrow next to URL Prefix and type the value(s) of the URL prefix that you want to use for the search.
5. Click on the green arrow next to **Sensitive Dictionary Search** and set its parameters as shown in the table **Sensitive Dictionary Search Parameters** below.
6. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Sensitive Dictionary Search Parameters

Option	Description
Search Mode	The following modes are available: • **Exact**: To search for the exact pattern as it is defined by the dictionary. • **Contains**: To search for the pattern that includes the pattern defined by the dictionary.
Dictionaries	Select the dictionaries that contain the patterns that you want to participate in the detection process.
Locations	Select one of the following locations: • **URL**: Request URLs. • **Parameters**: Parameters in the request. • **Headers**: HTTP headers. • **Response Content**: content returned by the web server. • **Non Normalized URL**: the raw URL, as it is received in the actual request.

OCSP Protocol Validation Policies

The OCSP protocol Validation policy is automatically created when the Parse OCSP requests option is enabled on an HTTP/S web application in the site tree (Definitions tab)

- For more information about the Parse OCSP request option, see Web Application Definitions Parameters on page 71.
- For an in-depth description of configuring support for OCSP, see Configuring OCSP Support on page 424.

OCSP Protocol Policy Rules

Rule	Description
Invalid ASN.1 Request Encoding	Validates Distinguished Encoding Rules (DER) encoding of OCSP message (the message transfer syntax specified by the ITU in X.690).
Invalid Base64 Request Encoding	Validate base64 encoding.
Invalid Message Content Type	Checks that content type of POST request is set to application/ocsp-request.
Invalid Request Structure	Validate that the request structure complies with ASN.1 Definition. Validate the type for all OCSP entities that are strictly defined in the ASN.1 Definition.
Invalid version	Validates the version supported. Enables you to configure a range of versions which need to be supported.
OCSP Message Too Long	Verifies that the length of the message in GET requests does not exceed 255 characters, and that the length of the message in POST requests does not exceed the specified value (default 12288).
Too many queries within a Request	Checks that the number of certificate queries within a request does not exceed the specified value (default 128).

CHAPTER 8

Reporting

SecureSphere includes a robust reporting mechanism that enables you to produce pre-defined or user-defined reports based on accumulated data that can be generated either automatically or on-the-fly, or schedule to run a regular intervals, then distributed as required.



Note: For information on archiving Reports, see the Admin Guide or speak to your SecureSphere administrator.

This section describes how to work with SecureSphere reports, and includes the following topics:

Reports Introduction 213
Report Terminology 214
Understanding the Reports Window 214
Working with Reports 215
Creating and Managing Reports 226
Filtering Reports 238

Reports Introduction

SecureSphere accumulates a wealth of data which can provide transparency regarding various aspects of your data, databases, file servers and users in your network. This information can be used to report on activity to assists in maintaining compliance with various regulatory bodies, distribute regular reports on system status, create regular maintenance reports that list SecureSphere alert activity, enable the monitoring of system status, and much more.



Note: For an overview of terminology used with reports, see Report Terminology on page 214.

Reports can be created in a number of ways in SecureSphere including:

- **Scheduled for maintenance or compliance**: Reports can be scheduled for automatic generation on a regular basis, contributing to a workflow where information is reported to provide status information and to comply with various regulatory bodies.
- **On-the-fly**: Reports can be customized and created upon request to provide information for immediate needs.

Report Terminology

There are a number of terms used for various SecureSphere report components. The following table provides a list of definitions.

Report Definitions

Name	Definition
Report	Basic framework of a report including layout, tables, and general definitions without actual data. Reports are listed in the Report Pane of the Manage Reports window.
Result	An instance of data returned as the result of running a report. Results are viewable from the View Results window and from the Results tab of the configuration pane.
Enabled Report	A report that has been made available to users by the SecureSphere administrator. Reports are enabled by selecting the enable checkbox in the relevant report's General Details tab.
Pre-defined Report	A report that has a fixed layout whose Tabular and Data Analysis views cannot be modified. Most pre-defined reports can be used to create a new report by choosing to create a new report, and basing the new report on the pre-defined report.

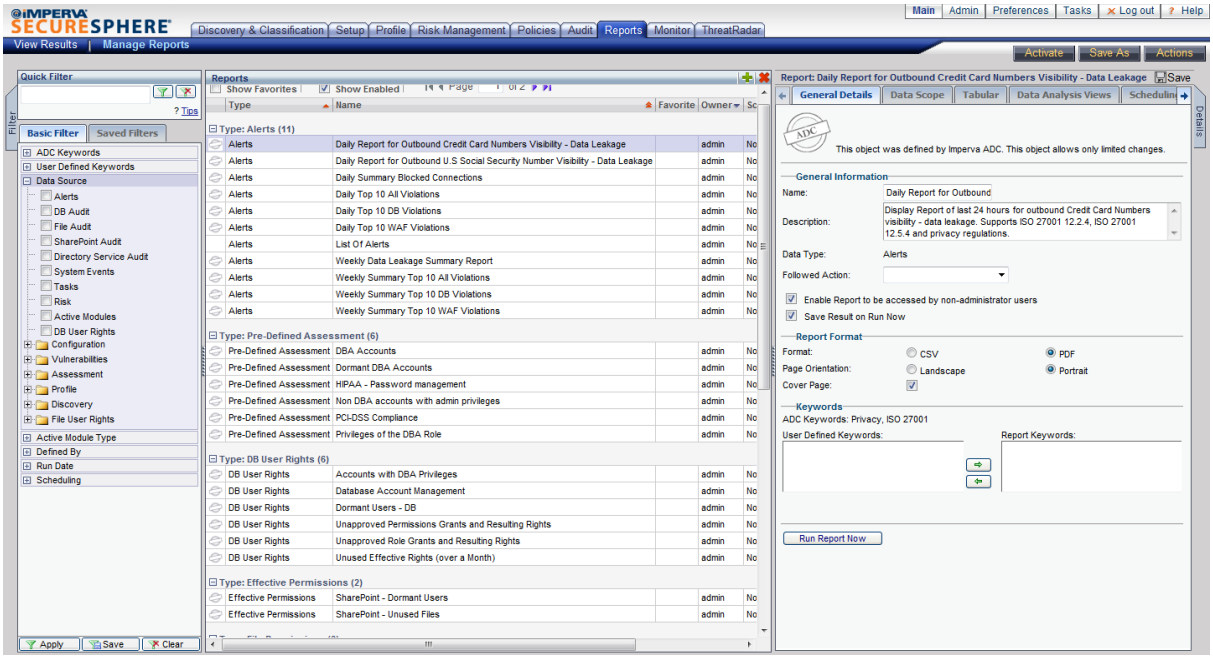
Understanding the Reports Window

The Reports window provides a wide selection of options that enable you to select pre-defined reports and configure customized reports for use with a variety of applications. The Reports window provides two main views, as represented by links in the Reports navigation bar:

- **View Results**: Displays reports that have been generated and saved. This includes scheduled reports and reports manually run that were configured with the Save Results on Run Now option. For more information on viewing results, see Viewing Generated Reports on page 237.
- **Manage Reports**: Enables you to manage, customize and run reports. A number of report categories offers hundreds of reports you can use.

To view the Manage Reports window:

1. In the **Main** workspace, select **Reports**. Then click **Manage Reports**. The **Manage Reports** window appears.



The Manage Reports window consists of the following items:

- **Filter Pane**: Enables you to filter items using the quick or basic filter. For instructions on how to filter reports, see Filtering Reports on page 238.
- **Display Options**: Enables you to configure what reports to display using the following tools:
  - **Show Favorites**: Shows reports that have been marked as a favorite. For further information, see Managing Favorite Reports on page 229.
  - **Show Enabled**: Shows reports that have been configured as enabled on the General Details tab. Only available if logged on as Administrator. For further information, see Configuring General Report Details on page 229.
  - **Arrow buttons**: Enables you to scroll through the list of available reports.
- **Report Pane**: Lists reports that can be used to generate results. For information regarding reports in the Report pane, see Understanding Report Types on page 216.
- **Configuration Pane**: Enables you to configure the various aspects of a report. For further information, see Configuring Reports on page 228.



Note: To sort reports in the report pane, click  on the heading you want to sort by. The arrow direction indicates whether the sort order is ascending or descending based on the selected column.

Working with Reports

This section reviews the various aspects of working with reports. It reviews the following subjects:

Understanding Report Types 216
Working with Alert Reports 218
Working with System Event Reports 219
Working with Task Reports 219

Working with Configuration Reports 220
Working with Vulnerability Reports 221
Working with Pre-Defined Vulnerability Reports 221
Working with Discovery Reports 222
Working with Profile Reports 223

Understanding Report Types

SecureSphere comes with a library of pre-defined reports for a wide variety of uses. These reports can be used as is. Many can be modified to match specific needs by changing various aspects of their configuration as described in Configuring Reports on page 228.

To view available reports:

1. In the Main workspace, click Reports. The Reports window appears.
2. In the Reports window, click Manage Reports. The Manage Report window appears.
3. Deselect the Show Favorites and Show Enabled (if present) check boxes. The complete list of available reports is displayed.

   Report definitions fall into the main categories that reflect a report's data source, and are displayed when creating a new report as described in Creating a Report on page 227. The following table lists the main report types available in SecureSphere

SecureSphere Report Types

Name	Description
System Events	Displays a list of system events collected in SecureSphere. For more information see Working with System Event Reports on page 219.
Tasks	Reports on tasks that have been created in SecureSphere and their details. For more information see Working with Task Reports on page 219.
Configuration	Reports information on the SecureSphere site configuration, security and audit policies that are configured in the system. For more information, see Working with Configuration Reports on page 220.
Alerts	Reports alert information based on data collected in the Alerts table. Alert reports can be used to report on events that have taken place in traffic and can be configured to report on a specific time frame to assist in forensics. For further information see Working with Alert Reports on page 218.
Pre-Defined Assessment	Displays a list of SecureSphere assessment tests that have been conducted in your network. This report is pre-defined. Its tabular and layout options cannot be modified. For more information see Working with Pre-Defined Assessment Reports.
Assessment	Displays a list of SecureSphere assessment tests that have been conducted in your network. This report can be customized with tabular and display options. These provide decision makers with the tools needed to guide SecureSphere configuration based on the current conditions of your network and data. For more information see Working with Assessment Reports.
Pre-Defined Vulnerabilities	Displays a list of vulnerabilities that have been detected through assessment tests. This report is pre-defined. Its tabular and layout options cannot be modified. For more information see Working with Pre-Defined Vulnerability Reports on page 221.
Vulnerability	Displays a list of vulnerabilities that have been detected through assessment tests. This report can be customized with tabular and display options. For more information, see Working with Vulnerability Reports on page 221.
Active Modules	Active module reports enable access to data generated and collected by active modules. For more information see Working with Active Module Reports.
Profile	Profile reports provide visibility into SecureSphere dynamically learned profiles: • **Web Profile Reports**: Assist in understanding data that has been profiled related to the web that include URLs, cookies, learned hosts, susceptible directories and web application users. • **DB Profile Reports**: Assist in understanding data that has been profiled in a data base that include database users and database user groups. For more information see Working with Profile Reports on page 223.
Discovery	A set of different reports that enables you to generate various information regarding service and data discovery. For more information see Working with Discovery Reports on page 222.

Working with Alert Reports

Alert reports are based on information collected in the Alerts window. Alert reports can be used to report on events that have taken place in traffic and can be configured to report on a specific time frame to assist in forensics. SecureSphere comes with a number of pre-defined reports that display data about alerts.



Note: Alert Reports can integrate both a data scope to filter data in an alert, and tabular columns.

- For a list of fields available for Report Data scopes see Alert and Violation Criteria on page 575.
- For a list of columns available for reports, see Columns Available in Alert Reports on page 611.

When the Tabular view is used in an alert report, the way resulting data is displayed may be affected by the type of data being displayed. If a column included in the report represents non-event data, then each value in the column is displayed in a single row. However, if the column includes data that is based on an event (for example, a session), and that event was experienced a number of times (such as part of an aggregated alert), an additional row is added to the report for each instance of the event.

For example: Lets say you configure a report on alerts using Tabular view. This report includes alerts that were part of SQL injection and Cross Site Scripting attacks to port 80. If you add the Session ID field as a column in the report, then an additional row is added to the report for each instance of SQL injection or Cross Site Scripting on port 80 (since session numbers are different for each event, and cannot be aggregated).

For example, you may want to list all alerts triggered by known web attack signatures. For a complete list of all available report definitions, see System Events Reports on page 645.

EXAMPLE: WEB SIGNATURE VIOLATIONS (LIST OF ALERTS REPORT)

This example describes how to generate a report that gives a list of all violations triggered by recognized HTTP signature attacks. This report is based on the List of Alerts pre-defined report. It arranges each alert with its aggregated violations.



Note: The List of Alerts report is pre-defined, meaning certain elements cannot be modified. The Tabular and Data Analysis tabs are not available.

1. In the **Main** workspace, select **Reports**. The **View Results** window appears displaying reports that have recently been generated.
2. Click **Manage Reports**. The **Report Management** window appears displaying available report definitions and configuration options.
3. In the **Reports** pane of the **Report Management** window, click **New** , then select **Alerts**. The **Create Alerts Report** dialog box appears.
4. Under **Name**, type **List of Web Signature Violations**.
5. Select **Use Existing**, then select **List of Alerts** from the dropdown menu.
6. Click **Create**, the report is created and appears in the **Reports** pane.
7. In the **Data Scope** tab, add the **Violation** field by clicking its green arrow . The **Violation** field is moved up to the **Enabled Fields** pane.
8. Click the plus sign (**+**) next to the field. Its options are displayed.
9. In **the Violation** field, add **HTTP Signature Violation** and set it to **Equals**.
10. In the **Data Scope** tab, add the **Last Few Days** field by clicking its green arrow . The **Last Few Days** field is moved up to the **Enabled Fields** pane.
11. Click the plus sign (**+**) next to the field. Its options are displayed.
12. In the **Last Few Days** field, type 7. This displays violations triggered by recognized HTTP signature attacks in the last 7 days.
13. From the upper right-hand side of the window, click **Save**.

14. From the **General Details** tab, click **Run Report Now**. The report is run. Once complete a dialog box opens enabling you to open the generated report.



Note: The List of Alerts report can additionally be created directly from the **Alert** window by selecting the action button **Save As** from the details pane and selecting PDF. The alerts filter is used to determine the report data scope.

Working with System Event Reports

System Event reports lists system events collected in SecureSphere. This can be helpful in analyzing different aspects of operation such as users that have logged in, or failed logins.



Note: System Event Reports can integrate both a data scope to filter data in an alert, and tabular columns.

- For a list of fields available for Report Data scopes see System Event Criteria on page 579.
- For a list of columns available for reports, see Columns Available in System Event Reports on page 616.

EXAMPLE: REPORT OF LOGIN SYSTEM EVENTS

This example describes how to generate a report that lists login events.

1. Create a **System Events** report as described in Creating a Report on page 227.
2. Select the report you just created from the **Reports** pane. Its details are displayed in the **Configuration** pane.
3. In the **Data Scope** tab, add the **Message** field by clicking its green arrow . The **Message** field is moved up to the **Enabled Fields** pane.
4. Click the plus sign (**+**) next to the field. Its options are displayed.
5. In the **Message** field, click  **New**, then type the message **User%logged in from**.
6. Repeat step 5, this time typing the message.
7. In the **Data Scope** tab, add the **Subsystem** field by clicking its green arrow . The **Subsystem** field is moved up to the **Enabled Fields** pane.
8. Click the plus sign (**+**) next to the field. Its options are displayed.
9. In the **Subsystem** field, add **User**. This displays the item that generated the system event.
10. In the **Tabular** tab, add the following columns:
    - Create Time
    - Message Time
11. In the upper right-hand side of the window, click **Save**.
12. In the **General Details** tab, click Run **Report Now**. The report is run. Once complete a dialog box opens enabling you to open the generated report.

Working with Task Reports

Task reports list task that have been configured and details regarding their execution. This can assist in fulfilling requirements for compliance, understanding open issues which may need to be addressed, and more.

EXAMPLE: REPORTING ON OPEN TASKS

This example describes how to generate a report that lists open tasks.

1. Create a **Tasks** report as described in Creating a Report on page 227.
2. Select the report you just created from the **Reports** pane. Its details are displayed in the **Configuration** pane.
3. In the **Data Scope** tab, add the **Status** field by clicking its green arrow . The **Status** field is moved up to the **Enabled Fields** pane.
4. Click the plus sign (**+**) next to the field. Its options are displayed.
5. In the **Status** field, add the **In Progress**, **New** and **Rejected** options.
6. In the **Data Scope** tab, add the **Priority** field by clicking its green arrow . The **Priority** field is moved up to the **Enabled Fields** pane.
7. Click the plus sign (**+**) next to the field. Its options are displayed.
8. In the **Priority** field, add **Urgent** and **High**.
9. In the **Tabular** tab, add the following columns:
   - Task Name
   - Current Status
   - Task Assignee
10. Under **Sorting**, select **Current Status**, and under **Sort**, select **Descending.**



Note: You can also configure an aggregation function for the columns. For more information on aggregation functions, see Understanding Aggregation Functions

11. In the upper right-hand side of the window, click **Save**.
12. In the **General Details** tab, click Run **Report Now**. The report is run. Once complete a dialog box opens enabling you to open the generated report.

Working with Configuration Reports

Configuration reports enable you to report on various aspects regarding SecureSphere configuration, and can be generated to display data on SecureSphere agents currently configured and operating in the network, and overall Site configuration, security or audit policy configuration. This can assist in obtaining a general overview of configuration such as existing agents and their status, a list of IPs and ports being protected by the site and their relevant gateway groups, and more.

Configuration Reports Available in SecureSphere

Report Type	Description
Sites	Reports information regarding the various aspects of a SecureSphere site and its configuration including server groups, applications, IPs being protect, and more.
Security Policies	Reports information about security policies configured on SecureSphere including policy names and types, information about rules used in the policies, creation time, and more.

EXAMPLE: REPORTING ON SECURESPHERE SITE CONFIGURATION

This example describes how to generate a report about SecureSphere site configuration.

1. Create a **Configuration > Sites** report as described in Creating a Report on page 227.
2. Select the report you just created from the **Reports** pane. Its details are displayed in the **Configuration** pane.

3. In the **Data Scope** tab, add the **Server IP Range** field by clicking its green arrow . The **Server IP Range** field is moved up to the **Enabled Fields** pane.
4. Click the plus sign (**+**) next to the field. Its options are displayed.
5. Configure a **From** and **To** range that represents a range of IPs monitored by SecureSphere.
6. In the **Tabular** tab, enable the **Tabular** view, then add the following columns:
   - Site
   - Server IP
   - Site Creation Time
   - Protecting Gateway Group
   - Hostname
   - Is Protected
7. From the upper right-hand side of the window, click **Save**.

From the **General Details** tab (or Action button), click **Run Report Now**. The report is run. Once complete a dialog box opens enabling you to open the generated report.

Working with Vulnerability Reports

Vulnerability Reports generate a list of vulnerabilities that have been detected in your network as the result of assessment tests. Vulnerability reports can be configured to display tabular information in addition to Data Analysis Views including charts.

Vulnerability Reports can assist in generating regular reports on open vulnerabilities. For example, you may want to generate this list at the beginning of every week then use this list to review and prioritize those vulnerabilities that need to be manually mitigated.

EXAMPLE: REPORT ON ALL VULNERABILITIES TESTS RUN ON

This example describes how to generate a report that lists all vulnerability tests that have been run in the past week.

1. Create a **Vulnerabilities** report as described in Creating a Report on page 227.
2. Select the report you just created from the **Reports** pane. Its details are displayed in the **Configuration** pane.
3. In the **Data Scope** tab, add the **Create Date** field by clicking its green arrow . The **Age Category** field is moved up to the **Enabled Fields** pane.
4. Click the plus sign (**+**) next to the field. Its options are displayed.
5. In the **Age Category** field, select the Equals operation, then add the **1-7 Days** option to the **Selected** pane.
6. In the upper right-hand side of the window, click **Save**.
7. In the **General Details** tab, click Run **Report Now**. The report is run. Once complete a dialog box opens enabling you to open the generated report.

Working with Pre-Defined Vulnerability Reports

Pre-Defined Vulnerability Reports generate a list of vulnerabilities that have been detected in your network as the result of assessment tests. These types of reports are pre-defined, their tabular and layout options cannot be modified.

Pre-Defined Vulnerability Reports can optionally include an executive summary of vulnerabilities, general vulnerability data, and details regarding the vulnerabilities discovered, and can assist in generating regular reports on open vulnerabilities. For example, you may want to generate this list at the beginning of every week then use this list to review and prioritize those vulnerabilities that need to be manually mitigated.

EXAMPLE: REPORT ON ALL VULNERABILITIES CREATED IN THE PAST WEEK

This example describes how to generate a report that lists all vulnerabilities that have been opened in the past week.

1. Create a **List of Vulnerabilities** report as described in Creating a Report on page 227.
2. Select the report you just created from the **Reports** pane. Its details are displayed in the **Configuration** pane.
3. In the **Data Scope** tab, add the **Create Date** field by clicking its green arrow . The **Create Date** field is moved up to the **Enabled Fields** pane.
4. Click the plus sign (**+**) next to the field. Its options are displayed.
5. In the **Create Date** field, select the Between operation, then configure the From date from the week previous and the To date to today's date.
6. In the **Data Scope** tab, add the **Severity** field by clicking its green arrow . The **Severity** field is moved up to the **Enabled Fields** pane.
7. Click the plus sign (**+**) next to the field. Its options are displayed.
8. In the **Severity** field, type **High**. This displays vulnerabilities that SecureSphere have determined are at a high level of risk.
9. In the upper right-hand side of the window, click **Save**.
10. In the **General Details** tab, click Run **Report Now**. The report is run. Once complete a dialog box opens enabling you to open the generated report.

Working with Discovery Reports

Discovery reports list information that was obtained while using SecureSphere discovery to discover services and data in your network.

Discovery Reports Available in SecureSphere

Report Type	Description
Service Discovery Detailed	Reports on all services discovered as part of SecureSphere service discovery.
Service Discovery Execution Summary	Generates a list of SecureSphere service discovery runs that have been conducted in your network.

Service Discovery Reports

Server Discover reports lists all services discovered using the Server Discovery Module. A number of report definitions are available for Server Discovery which lists the various servers discovered in your network. For example, you may want to list all servers discovered last week, or you may want to list all Oracle based databases that were discovered. For a complete list of all available report definitions, see System Events Reports on page 645.



Note: Open ports not listed in the ADC Default Ports List Global Object or manually configured in the active module settings are reported as false in the manually defined column of Server Discover reports.

EXAMPLE: REPORT ON ALL SERVERS DISCOVERED IN THE PAST 7 DAYS

This example describes how to generate a report that lists all servers discovered within the past 7 days.

1. Create an **Discovery & Classification > Service Discovery & Classification > Service Discovery Detailed** report as described in Creating a Report on page 227.
2. Select the report you just created from the **Reports** pane. Its details are displayed in the **Configuration** pane.
3. In the **Data Scope** tab, add the **First Discovery Date** field by clicking its green arrow . The **First Discovery Date** field is moved up to the **Enabled Fields** pane.
4. Click the plus sign (**+**) next to the field. Its options are displayed.
5. Under **To**, select today's date. Under **From**, select 7 days ago.
6. In the **Data Scope** tab, add the **Service Status** field by clicking its green arrow . The **Service Status** field is moved up to the **Enabled Fields** pane.
7. Click the plus sign (**+**) next to the field. Its options are displayed.
8. Add the **New** option to the **Selected** pane.



Note: When configuring columns, you can also configure an aggregation function for the columns. For more information on aggregation functions, see Understanding Aggregation Functions.

9. From the upper right-hand side of the window, click **Save**.
10. From the **General Details** tab, click **Run Report Now**. The report is run. Once complete a dialog box opens enabling you to open the generated report.

Working with Profile Reports

Profile reports provide visibility into web and database profiles the have been dynamically created by SecureSphere, providing valuable information about various aspects of your network and assisting in seeing assets.



Notes:

- It is highly recommended that when creating profile reports, you include the columns Site, Server Group, Server and Application. This assists in breaking up the data in situations when similar items such as URLs exist in different applications, and retaining a logical grouping similar to how SecureSphere profiles the application.
- SecureSphere profile data includes multiple one-to-many relationships between the report columns, for example one URL has multiple parameters. In these situations, each parameter adds a line to the report. This can result in large reports.



Note: Profile Reports can integrate both a data scope to filter data in an alert, and tabular columns.

- For a list of fields available for Report Data scopes see Profile Criteria on page 595.
- For a list of columns available for reports, see Columns Available in Profile Reports on page 630.

Profile Reports Available in SecureSphere

Report Type	Description
Web Profile - URLs	Reports on URLs learned by SecureSphere that are used by applications in the network including URLs, their HTTP and SOAP methods, parameters and their attributes. For more information on web profile reports, see Web Profile Reports on page 224.
Web Profile - Cookies	Reports on cookies, their learning status and other information. For more information on cookies, see Cookie Protection in Web Profiles on page 252.
Web Profile - Learned Hosts	Reports on application URLs and the hosts on which they have been discovered.
Web Profile - Susceptible Directories	Reports on standard web application directories on web servers.
Web Profile - Web Application User Tracking	Reports on application login pages and helps identify URLs that may no longer be in use.

Web Profile Reports

Web Profile reports assist in understanding data that has been profiled related to web operation that include URLs, cookies, learned hosts, susceptible directories and web application users. This information can then be used to track and analyze web applications, assisting with maintenance and providing reference, for example, in maintaining consistency by facilitating using the same parameters in different locations.

EXAMPLE: WEB PROFILE REPORT - REPORTING ON NEW AND UPDATED URLS

This example describes how to generate a Web Profile - URLs report listing URLs and their attributes that SecureSphere has profiled. This can assist you in identifying URLs which have recently changed, which could help with scanner integration by reducing the amount of URLs that would need to be scanned.

1. Create a **Profile > Web Profile > Web Profile URL** report as described in Creating a Report on page 227.
2. Select the report you just created from the **Reports** pane. Its details are displayed in the **Configuration** pane.
3. In the **Data Scope** tab, add the **URL Update Last Few Days** field by clicking its green arrow . The **URL Update Last ew Days** field is moved up to the **Enabled Fields** pane.
4. Click the plus sign (**+**) next to the field. Its options are displayed.
5. In the **Last** field, type 7. This reports on URLs that have been created or updated in the last 7 days.
6. In the **Tabular** tab, add the following columns:
   - Site
   - Server Group
   - Service
   - Application
   - URL
   - Parameter Name
   - Parameter Min. Length
   - Parameter Max Length
   - Parameter Prefix

- Parameter Read Only
- Parameter Required

7. From the upper right-hand side of the window, click .
8. From the **General Details** tab, click **Run Report Now**. The report is run. Once complete a dialog box opens enabling you to open the generated report.

EXAMPLE: WEB PROFILE REPORT - REPORTING ON ALL BROKEN LINKS

This example describes how to generate a Web Profile - URL report listing URLs that no longer exist but are still linked to from within an application. This can assist you to eliminate broken links which otherwise might result users receiving errors.

1. Create a **Profile > Web Profile > Web Profile URL** report as described in Creating a Report on page 227.
2. Select the report you just created from the **Reports** pane. Its details are displayed in the **Configuration** pane.
3. In the **Data Scope** tab, add the **Link Status** field by clicking its green arrow . The **Link Status** field is moved up to the **Enabled Fields** pane.
4. Click the plus sign (**+**) next to the field. Its options are displayed.
5. Move the **Broken Link** option to the **Selected** pane.
6. In the **Tabular** tab, add the following columns:
   - Site
   - Server Group
   - Service
   - Application
   - URL
7. From the upper right-hand side of the window, click .
8. From the **General Details** tab, click **Run Report Now**. The report is run. Once complete a dialog box opens enabling you to open the generated report.

EXAMPLE: WEB PROFILE REPORT - URL SQL CORRELATION

This example describes how to generate a Web Profile - URL report listing URLs and their related queries on application databases. This can assist you in managing performance issues by determining if a specific URL is connected to a long running database query, and also assists in monitoring points which can potentially provide access to sensitive data.

This example is only available when licensed for both Web and Database security products.

1. Create a **Profile > Web Profile > Web Profile URL** report as described in Creating a Report on page 227.
2. Select the report you just created from the **Reports** pane. Its details are displayed in the **Configuration** pane.
3. Leave the **Data Scope** tab blank.



Note: You may want to choose a Data Scope field in order to limit the size of the report.

4. In the **Tabular** tab, add the following columns:
   - Site
   - Server Group
   - Service
   - Application
   - URL
   - All SQL related parameters
5. From the upper right-hand side of the window, click .

6. From the **General Details** tab, click **Run Report Now**. The report is run. Once complete a dialog box opens enabling you to open the generated report.

EXAMPLE: WEB PROFILE REPORT - WEB APPLICATION USER TRACKING

This example describes how to generate a Web Profile - Web Application User Tracking report. This can assist in seeing all URLs from which users log into an application and help identify URLs that may no longer be in use.

1. Create a **Profile > Web Profile > Web Application User Tracking** report as described in Creating a Report on page 227.
2. Select the report you just created from the **Reports** pane. Its details are displayed in the **Configuration** pane.
3. Leave the **Data Scope** tab blank.
4. In the **Tabular** tab, add the following columns:
   - Site
   - Server Group
   - Service
   - Application
   - Login URL
   - Action URL
5. From the upper right-hand side of the window, click .
6. From the **General Details** tab, click **Run Report Now**. The report is run. Once complete a dialog box opens enabling you to open the generated report.

Creating and Managing Reports

This section reviews the various tasks involved with creating and working with reports in SecureSphere. It reviews the following subjects:

Report Management Main Tasks 226
Creating a Report 227
Configuring Reports 228
Scheduling Reports 235
Configuring Reports With Followed Actions 236
Running a Report 237
Viewing Generated Reports 237

Report Management Main Tasks

A number of actions can be taken to create, customize and run reports. They include the actions in the following table.

Creating and Managing Reports - Main Tasks

	Action	Description	For more information, see...

	Action	Description	For more information, see...
1	**Creating a New Report**	Create a new report framework to be used to generate report results.	Creating a Report on page 227
2	**Configuring Reports**	Configure reports to reflect specific needs including time frame, data scope, scheduling, charts and more.	Configuring Reports on page 228
3	**Scheduling a Report**	Schedule a report to be run on a regular basis.	Scheduling Reports on page 235
4	**Configuring Reports for Distribution**	Configure a report with an action interface that automatically e-mails the report when it is generated.	Configuring Reports for Distribution by e-mail on page 236.
5	**Running Reports**	Run a report once it has been configured to generate data and view information.	Running a Report on page 237
6	**Viewing Generated Reports**	View reports that have been run to obtain an overview of status and other information.	Viewing Generated Reports on page 237
7	**Archiving and Purging Reports**	Reports can be archived by the SecureSphere Administrator.	SecureSphere Administration Guide

Creating a Report

You can create a new report from scratch or base it on an existing report definition and customize its contents.

To create a report:

1. In the **Main** workspace, select **Reports**. The **View Results** window appears displaying reports that have recently been generated.
2. Click **Manage Reports**. The **Report Management** window appears displaying available report definitions and configuration options.
3. In the **Reports** pane of the **Report Management** window, click **New**. The **New report** type dropdown menu appears.
4. Select a report type as described in Understanding Report Types on page 216. The **Create Report** dialog box appears for the report type you selected.
5. Type a name for the report. This name appears both in the generated report and in the **View Results** window.
6. Type a **Report Description** (optional). This description appears in the generated report.
7. Click the radio button to identify the basis of your new report. You have to options:
   - **From Scratch**: Creates a report from scratch with all available parameters needing to be configured. If creating an Active Module type report, select the active module you want to report on.
   - **Use existing**: Enables you to select an existing report definition to use as the basis for the new report. If you click this radio button, select the existing report to be used as the basis for the new report from the **Use existing** drop-down list. Only available for alerts, system events, and active modules.
8. Enable the **Copy Permissions from Existing Report** if desired. This will apply the same permissions to the new report definition as those from the existing report that was selected.
9. Click **Create**. SecureSphere creates the requested report.

10. Configure report properties as required. For a detailed description of available options, see Configuring Reports on page 228.



Note: The minimum items that need to be configured to use a report include Name, Report Format, Data Scope and Tabular or Data Analysis View. For more information on configuring reports, see Configuring Reports on page 228.

Configuring Reports

SecureSphere enables you to configure and customize reports using a number of features. For pre-defined reports, many of these properties are already pre-configured, though it is also possible to further customize pre-defined reports to meet your needs.

To configure a report:

1. In the **Main** workspace, select **Reports**. The **View Results** window appears displaying reports that have recently been generated.
2. Click **Manage Reports**. The **Report Management** window appears displaying available reports and configuration options.
3. Select the report you want to configure. That report's configuration details are displayed in the configuration pane.
4. Configure the report shown in the table below.
5. Once the report has been configured, it can be run. For more information on running reports, see Running a Report on page 237.

Tab	Description
General Details	Configure the report name and description, attach a followed action, determine the report format and tag the report with keywords. For more information, see Configuring General Report Details on page 229.
Data Scope	Configure the subset of data to be included in the report. For more information, see Configuring Data Scope on page 230.
Tabular	Determine whether to display information in table format, select columns to include in the table and set the sorting order of information. For more information, see Configuring Tabular Options on page 231.
Data Analysis View	Configure one or more charts to display the resulting data in graphic format. For more information, see Configuring Data Analysis View on page 232.
Scheduling	Configure a schedule for running the report on the regular basis. For more information, see Scheduling Reports on page 235.
Results	Enables you to easily find reports based on the definition selected in the Reports window. Lists the instances this specific report was run and enables you to view the various reports. For more information, see Viewing Results on page 234.
Permission	Determine which users or groups have permissions to view the report. For more information, see Configuring Permissions on page 235.

Managing Favorite Reports

You can mark a report as a favorite to make it easy to find. Favorites are user specific. Therefore, a report marked as a favorite by one user does not appear as a favorite to another logged on user. Reports created by a specific user are automatically added to that user's favorites. Selecting the favorites option is the easiest way to display all of the reports you have created.

To mark a report as a favorite:

1. In the **Main** workspace, select **Reports**. The **View Results** window appears displaying reports that have recently been generated.
2. Click **Manage Reports**. The **Report Management** window appears displaying available reports and configuration options.
3. In the **Reports** pane, right-click the report you want to mark as a favorite, then choose **Add to Favorites**. The report is added as a favorite and a check mark appears in the Favorite column. When the **Show Favorites** option is enabled, this report appears together with all other reports marked as a favorite.



Notes:

- You can select multiple reports and mark them all as favorites at once by using shift + click and CTRL + Click.
- To remove a report from favorites, right-click the report in the Report pane and choose Remove from Favorites.

Configuring General Report Details

The General Details tab enables you to configure various general aspects of a report.

To configure general details:

1. In the **Main** workspace, select **Reports**. The **View Results** window appears displaying reports that have recently been generated.
2. Click **Manage Reports**. The **Report Management** window appears displaying available reports and configuration options.
3. Select the report you want to configure. That report's configuration details are displayed in the configuration pane.
4. Click the **General Details** tab. General details options are displayed. Information that can be configured in the general details tab includes the following:

Option	Description
General Information	**Name**: Name used to identify the report. This name is displayed as the report Name in the View Results window is open, and is also appears in printed reports. When modifying a report name, names that appear in results are automatically changed. **Description**: Enables you to add a free text description to a report. This is useful for providing additional information that you may want to include with the report such as special conditions for its creation. The description subsequently appears on the report cover page. **Data Type (not editable)**: Displays the type of data the report is based on. **Followed Action**: Enables you to attach a followed action that is conducted whenever a report is generated. For example, you can attach a followed action that includes an e-mail action interface that automatically e-mails a copy of the report to a number recipients every time the report is generated. For further information on configuring s, see Configuring Reports With Followed Actions on page 236. **Save Results on Run Now**: Automatically saves a copy of the generated report when the report is generated using the Run Report Now button located at the bottom of the General Details tab or using the Run Report Action Button. When this option is not selected, report results are not saved and cannot be viewed later even by the report's owner.
Sections	**Executive Summary**: Includes an executive summary at the beginning of the report. **Note:** The following two options are only available with Vulnerability Reports. **Vulnerability Data**: Includes vulnerability data with the report. **Details**: Includes vulnerability details with the report.
Report Format	**Format**: Determines the format of the output file. Options include Comma Separated Values (CSV), or Portable Document Format (PDF). Not all options are available for all reports. **Page Orientation**: Determines orientation of the page. Options include Landscape and Portrait. **Cover Page**: Determines whether SecureSphere includes a cover page with the report. Report cover page displays the report name, description as configured in general information, data scope, creator of the report and the date created. For information on how to modify the report cover page, see the Imperva Knowledge Base.
Keywords	Tags the report with specific keywords that can later be used to search for reports using the filter. There are two types of keywords, ADC keywords which cannot be modified and user-defined keywords that can be created by the SecureSphere administrator.

Configuring Data Scope

The Data Scope tab enables you to limit the data included in the report to that of the data scope field selected. Configuring Data Scope is not required, however it helps to focus on a specific subset of data.

Notes:

- Data scope options are context sensitive.
- Data scope definitions are not included with reports unless a cover page is used.

To configure the data scope:

1. In the **Main** workspace, select **Reports**. The **View Results** window appears displaying reports that have recently been generated.
2. Click **Manage Reports**. The **Report Management** window appears displaying available reports and configuration options.
3. Select the report you want to configure. That report's configuration details are displayed in the configuration pane.
4. Click the **Data Scope** tab. Data scope options are displayed. Information that can be configured in the data scope tab includes the options in the table below.

   Note: When configuring a Time Frame for Alert reports, you must also configure the field Last Few Days. This field must be configured with a number greater than the amount of days from the beginning of the configured time frame.

5. Click the green arrow of the **Available Field** you want to add as part of the search. The field is moved up to the **Enabled Fields** pane.
6. Click the plus sign (**+**) next to the desired field.
7. Configure the fields as required. Information selected here is used to filter retrieved data.
8. Repeat steps 3-5 to add additional filtering criteria as required.

Data Scope Tab Options

Name	Description
Filter	Selects the data to be included in the report. Fields that are available depend upon report type and the data source used for the report. For a comprehensive list of data fields, see Filter Criteria and Report Data Fields Overview on page 575.

Configuring Tabular Options

The Tabular tab enables you to add a table to a report and configure its content. Available tabular options depend on the type of report you are configuring.

To configure tabular options:

1. In the **Main** workspace, select **Reports**. The **View Results** window appears displaying reports that have recently been generated.
2. Click **Manage Reports**. The **Report Management** window appears displaying available reports and configuration options.
3. Select the report you want to configure. That report's configuration details are displayed in the configuration pane.
4. Click the **Tabular** tab. Tabular options are displayed.
5. Select **Tabular View** at the top of the **Tabular** window. Tabular options are enabled. A table will be included with reports that are run. Information that can be configured in the Tabular tab includes the following:

Option	Description
Title	Determines the title of the table as it appears in the report.
Available Columns	Enables you to determine which columns to include in the table. You can add and remove columns using the add and remove arrows and determine the order of the columns using the up and down arrows. Some columns cannot be used together in reports due to the types of data they contain. If the columns you have selected are not compatible, you will receive an error message. For a comprehensive list of columns that can be used in reports, see Report Columns on page 611.
Sorting	Enables you to determine the order in which data is sorted in the table. Sorting can be configured based on a first and second parameter, and can be either ascending or descending.
Generate Group Headers	Groups information in the table by values selected in the sorting column. For example, suppose you are generating an alert report. If you select generate group headers and sort by session ID, a row heading will precede each group of alerts with the same session ID, for each session ID in the data being reported on.

Configuring Data Analysis View

The Data Analysis Views tab enables you to add charts to a report to provide graphic representation of data and configure their content. Up to five charts can be added to a report. Each chart is configured as a separate entity, with its own title, customized chart type and other parameters.

To configure data analysis options:

1. In the **Main** workspace, select **Reports**. The **View Results** window appears displaying reports that have recently been generated.
2. Click **Manage Reports**. The **Report Management** window appears displaying available reports and configuration options.
3. Select the report you want to configure. That report's configuration details are displayed in the configuration pane.
4. Click the **Data Analysis** tab. Data analysis options are displayed.
5. Select the **Data Analysis View** checkbox at the top of the window for each chart you'd like to create. Tabular options are enabled. Information that can be configured in the Data Analysis View tab (per chart) include the following:

Option	Description
Title	Determines the title of the chart as it appears in the report.
Chart Type	Determines the type of chart used to display data. Options include Pie Chart, Bar Chart, Stacked Bar Chart and Cylinder.
X-Axis	Determines the data field used to plot the X-Axis. Options include any data field that is available in the selected report type.
Y-Axis	Depending on report type may represent the following: Determines the data field used to plot the Y-Axis based on fields that are available in the selected report type. Determines the location of the Y-Axis in the chart. Non quantitative columns selected to be used as the Y axis display the number of distinct values for this field. For example: If you choose URL for your X-axis and Parameter Name for the Y-axis (a non quantitative column), the graph displays the various parameters by URL.
Grouping	Groups results in the chart according to the selected attribute. For example, you can group the application user names by application name. Only available with bar and stacked bar type charts. Only available in stacked bar charts.
Maximal number of values displayed in X-Axis	Determines the maximum number of values used as the markers for the X-Axis that will be displayed in a chart. **Show total of others**: Shows the number of all items that didn't fall into the "top x" items in a column named **Others**. For example, if your x-axis is users and you only wanted to show the top 5 users, and there were 20 users all together, the Others column would display 1.
Maximal number of values displayed in grouping	Determines the maximum number of values displayed in a grouping as based on grouping configuration of the chart. 3 **Show total of others**: Shows the number of all items that didn't fall into the "top x" items in a column named **Others**. For example, if your y-axis is users and you only wanted to show the top 5 users, and there were 20 users all together, the Others column would display 20.
Present Chart	Includes the Chart in the report. Enabled by default.
Present Tabular View	Includes a table in the report that displays all data for the configured data scope. Enabled by default.

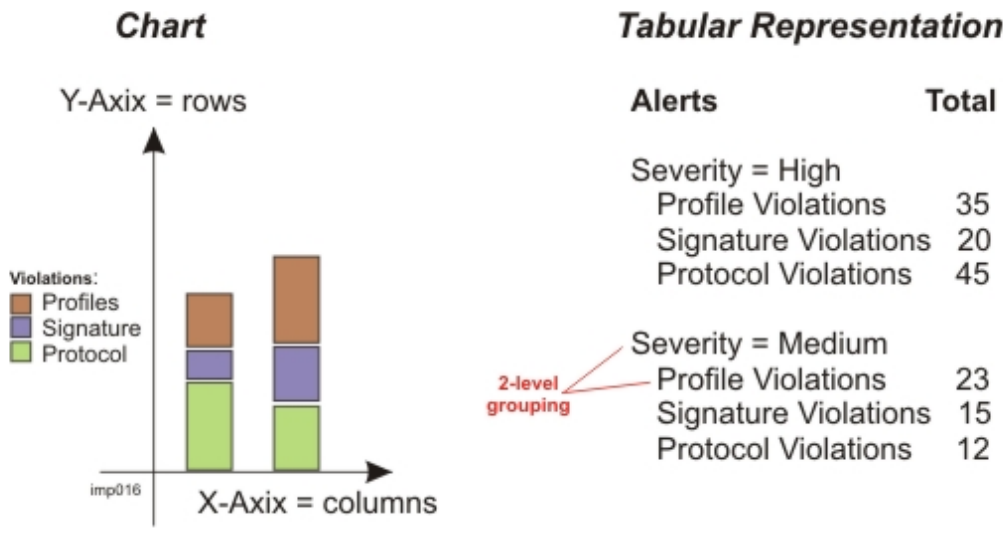


Figure 27: Tabular and Chart Examples

Understanding Data Analysis View Data Sources

Charts generated as part of the data analysis view are based on specific sources of data. This impacts what information is displayed in the Y-Axis. The following table lists data sources and the information used to build the chart.

Data Sources

Data Source	X-Axis	Y-Axis	Comment
Alerts	Corresponds to value selected when configuring the chart.	Groups by number of alerts and related events.	Y-Axis corresponds to the rows in the detailed view
Assessments	Corresponds to value selected when configuring the chart.	Groups by number of test run instances.	Instance = the unique combination of a: job, test name, time (occurrence), and IP address The data scope of an assessment is defined by the selection of instances
System Events	Corresponds to value selected when configuring the chart.	Groups by number of system events.	

Viewing Results

The Results tab lists all reports that have been generated based on the selected report definition. This assists in finding report results related to that report, and provides a comprehensive history of report generation in a single location.

To view results of reports previously run:

1. In the **Main** workspace, select **Reports**. The **View Results** window appears displaying reports that have recently been generated.
2. Click **Manage Reports**. The **Report Management** window appears displaying available reports and configuration options.

3. Click the report whose results you want to display. Report configuration options are displayed in the configuration pane.
4. Click the **Results** tab. Previous results are displayed.
5. Click the results you want to view. The selected results are displayed.

Configuring Permissions

The Permissions tab enables you to determine which users and user groups have permission to view a report. By default report creator and admin group users have permissions for any report.



Note: Pre-defined reports are reports that come with SecureSphere and are defined by the SecureSphere ADC department. You can base a new report on a predefined report or modify it on the fly for the current report, but you cannot save changes. Pre-defined reports can be seen by selecting ADC in the Defined by Basic filter category.

To configure permissions:

1. In the **Main** workspace, select **Reports**. The **View Results** window appears displaying reports that have recently been generated.
2. Click **Manage Reports**. The **Report Management** window appears displaying available reports and configuration options.
3. Click the report whose permissions you want to modify. Report configuration options are displayed in the configuration pane.
4. Click the **Permissions** tab. You may need to scroll through the tabs using the scroll buttons. Permission options are displayed.
5. Use the arrows to add or remove users or user groups from the Users with View Permissions pane and subsequently assign permissions.

Scheduling Reports

SecureSphere enables you to schedule reports for automatic generation. This can be useful to create a regular workflow in which reports need to be generated on a regular basis or where one or more parties may need to review regular reports.



Note: When a SecureSphere user who created a report is deleted from SecureSphere, any scheduled reports they had configured no longer run.

To schedule reports for automatic generation:

1. In the **Main** workspace, click **Reports**. The **Reports** window appears.
2. In the **Reports** window, click **Manage Reports**. The **Manage Reports** window appears.
3. In the **Report Template** pane, select the report you want to configure for scheduling. That report's configuration options are listed in the **Configuration** pane.
4. In the **Configuration** pane, click **Scheduling**. Scheduling options appear.
5. Select the occurrence as follows:
   - **None**: Configures the report so it is not generated automatically.
   - **Once**: Schedules the report to be run on a single occasion. If you select this option, configure a data and time for running.
   - **Recurring**: Schedules the report to be run on a regular basis. If you select this option, configure scheduling options as shown in the table **Types of Occurrences** below.

The report is scheduled as configured. Alternatively you can also run the report now. For further information, see Running a Report on page 237.

Types of Occurrences

Parameter	Description
Type of Occurrences	Daily, Weekly, Monthly or Custom. Custom enables you to schedule a report to run at intervals of minutes.
Interval	Context sensitive. The interval changes based on the selected Type of occurrence, as follows: **Daily**: Enter every number of days the report should be run. **Weekly**: Enter every number of weeks the report should be run and the day of the week the report should be run on. **Monthly**: Select day or day of the week. **Custom**: Enter every number of minutes the report should be run.
Starting From	Select the first date on which you want the reoccurring report to be run.
At Time	Select the first time at which you want the reoccurring report to be run.

Configuring Reports With Followed Actions

Followed actions enable you to cause an event to happen when a report is generated. There are two primary actions that can be configured and applied to reports. These include:

- **Distribution by e-mail**: For more information see Configuring Reports for Distribution by e-mail on page 236.
- **Creation of review task**: For more information see Configuring Reports with Tasks on page 236.

Configuring Reports for Distribution by e-mail

SecureSphere enables you to configure reports to be distributed once they have been generated. This is accomplished by configuring an action set with the required action interfaces, then attaching the action set as a followed action to the report configuration.

To configure reports for distribution:

1. Create an action set with an e-mail action interface as described in Creating Custom Action Sets.

   For information on specifically configuring the Send an Email action interface, see the Send an Email action interface under Configuring Action Interface Parameters on page 332.
2. In the **Main** workspace, click **Reports**. The **Reports** window appears.
3. In the **Reports** window, click **Manage Reports**. The **Manage Reports** window appears.
4. In the **List** pane, select the report you want to configure for distribution. That report's configuration options are listed in the **Details** pane of the **General Details** tab.
5. From the followed action dropdown list, select the followed action you configured in step 1. The report has been configured for distribution.

Configuring Reports with Tasks

SecureSphere enables you to configure reports to be accompanied with tasks when they are generated. This can be useful when wanting to assign someone to review a report, and is accomplished by configuring an action set with the required action interfaces, then attaching the action set as a followed action to the report configuration.

To configure reports with a task:

1. Create an action set with a **Review** or **Assignment** task as described in Creating Custom Action Sets on page 330.

2. In the **Main** workspace, click **Reports**. The **Reports** window appears.
3. In the **Reports** window, click **Manage Reports**. The **Manage Reports** window appears.
4. In the **List** pane, select the report you want to configure with a task. That report's configuration options are listed in the **Details** pane of the **General Details** tab.
5. From the **Followed Action** dropdown list, select the followed action you configured in step 1. The report has been configured with a task. Every time the report is generated, a task will be created.

Running a Report

When you create a report you create the framework for a report. Once the report template is created, you can either schedule it to run on a regular basis or run it immediately. This procedure describes how to run a report immediately. For information on how to schedule a report to be run, see Scheduling Reports on page 235.



Notes:

- Generating audit reports based on event data, as opposed to index data, may result in a long generation time.
- Generating audit reports require that you select a specific policy as part of the Datascope.

To run a report:

1. In the **Main** workspace, select **Reports**. The **View Results** window appears displaying reports that have recently been generated.
2. Click **Manage Reports**. The **Report Management** window appears displaying available reports and configuration options.
3. Select the report you want to run. That report's configuration details are displayed in the configuration pane.
4. From the bottom of the **General Details** tab, click **Run Report Now**, or from the **Action** button at the top of the window, click **Run Report**. The report is run. When complete, the Download Now dialog box is displayed enabling you to open and view the report.



Note: If you have made changes to a Report Definition, you must save the report before clicking Run Report Now in order for the changes to be implemented in the Report Result.

Viewing Generated Reports

Reports that have been generated are listed in the **View Results** window. You can select the reports in this window and display them at any time.

To view a generated report:

1. In the **Main** workspace, click **Reports**. The **Reports** window appears.
2. In the **Reports** window, click **View Results**. Reports that have been generated are displayed in the window.
3. Click a report. The report is opened in its relevant format.

   The Reports Results window additionally provide you with the following options:

   - **Show Favorites**: Shows reports that have been marked as a favorite. For further information, see Managing Favorite Reports on page 229.
   - **Show Enabled**: Shows reports that have been configured as enabled on the General Details tab. Only available if logged on as Administrator. For further information, see Configuring General Report Details on page 229.

Filtering Reports

You can use SecureSphere filters to search for and display reports based on a number of criteria. To access the filters, click the **Filter** tab. Two types of filter are available with Reports:

- **Quick Filter**: Enables you to filter reports according to report name.
- **Basic Filter**: Enables you to select various criteria on which to filter reports. For more information see Basic Filter Criteria on page 238

Basic Filter Criteria

The Basic Filter enables you to select the desired search criteria and apply it to display data containing specific attributes. For information on how to use the Basic Filter, see Using the Basic Filter on page 110. The following table lists the available basic filter criteria.

Basic Audit Filter Criteria

Criteria	Description
ADC Keywords	Filters reports by ADC keywords that are built into SecureSphere. These keywords cannot be modified.
User Defined Keywords	Filters reports by user defined keywords that were added to the report in the Details Tab. User defined keywords can be created by the SecureSphere administrator.
Data Source	Filters reports based on the data source the report is based on. Alternatively known as the report type. For a list of report SecureSphere report types, see Creating a Report on page 227.
Active Module Type	Filters reports based on active module types. Results only include active module reports.
Defined By	Filters reports based on type of report. Options include ADC or User Defined.
Run Date	Filters reports based on the date the report was run. Options include Today, Last Week, Other, Never Run.
Scheduling	Filters reports based on scheduling configuration. Options include None, One Time, Daily, Weekly and Monthly.

CHAPTER 9

Web Application Profiles

SecureSphere's Dynamic Profiling technology automatically examines Web application to create a comprehensive profile of its structure and behavior. Dynamic profiling overcomes the biggest drawback of other application - manual creation and maintenance of security policies. Valid application changes are automatically recognized and incorporated into the profile over time, ensuring that SecureSphere detects potentially malicious exceptional activity. Administrators can manually tune and control the automatically generated profile at any time.

SecureSphere begins creating an application's profile the first time it sees traffic for the application. Over time, SecureSphere sees more traffic and refines the profile accordingly. Eventually SecureSphere builds an accurate model of the application and begins to enforce the profile. All of this happens automatically, though at every stage the administrator can manually tune the profile to improve its accuracy.

This section describes Web application profiles, including:

Understanding Web Profiles in SecureSphere 240
Web Profile Tasks 242
Tuning Web Profiles 257
Web Profile Graphs 266
Web Profile Optimization 267
Web Profile Policy 275
Web APU (Automatic Profile Update) 279
Web Profile System Definitions 283

Understanding Web Profiles in SecureSphere

Web profiles are comprised of a number of different aspects. SecureSphere has a variety of features that assists in managing Web profiles including URLs and Cookies. The following topics provide information to assist you in understanding how SecureSphere database profiles work:

Overview of SecureSphere Web Profiles 240
Web Profile Components 240
Web Profile Structure 241
Profile Modes 241

Overview of SecureSphere Web Profiles

A Web profile is a model of a Web application, specifying how the application's URLs and cookies are normally accessed. The profile is built up over time automatically by SecureSphere as it learns the application, and then refined by the administrator. Once the learning period is over, SecureSphere starts protecting the application by identifying profile deviations, that is, HTTP requests which do not conform to the profile, and taking the action specified in the policy associated with the profile. Profiles represent the "whitelist" security model, what is allowed, in contrast to ADC signatures, for example, which represent the "blacklist" model, what is not allowed. For information about signatures, see Working with Signatures on page 307.

A Web application profile is built around URLs and cookies, and SecureSphere begins creating the application's profile the first time it sees traffic for the application.

Each application has an associated profile policy, which defines what SecureSphere does when a profile deviation is detected. In addition to the profile policy, an application can have any number of other policies associated with it.

The following topics provide information to assist you in understanding how SecureSphere web profiles work:

Web Profile Components

A Web application profile consists of the following components:

- **URLs List**: see Learned URLs on page 246
- **URL Patterns**: see URL Patterns on page 252
- **Cookies List**: see Cookie Protection in Web Profiles on page 252
- **Learned Hosts**: see Displaying Information About a Profile's URLs on page 256
- **Web application user tracking**: see Web Application User Tracking on page 287

Web Profile Structure

A Web application profile is structured as follows:

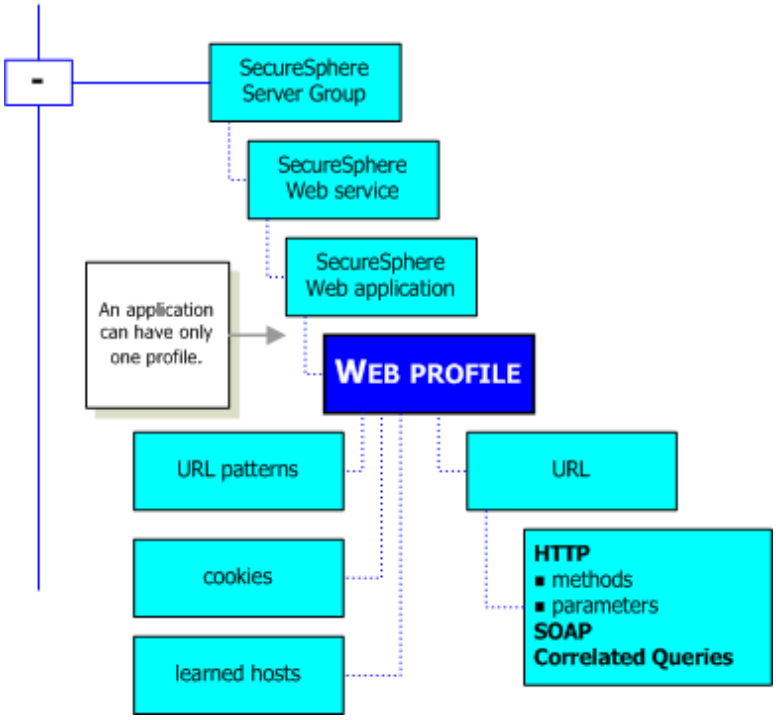


Figure 28: Web Profile Structure

Profile Modes

Each URL can be in one of the following modes:

- **Learning mode**: SecureSphere is not enforcing the profile policy for this URL, because it is still learning how the URL is accessed. For more information, see Learning Mode on page 242.
- **Protect mode**: SecureSphere is enforcing the profile policy. For more information see Protect Mode on page 242.

An additional mode (**Ignore mode**) is used for protecting cookies. For more information, see Cookie Protection in Web Profiles on page 252.

During the time the profile is being created, the application is protected from generic and known attacks by ADC content and user-defined policies, and SecureSphere does not learn behavior which violates existing policies. For example, SecureSphere does not learn a client request which includes abnormal HTTP behavior or which SecureSphere identifies as containing a worm.



Note: Since SecureSphere does not learn behavior which violates existing policies, you can further restrict the behavior SecureSphere learns by defining an appropriate policy. For example, if you want SecureSphere to learn only the GET and POST methods, define a policy which flags other methods as violations.

Learning Mode

At first, each URL is in Learning mode, during which time SecureSphere builds a list of its parameters based on client requests and application/server responses. While in Learning mode, SecureSphere does not enforce the profile policy for the URL.

In addition to URLs which SecureSphere process learns automatically, the administrator can manually add URLs to the profile. For more information, see Manually Adding a URL on page 263.

To avoid learning redundant URLs, SecureSphere can be configured to group URLs based on patterns. For more information, see URL Patterns on page 252.

The default value for the maximum length of time a URL can stay in Learning mode before being automatically switched to Protect mode is specified in the **Protect mode Thresholds (Web)** window (**Admin** > **System Definitions** > **Switching to Protect Mode Thresholds (Web)**).

However, a URL may be switched sooner, based on the following factors:

- The number of occurrences in the traffic
- Whether the URL has parameters
- The time passed since it was first seen by SecureSphere

The more frequently the URL occurs in traffic, the sooner it is switched to Protect mode.

The administrator can switch a URL between Learning and Protect mode at any time.

Protect Mode

For URLs in Protect mode SecureSphere inspects client requests and server responses, detecting violations and enforcing the profile policy.

When in Protect mode, URLs are being learned while SecureSphere is also applying security policies to monitor and protect protected URLs.

The administrator can tune the profile by modifying the parameters of a URL in Protect mode. For example, administrators can remove false positives or define additional checks on a URL's parameters. All of a URL's parameters, except for correlation data, can be manually modified.

Web Profile Tasks

SecureSphere begins building an application's profile the first time it sees traffic for the application. The process is automatic, but the administrator can improve the profile's effectiveness by monitoring the profile as it takes shape to remove redundant URLs and false positives.

The SecureSphere administrator must perform the tasks described in the following table.

Web Profile Task Overview

	Task Overview	Description	For more information, see...

	Task Overview	Description	For more information, see...
1	**Define the Web applications.**	Assign applications to services. SecureSphere builds each application's profile from the application's traffic. If you do not explicitly define an application, SecureSphere creates a Default Web Application.	Managing Web Applications on page 69
2	**Monitor the Web profile as it is being built during the learning period.**	Review the list of URLs and the logs to identify patterns and determine if plugins are required. If you define a plugin, the URLs affected by the plugin must be relearned.	Custom Web Plugins on page 487
3	**Customize the web profile policy.**	Review the default Web Profile policy and modify as required.	Web Profile Policy on page 275
4	**As URLs and cookies switch to Protect mode, periodically review the profile and the log.**	Identify violations, eliminate false positives, and tune the profile and profile policy.	Tuning Web Profiles on page 257
5	**Optionally, modify the profiling parameters.**	There is rarely a need to change the defaults. You should do this only in special cases, for example, where limits are exceeded. In any case, it is recommended that you contact support before doing so because there may be other approaches to solving the problem.	**Admin** > **System Definitions** > **Profile Size Limits**

Viewing Profiles

You can view a summary of all the profiles and statistics about them by selecting Profiles > Overview from the **Main** workspace.

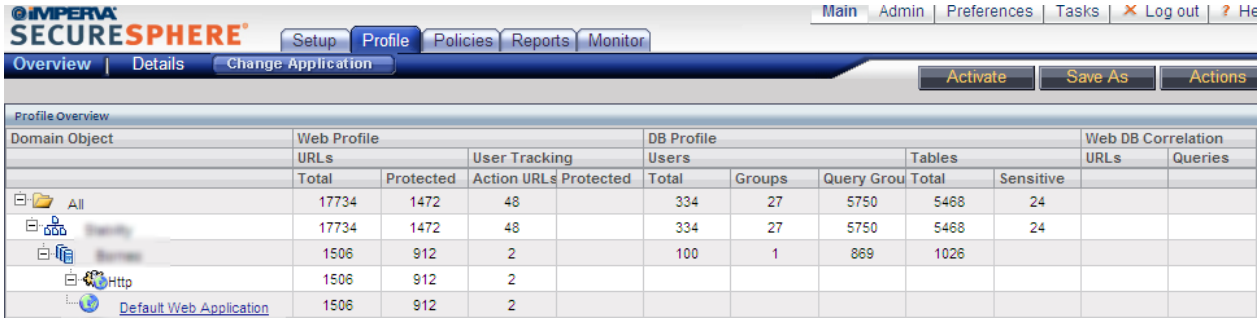


Figure 29: Profiles Overview

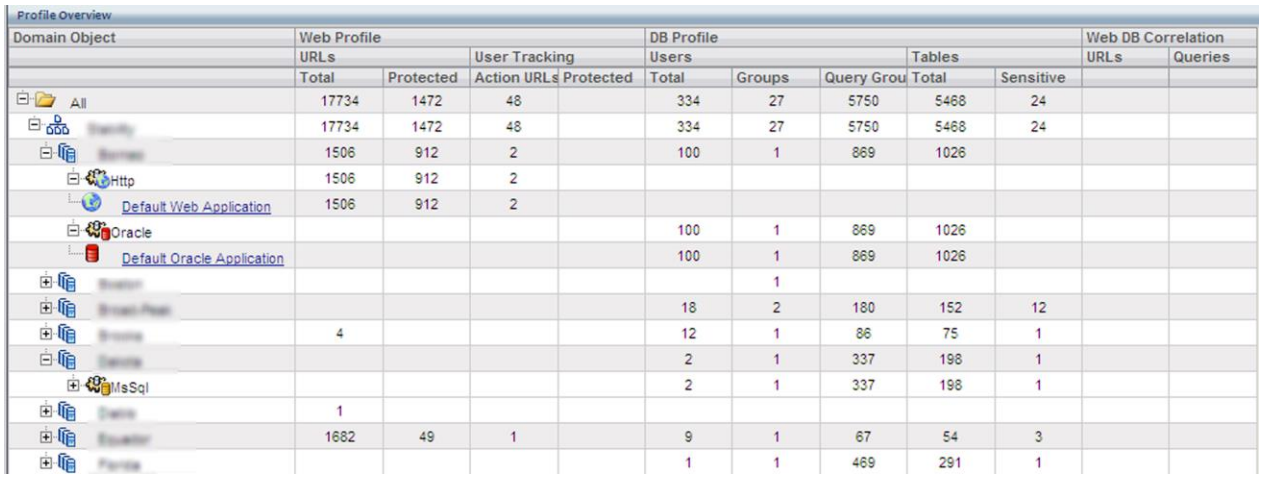


Figure 30: Profiles Overview

Under Domain Object, a list of server groups (and within them, applications), is displayed in the form of a tree. For each server group, statistics are displayed for each of the profile types associated with its applications:

- **Web profile**: see the table below

- **Web-DB Correlation**: see Correlated Queries Tab on page 250



Note: A warning icon displayed next to a web application in the profile overview tree indicates that the application's profile has optimization issues (see Web Profile Optimization on page 267). The warning icon "floats" up the hierarchy (service, server group, site *etc.*), but only the icon next to the application is clickable and takes you to the the **Optimization Issues** page.

Web Profile Data

Columns	Description
URLs - Total	The number of learned URLs.
URLs - Protected	Of these, the number of URLs in Protect mode.
User Tracking - Action URLs	See Web Application User Tracking on page 287 .
User Tracking - Protected	Web Application User Tracking on page 287.

Web Application Profiles

To display a Web application's profile, do one of the following:

- Click on the application in the **Profiles Overview** window shown in Viewing Profiles on page 244
- If a Web profile is already displayed, click **Change Application** in the profile menu
- In the **Main** workspace, select **Profiles** > **Application View**, in which case the last application is re-displayed.

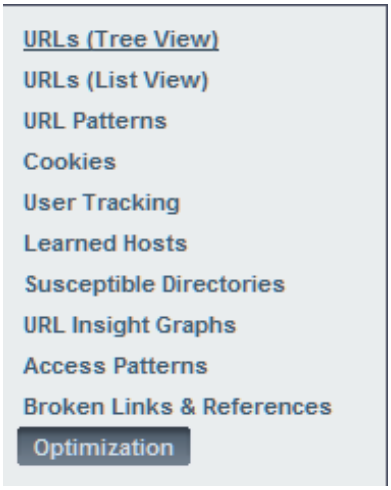


Figure 31: Web Profile Menu

Note: Application View provides the functionality that was partially implemented in the **Display** button in previous SecureSphere versions.

Web Profile Menu

Web Profile Menu

Menu Item	For more information, see...
URLs (Tree View)	Viewing URLs on page 246
URLs (List View)	Viewing URLs on page 246
URL Patterns	URL Patterns on page 252
Cookies	Cookie Protection in Web Profiles on page 252
User Tracking	
Learned Hosts	
Susceptible Directories	Configuring Susceptible Directories on page 426
URL Insight Graphs	Overall Graphs on page 266
Access Patterns	Access Patterns on page 267
Broken Links & References	Broken Links and References on page 251
Optimization	Web Profile Optimization on page 267

Learned URLs

SecureSphere learns the application URLs based on the traffic to the application, collecting the following information:

- URLs, their HTTP methods, parameters and their attributes
- Cookies
- Host names
- Login URLs
- Correlation information

A URL's status is indicated by the icon to the left of its name, as listed in the following table.

URL Status Icons

Icon	Status	Description
	Learning mode	Learning Mode on page 242
	Protect mode	Protect Mode on page 242
	broken link or reference	Broken Links and References on page 251

You can manually switch a URL from Learning mode to Protect mode at any time. See Switching Between the URL Modes on page 261 for more information.

Viewing URLs

You can view all the URLs learned by SecureSphere.

To view a URL:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.

If you do not see the application, either expand the server group or click **Change Application**.

The URL list is displayed in two forms:

- **Tree** - click **URLs (Tree View)** in the profile menu, as seen in the figure under Web Application Profiles on page 245

  In the **URLs** pane, the numbers in the brackets next to the directory name indicate the number of protected URLs and the total number of URLs.
- **List** - click **URLs (List View)** in the profile menu, as seen in the figure under Web Application Profiles on page 245

  In the **URLs** tab, statistics about the URLs are listed: methods, occurrences, number of parameters, response time and SQL queries.

To sort the list by on any of these columns, click the column heading. The sort sequence is indicated by an arrow.

To change the sequence, click the column heading.



Note: You can sort the URLs in list view by clicking the column header arrow.

3. Select a URL. The URL's details are displayed in three tabs on the right of the window:
   - **HTTP**: see HTTP Methods and Parameters on page 247
   - **SOAP**: see SOAP Actions on page 248
   - **Correlated Queries**: see Correlated Queries Tab on page 250

HTTP Methods and Parameters

The HTTP learned methods are listed in the **HTTP** tab, along with the learned parameters and their constraints. During the early stages of the learning phase, the constraints are generous. For example, at first a parameter's minimum and maximum values define a large range of permitted values, but as time passes, SecureSphere reduces the range to reflect the values actually seen in the observed traffic.



Note: You can reduce the size of a Web profile by not learning URLs with no parameters. For more information, see the Imperva Knowledge Base.

During the process of constant learning, parameters are identified by SecureSphere as JSON, if they are received as POST parameters and if the request specifies the required content type. The content types that we accept and parse as JSON are `application/json`, `application/json-rpc` and `application/jsonrequest`. First, the text must be structured according to RFC 4627, which defines required grammar, legal values, objects, arrays, numbers and more. If the message does not conform to RFC 4627 (*e.g.*, an array that is not closed, etc), SecureSphere generates an alert about malformed JSON.

If the text is structured as required, SecureSphere then parses parameter names and values to verify that their content is not malicious. If SecureSphere identifies malicious content, an alert is generated. If there is no malicious content, SecureSphere translates JSON parameters into regular parameter names by adding a dot between name components.

HTTP Tab Parameters

information	Description	Examples
HTTP Methods	The list of methods learned for this URL.	GET, POST, SOAP, XML, JSON and so on. You can restrict the methods learned by defining a custom policy that allows only certain methods. **Note:** JSON, XML and SOAP are sent as POST actions and SecureSphere treats each of them separately and not as a regular HTTP action. If a POST request specified a JSON, XML or SOAP content type, SecureSphere learns the request as these instead of a generic POST. **Note:** Because the web server does not respond to the RPC_IN_DATA method, SecureSphere cannot know whether the call was successful, so SecureSphere automatically learns the RPC_IN_DATA method only if the URI is `/rpc/rpcproxy.dll`. Otherwise, the SecureSphere administrator should add the RPC_IN_DATA method to the profile manually (see Editing a URL's Methods on page 260).
URL parameters and their attributes	The list of parameters learned for this URL, and their attributes.	Value type, minimum and maximum lengths, whether the parameter is required or optional, read-only, or a parameter prefix

SOAP Actions



Note: SecureSphere supports SOAP 1.1 and 1.2.

SecureSphere automatically identifies URLs containing XML and SOAP content and learns their SOAP actions. If the URL does not include SOAP actions but contains XML, SecureSphere shows the XML as the default SOAP action.

SecureSphere parses XML into structures, with each structure representing a value. The XML structures are learned and added to the **Parameters** section of the SOAP action. For each of these structures SecureSphere learns its minimum and maximum size, its value type and whether or not it is required (in the same way as SecureSphere learns regular URL parameters). When the URL is switched to Protect mode, SecureSphere invokes the appropriate violation whenever the input does not conform to the profile, see table **Web Policy Rules** found in Viewing a Web Profile Policy on page 276.



Note: If SOAP traffic has been identified on the URL, the SOAP method is shown in the tree. Although SOAP is usually done over POST, the URL will not be profiled as accepting POST if only SOAP traffic has been seen and not regular, non-SOAP POST HTTP requests.

XML Parsing

Structures are created using the full hierarchy of nested tags containing each value. For example, an XML file based on the following schema:

```
<code><schema> </code>
            <code><complexType name="purchaseOrder"> </code>
                  <code><element name="comment" type="string"/> </code>
                  <code><element name="item" minOccurs="0"> </code>
                        <code><complexType> </code>
                              <code><element name="productName" type="string"/>
</code>
```

```
                                    <code><element name="quantity" /> </code>
                                    <code><element name="Price" type="decimal"/>
</code>
                                          <code><attribute name="partNum"
type="SKU"/> </code>
                           <code></complexType> </code>
                      <code></element> </code>
               <code></complexType> </code>
<code></schema> </code>
```

is represented as:

```
<code>XML/purchaseOrder/comment </code>
<code>XML/purchaseOrder/item/productName</code>
<code>XML/purchaseOrder/item/quantity</code>
<code>XML/purchaseOrder/item/Price </code>
```

Nested XML Parsing

SecureSphere can optionally parse nested XML, so that if an element's value is itself an encoded XML element, SecureSphere decodes it and parses its values and attributes as parameters.

Only valid XML is parsed. If the XML is not valid, then it is not parsed but understood by SecureSphere as text. For example, the string `<xxx<yyy` is not parsed. No error message is generated in the event of invalid nested XML.



Note:

- Nested parsing is single-pass, so it does not parse its own output. For example, `<` is parsed as `&lt` and not as <.
- The first two levels (`<xml><type>`) are removed and do not participate in the parameter name. `<type>` must be defined. If type is not SOAP, it must be something else.

You can enable Nested XML parsing for HTTP services in a service group, globally for all Web applications in the service. For more information, see the **XML Parsing** parameter in the **Operation** tab of the HTTP service definition in table **Web Services - Operations Parameters** found in Web Service Operations on page 65.

EXAMPLE: NESTED XML PARSING - DEPTH OF 1

If nested XML parsing is not enabled, SecureSphere represents the following XML:

```
<BookStore>
   <Publisher>&lt;Title&gt;I Love XML&lt;/Title&gt</Publisher>
</BookStore>
```

as the structure `BookStore/Publisher` with a value of: `<Title>I Love XML</Title>`

That is, the value of the element `Publisher` is understood by SecureSphere to be a text string, not XML.

However, if nested XML parsing is enabled, SecureSphere parses the string:

```
&lt;Title&gt;I Love XML&lt;/Title&gt
```

as: `<Title>I Love XML</Title>`

and represents the XML as:

`BookStore/Publisher/Title` with a value of: `I Love XML`

EXAMPLE: NESTED XML PARSING - DEPTH OF 2

XML Nested parsing represents the following XML:

```
<a><b>&lt;c&gt;&lt;d&gt;ABC&lt;/d&gt;&lt;/c&gt</b></a>
```

as: `a/b/c/d` with a value of `ABC`

Enabling Nested XML Parsing

To enable Nested XML parsing:

1. From the **Main** workspace, select **Sites**.
2. Select a server group.
3. Select the HTTP service in the server group.
4. In the **Operation** tab, enable **Parse Nested XML Values**.
5. Specify the **Maximum Parsing Depth**.
   The default value of **Maximum Parsing Depth** is 20.
6. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.



Note: After you enable Nested XML parsing, URLs that were previously in Protect mode may generate "Unknown Parameter" and "Missing Parameter" violations. For example, if a Protect mode URL has learned the parameter `a` and now receives the input "`<a><b>XXX</b></a>`", a "Missing Parameter" violation will be generated for the parameter `a`, and an "Unknown Parameter" violation will be generated for the new parameter `a/b`.

Correlated Queries Tab

The **Correlated Queries** tab displays a list of SQL Queries associated with this URL in the application's DB profile. When a user accesses a URL, this often leads to a query against the database. For example, when a user accesses a search URL on a commercial site, a query of the form "SELECT catid, catname FROM prodcat" may be performed.

SecureSphere is able to associate URLs with the SQL queries when the SecureSphere gateway is configured to monitor both Web traffic and database traffic.

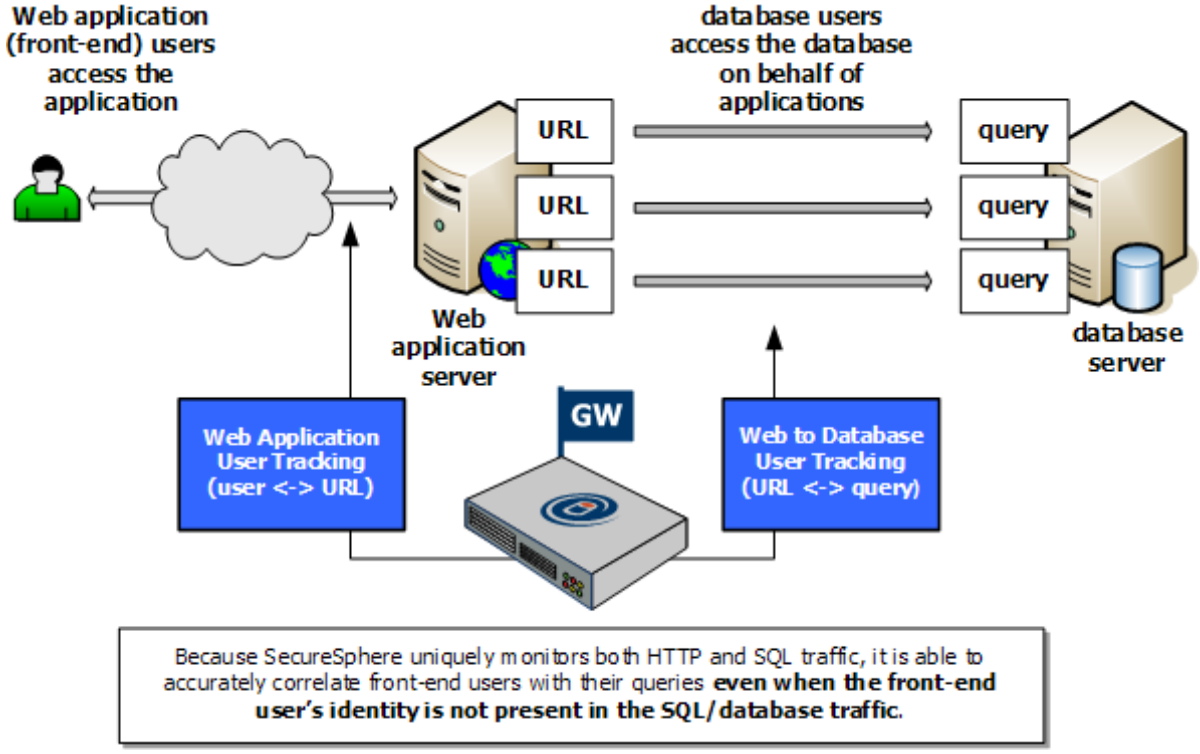


Figure 32: Web to Database User Tracking

SecureSphere Universal User Tracking (UUT) combines the information collected by the Web DB Correlation and the Web Application User Tracking features.

For more information, see Universal User Tracking.

Static URLs

You can choose whether to learn static URLs in the profile by selecting **Learn all URLs, except for static URLs without parameters** under **Profiling URLs without Parameters** in the web application's **Definitions** tab. Then the file types defined in the static file list specified in **Static Files** in the web service's **Definitions** tab (by default **Global Objects** > **File Type Groups** > **Default Static File List**) are not learned in the profile (that is, only dynamic URLs are learned) but signatures and protection engines (XSS, SQLi etc.) are applied to them.

You can enable/disable static URL profiling (with or without parameters) at any time. For more information, see the article in the Customer Portal titled **StaticURLs-How-to-Enable-Static-URL-Profiling-in-SecureSphere-v10-0**.

Broken Links and References

SecureSphere identifies broken links and references and displays them in the profile.

A broken link is a link from one of the application's learned hosts which points to a non-existing URL.

A broken reference is like a broken link, except that the link is from an external source.

- To view broken links, click **Display** and select **Broken Links Graph** from the menu.
- To view broken references, click **Display** and select **Broken References Graph** from the menu.

The broken links and references are listed together with the referer URLs.

The list of broken links can be useful in debugging applications.

URL Patterns

URL patterns enable you to reduce the number of URLs in the profile when a large number of URLs can be handled in the same way.

URL patterns describe a group of URLs which SecureSphere does not treat as distinct, different URLs but rather as repeated instances of the same URL. For example, if a site creates a folder for each of its users and all these folders contain files with the same name, the administrator can define a prefix or a suffix. A new URL which matches the pattern is not considered a new URL but another occurrence of the same URL pattern. All URLs which match the pattern are protected in the same way, for example, they share the same HTTP methods.

When a URL pattern is defined, it is in Protect mode by default. The administrator can then move the URL pattern to Learning mode, in which case SecureSphere will learn the parameters of the URL pattern until it moves it to Protect mode.

Prefix patterns are suitable for folders which contain a large number of files of the same type. For example, if the folder `/public/calculators/` contains many files with the same parameters and the same behavior, you can define `/public/calculators/` as a URL prefix and any file in the directory which matches the pattern is handled by SecureSphere in the same way.

Suffix patterns are suitable for:

- File types – (for example `aspx`)
- Specific files – (for example `order.asp`)
- A file name and part of its path – (for example `/public/print.asp` matches both `/scripts/public/print.asp` and `/home/public/print.asp`)

For information about viewing URL patterns, see Viewing a URL Pattern on page 264.



Note: Another way to reduce the number of learned URLs is to use plugins. For more information, see Custom Plugins on page 479.

Cookie Protection in Web Profiles

SecureSphere tracks the cookies which are sent to an application, that is, cookies for which it sees a SET command from the server, and protects against:

- **Cookie injection**: When a client sends the application a cookie it did not originally receive from the application.
- **Cookie tampering**: When a client alters a cookies that the application expects will not be altered.

In addition to tracking a Web application's cookies as part of the application's profile, SecureSphere provides two additional protection features:

- **Cookie signing**: SecureSphere adds a cryptographic signature to prevent tampering.
- **Cookie encryption**: SecureSphere encrypts cookies to prevent injection.

For information about viewing cookies, see Viewing and Editing Cookies on page 255.

Session Cookies

Session cookies are created by the Web application and sent to the client using the SET command. They are valid for the duration of the session and cannot be used outside the context of the session. SecureSphere sees the SET command for session cookies and monitors traffic to ensure that the cookie's contents are not changed by the client.

Persistent Cookies

Persistent cookies, like session cookies, are created by the Web application and sent to the client using the SET command. However, they are meant to be used not only in the context of the session in which they are created but also afterwards. The client saves the cookie locally and can reuse it at later time. This means that SecureSphere does not necessarily see the SET command for the cookie in each session in which the cookie is used.

For persistent cookies, there are two cases:

- **Signed cookies**: If the cookie is signed, SecureSphere verifies the signature.

  The cookie signing validation policy is not part of the Web profile but is defined under **Main** > **Policies** > **Cookie Signing Validation**.
- **Unsigned cookies**: If the cookie is not signed, SecureSphere assumes that the first cookie received is valid and monitors subsequent SET commands and the cookies' values.

The following table lists the situations where the different protection methods are effective.

Cookie Protection Mechanism

Supports/Survives	Profiling	Signing	Encryption	Explanation
Maintained through gateway boot	No	Yes	Yes	When a gateway fails, session information is lost, so when the gateway comes back up, SecureSphere does not know the initial value of the cookie and cannot monitor changes to that value. However, SecureSphere is able to verify the cookie signature and decrypt the cookie.
Maintained through gateway failover	No	Yes	Yes	When the failover gateway takes over, session information maintained by the failed gateway is unavailable to the failover gateway, so SecureSphere does not know the initial value of the cookie and cannot monitor changes to that value. However, the failover gateway is able to verify the cookie signature and decrypt the cookie.
Supported in Bridge mode	Yes	No	No	Cookie signing and encryption are available only in the Reverse Proxy modes.
Supported in Reverse Proxy mode	Yes	Yes	Yes	
Conceals cookie value	No	No	Yes	The value of an encrypted cookie is available only to SecureSphere, and the Web client cannot understand an encrypted cookie. In contrast, the value of a signed cookie is visible to the Web client.
Modifies data	No	Yes	Yes	
Modifies cookie value	No	No	Yes	
Cookie remains functional when gateway is in bypass	Yes	Yes	No	

Grace Period

When a gateway reboots, and when a failover gateway takes over from a failed gateway, there is a grace period during which SecureSphere treats session cookies and read-only parameters as valid the first time it sees them in the traffic, and assumes that their correct values are the first values it sees after the reboot or failover.

Cookie Status

A cookie's status can be any of the following:

- **Learn**: When SecureSphere sees a cookie for the first time, it sets its status to Learn. At a later time, SecureSphere switches the cookie to either Protect or Ignore, depending on whether the cookie's value changes during the session.

- **Protect**: If the cookie's value does not change, SecureSphere assumes that the application expects that the value of the cookie will not be changed (either by the Web client or by another Web application server) during a user session, and SecureSphere changes the cookie's status to Protect.

  For cookies whose status is Protect, SecureSphere protects against cookie tampering and optionally, against cookie injection.
- **Ignore**: If the cookie's value does change, SecureSphere assumes that the Web client or another Web application server changed the value, and SecureSphere changes the cookie's mode to Ignore.

  For cookies whose status is Ignore, SecureSphere does not protect against cookie tampering and cookie injection.

Over time, SecureSphere learns for which cookies it usually sees the SET command and for which it does not, for example, for persistent cookies. Another case in which SecureSphere might not see the SET command is when the SecureSphere was not running when the SET command was issued. Because of this, there is a grace period for a short time after SecureSphere starts during which no cookies are learned. For more information, see Grace Period on page 254.

Cookie Tampering

If the cookie's status is Protect, SecureSphere validates the cookie by inspecting the application's SET commands and monitoring the cookie's value throughout the session. If the cookie's value is changed by the user, SecureSphere generates a Cookie Tampering violation.

If the cookie's mode is Ignore, SecureSphere ignores changes to the cookie's value and does not generate Cookie Tampering violations.

Cookie Injection

If a protected cookie is received for which there was no previous SET command, SecureSphere generates a Cookie Injection violation.

Injected cookies are not learned.

Cookie Prefixes

Some cookies have dynamic names, for example, the Microsoft IIS ASPSESSION cookie. These cookies usually have a fixed prefix followed by a dynamic suffix, for example, ASPSESSIONIDFFEDSSE. For these cookies, enter only the fixed prefix as the cookie name, otherwise the number of cookies will grow too large. Monitoring is at the prefix level.

Viewing and Editing Cookies

You can edit a cookie's status and other data.

To edit a cookie's status and other data:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.

   If you do not see the application, either expand the server group or click **Change Application**.
3. Click Display and select **Cookies**.
4. Right-click the cookie and, from the menu, select **Edit Cookie**.
5. In the **Edit Cookie** window, edit the data in the window as required using the parameters in the table **Edit Cookie Window Parameters** below.
6. Click **Update**.

Edit Cookie Window Parameters

Field	Description
Name	The cookie's name. If **Prefix** is enabled, the cookie's prefix.
Status	Select Protect or Ignore. When creating a new cookie, you cannot add it to the learning list. Instead, add it to either the protect or ignore list, and then edit the cookie and change its status.
Injection	Select to enable checking for injection for this cookie.
Prefix	Select if **Name** is a prefix. All cookies whose names begin with the prefix are treated as a single cookie, see Cookie Prefixes on page 255.

To manually add a new cookie:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. Click Display and select **Cookies**.
4. Click .
5. In the **Create New Cookie** window, enter the data in the window as required (see the table **Edit Cookie Window Parameters** above).



Note: When you manually create a new cookie, you can set its status only to **Protect** or **Ignore**.

Displaying Information About a Profile's URLs

You can display a wide variety of information about a profile's URLs.

To display additional information about a profile's URLs:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. Click Display and select one of the following:

Web Profile Display Menu Options

Select	to display	See
URLs	A list of application's URLs.	Learned URLs on page 246
URL Patterns	A list of URL patterns defined for the application.	URL Patterns on page 252
Cookies	A list of the application's learned cookies and their statuses.	Cookie Protection in Web Profiles on page 252
Web Application User Tracking	A list of the application's login action URLs.	Web Application User Tracking on page 287
Learned Hosts	A list of the hosts on which the application's URLs are located.	
Susceptible Directories	A list of standard Web application directories (folders).	Configuring Susceptible Directories on page 426
Overall Status Graphs	A graphic display of the application's overall status: • URLs in Learning vs. URLs in Protection • URL Type Distribution (plain HTTP, SOAP and mixed) • Valid URLs vs. Broken Links vs. Broken References • Number of Added URLs Over Time	Tuning Web Profiles on page 257
Most Accessed URLs Graph	A graphic representation of the application's most frequently accessed URLs.	
Hit Distribution Graph	A graphic representation of the distribution of hits for the application's most frequently accessed URLs.	
Broken Links Graph	A list of all broken links (links in the application to a URL that does not exist) in the selected Web server groups and, for each link, the referer fields which linked to the URL. This information can be useful for debugging applications.	
Broken References Graph	A list of all broken links (external links, that is, links external the application to a URL that does not exist) in the selected web server groups and, for each link, the referer URL.	
Longest Response Times Graph	A graph showing the Learning mode URLs with the longest average response times, as measured from when profiling of this application began. This information can be useful for debugging, for example, database applications associated with the Web application.	

Tuning Web Profiles

You can tune a profile by performing any of the following actions:

- Viewing and Editing URLs on page 258
- Editing a URL's Methods on page 260
- Editing a URL Parameter Value Type on page 260
- Switching Between the URL Modes on page 261

- Switching the Mode of Multiple URLs on page 261
- Switching the Mode of a URL Directory which is also found in Switching the Mode of Multiple URLs on page 261
- Locking a URL Directory on page 262
- Locking a Web Profile on page 262
- Manually Adding a URL on page 263
- Deleting a URL on page 264
- Viewing a URL Pattern on page 264
- Creating a URL Pattern on page 264
- Creating a URL Pattern from an Existing URL on page 264
- Manually Adding a Learned Host on page 265



Note: For information on when configuration changes take effect, see Activating Settings on page 75.

Viewing and Editing URLs

You can view and edit a profile's URLs.

To view or edit a profile's URLs:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.

If you do not see the application, either expand the server group or click **Change Application.**

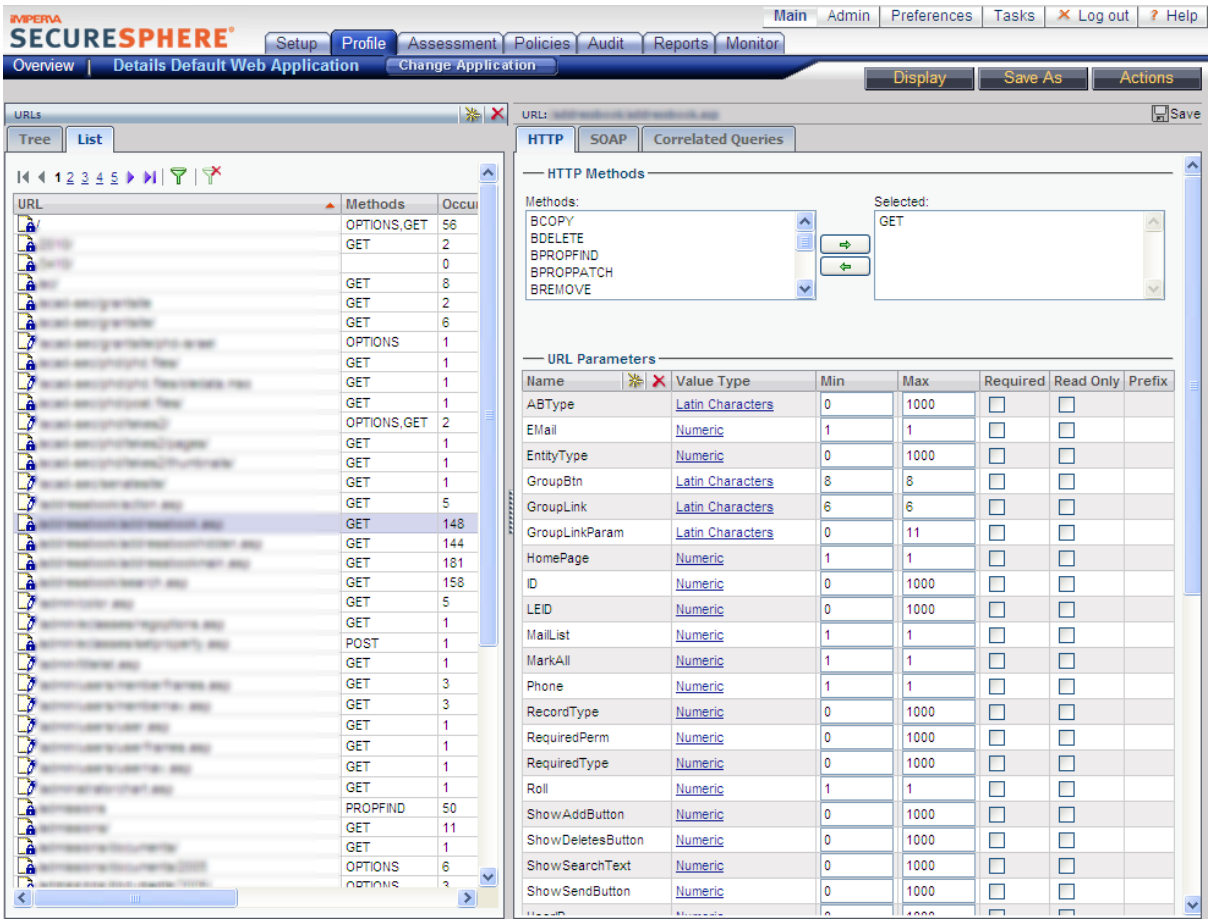


Figure 33: Web Profile URLs

3. In the **Tree** tab or the **List** tab, select a URL.
   The URL details are displayed in three tabs on the right of the window:
   - **HTTP**: see HTTP Methods and Parameters on page 247
   - **SOAP**: see SOAP Actions on page 248
   - **Correlated Queries**: see Correlated Queries Tab on page 250
4. If the URL is in Protect mode, you can modify the information in the tabs as follows:
   - In the HTTP tab, you can add or delete methods and modify parameters.
   - In the SOAP tab, you can enable and disable and modify the SOAP Action and its elements.

Deleting All URLs and Folders in a Profile

To delete all the URLs and folders in a profile:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. In the **Tree** tab or the **List** tab, select the root directory.
4. Right-click and select **Empty directory** from the menu.

Editing a URL's Methods

You can edit a URL's methods.

To edit a URL's methods:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. In the **Tree** tab or the **List** tab, select a URL.
4. Select a method from the **Methods** list or the **Selected** list.
5. Click the appropriate arrow to move the method to the other list.
6. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Editing a URL Parameter Value Type

You can edit a URL parameter's value type.

To edit a URL parameter's Value Type:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. In the **Tree** tab or the **List** tab, select a URL.
4. Click on the parameter's **Value Type** (for example, on the words **Latin Characters**).
5. In the **Configure Value Type** window:
   - Select **Custom Value Type** and select a **Parameter Type Configuration (**named regular expression) from the list.
     For example, if you have defined a regular expression describing a Social Security number in **Main** > **Setup** > **Global Objects** under **Web Related Objects** > **Parameter Type Configuration**, you can select that regular expression here. For more information about the use of regular expressions in SecureSphere, see Regular Expressions in SecureSphere on page 475.
     Or,
   - Select **Primary Value Type** and select a type, for example, **Numeric**, from the list.
6. Under **Additional Allowed Character Groups**, check the characters that are allowed, in addition to those included in the **Primary Value Type**.



Note: You can protect against attacks such as XSS (cross site scripting) and SQL injection by profiling the parameters and not allowing the special characters which are usually used in such attacks, for example: spaces, = , ; > and <.

7. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Switching Between the URL Modes

You can switch a URL's mode between Protect and Learning. For more information about URL modes, see Profile Modes on page 241.

To switch a URL's mode between Protect and Learn:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. In the **Tree** tab or the **List** tab, select a URL.
4. Right-click the URL or click Actions.
5. From the menu, select either **Switch to Protect** or **Switch to Learn**.
   When you switch a URL to learning mode, SecureSphere "forgets" the learned parameter values (but not the parameters themselves) and restarts the learning process from scratch.
6. Click **Save**.
   Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Switching the Mode of Multiple URLs

Youcan switch the mode of multiple URLs between Protect and Learning. For more information about URL modes, see Profile Modes on page 241.

To switch the mode of multiple URLs between Protect and Learn:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. In the **List** tab, use the filter to display only the URLs whose mode you want to change.
4. Select all the URLs.
5. Right-click on the URLs.
6. From the menu, select either **Switch to Protect** or **Switch to Learn**.
   When you switch a URL to learning mode, SecureSphere "forgets" everything it previously learned about the URL (parameters, maximum and minimum values, *etc.*) and restarts the learning process from scratch.
7. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

You can switch the mode of an entire URL directory.

To switch a URL directory's mode between Protect and Learn:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. In the **Tree** tab, right-click a directory.



Note: To switch the entire application, select the root directory.

4. From the menu, select **Switch Mode**.

5. In the **Switch Directory Status** window, select **All URLs in this directory** or **All URLs in this directory and in all subdirectories**.
6. Select **Protect** or **Learn**.
7. Click **Switch**. When you switch a URL to learning mode, SecureSphere "forgets" everything it previously learned about the URL (parameters, maximum and minimum values, *etc.*) and restarts the learning process from scratch.
8. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Locking a URL Directory

You can lock a URL directory, so that SecureSphere does not update the protected URLs in the directory. For more information, see Web APU (Automatic Profile Update) on page 279.

To lock a URL directory:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. In the **Tree** tab, right-click a tree entry.
4. From the menu, select **Lock Directory**.
5. In the **Lock Directory** window, select whether to lock the directory and optionally, its sub-directories, and whether to lock only the protected URLs in the directory and optionally, in its sub-directories.
   The difference between locking a directory and locking a URL is this:
   - If you lock a directory, no more URLs can be added to the directory.
   - If you lock a URL, the Automatic Profile Update feature does not update the URL, even if there is a match.

   For more information about the Automatic Profile Update feature, see Web APU (Automatic Profile Update) on page 279
6. Click **Lock**.
7. Click **Save**.
   Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Locking a Web Profile

You can lock a Web profile, so that SecureSphere does not update any of its protected URLs. For more information, see Web APU (Automatic Profile Update) on page 279.



Note: Locking a Web profile locks the URLs and directories, but does not lock the profile's cookies.

To lock a Web profile:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.

3. In **Tree** tab, right-click the root directory.

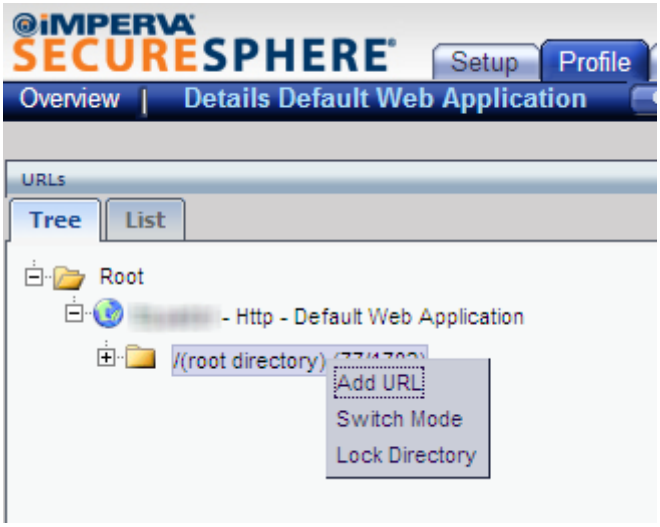


Figure 34: Locking a Web Profile

4. From the menu, select **Lock Directory**.
5. In the **Lock Directory** window, select all the checkboxes, that is, lock the root directory and its sub-directories, as well as the protected URLs in the root directory and its sub-directories.

   The difference between locking a directory and locking a URL is this:

   - If you lock a directory, no more URLs can be added to the directory.
   - If you lock a URL, the Automatic Profile Update feature does not update the URL, even if there is a match.

   For more information about the Automatic Profile Update feature, see Web APU (Automatic Profile Update) on page 279
6. Click **Lock**.
7. Click **Save**.

   Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Manually Adding a URL

You can manually add a URL to a Web profile.



Note: The limits defined in Maximum allowed URLs per application and in Maximum allowed URLs per directory in **Admin** > **System Definitions** > **Profile Size Limits** are not enforced when URLs are added manually, only when they are learned.

To add a URL:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.

   If you do not see the application, either expand the server group or click **Change Application**.
3. Click .
4. Enter the details of the new URL.
5. Click **Save**.

   Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Deleting a URL

You can manually delete a URL from a Web profile.

To delete a URL:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. In either the **Tree** tab or the **List** tab, right-click the URL you want to delete.
4. Select **Delete URL** from the menu.
5. Click **Save**.
   Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Viewing a URL Pattern

You can view the URL patterns for a Web profile.

To display the URL Patterns:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. Click **Actions**.
4. Select **URL Patterns**.

Creating a URL Pattern

You can create a URL pattern for a Web profile.

To create a URL Pattern:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. Click **Actions**.
4. Select **URL Patterns**.
5. Click .
6. In the **Create a New URL Pattern** window, enter the URL pattern and specify whether it is a prefix or a suffix.
7. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Creating a URL Pattern from an Existing URL

Youcan create a new URL pattern from an existing URL.

To create a new URL Pattern:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. In the **Tree** tab (but *not* in the **List** tab), select the URL from which to create the URL pattern.
4. Right-click the URL or click Actions.
5. Select **Save as Pattern** from the menu.

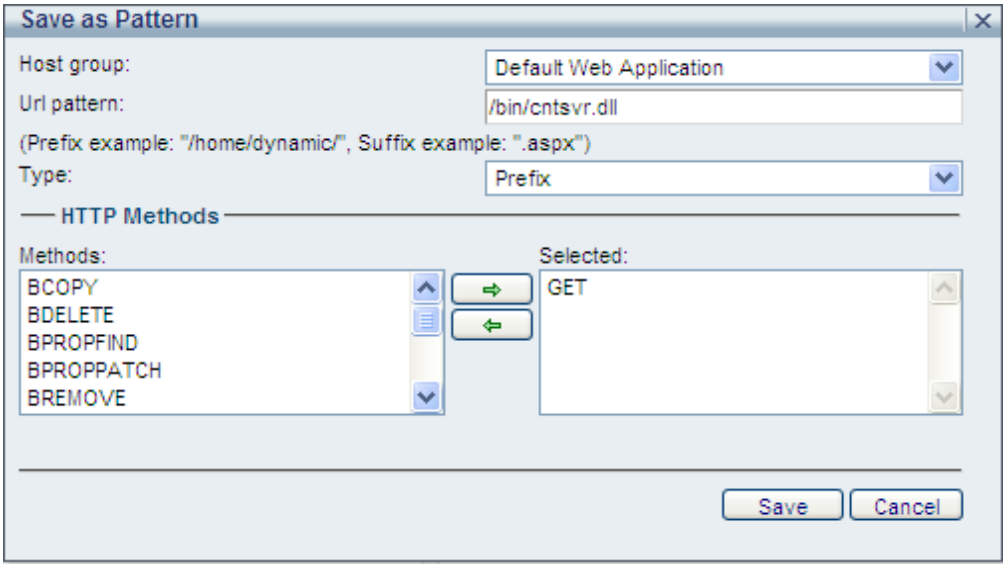


Figure 35: Save as Pattern window

6. In the **Save as Pattern** window, enter the data shown in the table below:
7. Click **Save**.
   Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Parameter	Description
Host group	The default host group shown is the current Web application, and you can change it as required.
Url pattern	The default pattern shown is derived from the URL, and you can change it as required.
Type	Select **Prefix** or **Suffix**.
HTTP Methods	Move methods between the **Methods** and **Selected** lists as required.

Manually Adding a Learned Host

You can add manually a learned host to a Web profile.

To manually add a learned host to a Web profile:

1. In the **Main** workspace, select Profile.

2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. Click Display.
4. Select **Learned hosts** from the menu.
   A list of the profile's learned hosts is displayed.
5. Click .
6. In the **Add New Host** window, enter a host name or an IP address.
7. Click **Create**.
8. Click **Save**.
   Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Deleting All URLs and Folders in a Profile

To delete all the URLs and folders in a profile:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, or click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. In the **Tree** tab or the **List** tab, select the root directory.
4. Right-click and select **Empty directory** from the menu.

Web Profile Graphs

This section describes the graphical information that is available in Web profiles, and includes:

Overall Graphs on page 266
Access Patterns on page 267

Overall Graphs

The following graphs are displayed when you select **URL Insight Graphs** from the **Web Profile** menu as shown in Web Application Profiles on page 245.

Graph	Displays
URLs in Learning vs. URLs in Protection	Shows the distribution of Learning and Protected URLs in the profile.
Valid URLs vs. Broken Links vs. Broken References	Shows the distribution of valid URLs, broken URLs and broken references in the profile.
URL Type Distribution	Shows the distribution of HTTP URLs, SOAP URLs and mixed URLs in the profile.
Number of Added URLs Over Time	Shows the number of URLs added to the profile over time.
10 URLs With Longest Average Response Time	Shows 10 URLs with the longest response time, and lists the minimum, average and maximum response time for these URLs.

Access Patterns

The following graphs are displayed when you select **Access Patterns** from the **Web Profile** menu as shown in Web Application Profiles on page 245.

Graph	Displays
URLs Hit Distribution	Shows the distribution of URL hits.
Most Accessed URLs	Shows, in graphic and tabular form, the URLs most frequently accessed.

Web Profile Optimization

This section describes the Web profile optimization feature, and includes the following:

Web Profile Optimization Overview on page 267
Web Profile Optimization Issues on page 268

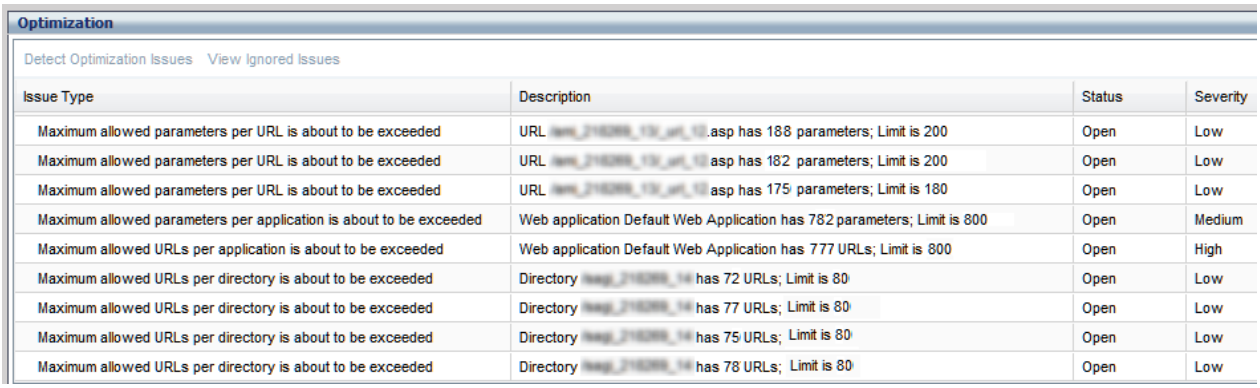
Web Profile Optimization Overview

As time goes by, profiles can sometimes grow unwieldy, that is, they become filled with URLs and parameters that can be removed or combined without compromising security. Also, it can happen that the default maximum of URLs and parameters defined for the profile are insufficient, in which case unnecessary alerts are generated. Reducing the size of a profile and eliminating unnecessary alerts can improve throughput, sometimes dramatically.

SecureSphere can help administrators optimize Web profiles by identifying issues and suggesting a resolution.

A warning icon displayed next to a web application in the profile overview tree indicates that the application's profile has optimization issues. The warning icon "floats" up the hierarchy (service, server group, site *etc.*), but only the icon next to the application is clickable and takes you to the **Optimization Issues** view.

When you select **Optimization** from the Web Profile menu as can be seen in Web Application Profiles on page 245, the optimization issues are displayed in the **Optimization**.



Issue Type	Description	Status	Severity
Maximum allowed parameters per URL is about to be exceeded	URL [redacted].asp has 188 parameters; Limit is 200	Open	Low
Maximum allowed parameters per URL is about to be exceeded	URL [redacted] asp has 182 parameters; Limit is 200	Open	Low
Maximum allowed parameters per URL is about to be exceeded	URL [redacted] asp has 175 parameters; Limit is 180	Open	Low
Maximum allowed parameters per application is about to be exceeded	Web application Default Web Application has 782 parameters; Limit is 800	Open	Medium
Maximum allowed URLs per application is about to be exceeded	Web application Default Web Application has 777 URLs; Limit is 800	Open	High
Maximum allowed URLs per directory is about to be exceeded	Directory [redacted] has 72 URLs; Limit is 80	Open	Low
Maximum allowed URLs per directory is about to be exceeded	Directory [redacted] has 77 URLs; Limit is 80	Open	Low
Maximum allowed URLs per directory is about to be exceeded	Directory [redacted] has 75 URLs; Limit is 80	Open	Low
Maximum allowed URLs per directory is about to be exceeded	Directory [redacted] has 78 URLs; Limit is 80	Open	Low

Figure 36: List of Optimization Issues

Web Profile Optimization Issues

This section describes how to detect and respond to optimization issues identified by SecureSphere, and includes the following:

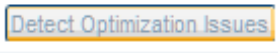
Detecting Optimization Issues on page 268
Reviewing Optimization Issues on page 269
Automatically Adjusting the Web Profile on page 273
Ignoring Optimization Issues on page 273
Regenerating the List of Optimization Issues on page 274

Detecting Optimization Issues

SecureSphere automatically runs a detection job every night, in which it detects optimization issues for all web application profiles.



Note: Parameters relating to optimization issues are defined in **Admin** > **System Definitions** > **Profile Optimization**.

To display optimization issues for the selected web profile or to refresh a displayed list, click 

above the list in the **Optimization** pane as shown in Web Profile Optimization Issues on page 268, and in the **Detect Optimization Issues** window, click **Run**.

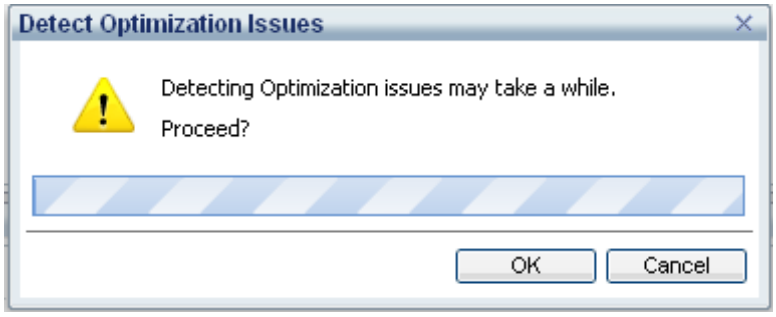


Figure 37: Detect Optimization Issues Window

The process may take some time, depending on the profile's size and complexity, and it may adversely affect performance, so you should run it when traffic is low.

Reviewing Optimization Issues

The following table lists the optimization issues which SecureSphere detects in a profile.

Optimization Issue	Description
Large number of similar alerts	The alert is displayed in the **Description** column in the **Optimization** pane. For example, the problem may be that a large number of parameter type violations have been found, which indicates that the parameter type may need to be changed.
Maximum allowed parameters per URL is about to be exceeded	The URL is displayed in the **Description** column in the **Optimization** pane, along with the number of parameters found and the limit defined by Maximum allowed parameters per URL in **Admin** > **System Definitions** > **Profile Size Limits**. For information about the default maximum allowed value, see the Systems Definitions chapter in the **SecureSphere Administration Guide**.
Maximum allowed parameters per application is about to be exceeded	The application's name is displayed in the **Description** column, along with the number of parameters found and the limit defined by Maximum allowed parameters per application in **Admin** > **System Definitions** > **Profile Size Limits**. For information about the default maximum allowed value, see the Systems Definitions chapter in the **SecureSphere Administration Guide**.
Maximum allowed URLs per application is about to be exceeded	The application's name is displayed in the **Description** column in the **Optimization** pane), along with the number of URLs found and the limit defined by **Maximum allowed URLs per application** in **Admin** > **System Definitions** > **Profile Size Limits**. For information about the default maximum allowed value, see the Systems Definitions chapter in the **SecureSphere Administration Guide**.
Maximum allowed URLs per directory is about to be exceeded	The directory's name is displayed in the **Description** column in the **Optimization** pane), along with the number of URLs found and the limit defined by Maximum allowed URLs per directory in **Admin** > **System Definitions** > **Profile Size Limits**. For information about the default maximum allowed value, see the Systems Definitions chapter in the **SecureSphere Administration Guide**.

When you select an optimization issue in the list, as can be seen in the Optimization window shown in Web Profile Optimization Overview on page 267, its details are displayed in the **Recommended Adjustments** pane, as shown below.

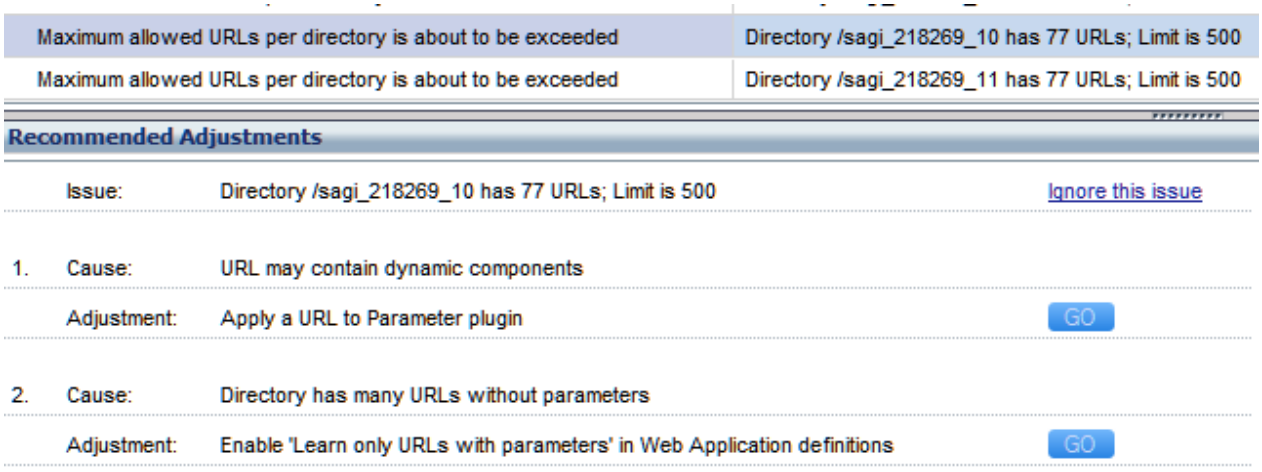


Figure 38: Recommended Adjustments

For each optimization issue that has not been ignored or resolved, SecureSphere displays a probable cause and in many cases, suggests an adjustment to the profile to address the issue.

Optimization Issue	Recommended Adjustment
Large number of similar alerts	For example, the issue might indicate that the **Cause** is a parameter type violation, and **Recommended Adjustment** would be to change the parameter type defined in the profile.
Maximum allowed parameters per URL is about to be exceeded	If no Parameter Modification plugin is applied to this web service, SecureSphere recommends that you apply one. Alternatively, you might consider changing the limit defined by Maximum allowed parameters per URL in **Admin** > **System Definitions** > **Profile Size Limits.** You might also consider adding a parameter prefix to the profile (see URL Patterns on page 252 for more information).
Maximum allowed parameters per application is about to be exceeded	If no Parameter Modification plugin is applied to this web service, SecureSphere recommends that you apply one. Alternatively, you might consider changing the limit defined by Maximum allowed parameters per application in **Admin** > **System Definitions** > **Profile Size Limits.** You might also consider adding a parameter prefix to the profile (see URL Patterns on page 252 for more information.
Maximum allowed URLs per application is about to be exceeded	If this is caused by URLs with dynamic components which result in many functionally identical URLs being treated as unique URLs, then the recommended adjustment might be to apply a URL to parameter plugin. For more information, see URL to Parameter Plugin on page 481. The cause may also be that the application has a large number of URLs without parameters. In this case, the recommended adjustment would be to enable **Learn only URLs with parameters** in the application's **Definitions** tab. You will have the option of deleting from the profile the already-learned URLs without parameters.
Maximum allowed URLs per directory is about to be exceeded	If this is caused by URLs with dynamic components which result in many functionally identical URLs being treated as unique URLs, then the recommended adjustment might be to apply a URL to parameter plugin. For more information, see URL to Parameter Plugin on page 481. The cause may also be that the application has a large number of URLs without parameters. In this case, the recommended adjustment would be to enable **Learn only URLs with parameters** in the application's **Definition** tab. You will have the option of deleting from the profile the already-learned URLs without parameters.



Note: Parameters relating to optimization issues (Alert Occurrences, Alert Timeframe in Hours and Size Limit) are defined in **Admin** > **System Definitions** > **Profile Optimization**. For more information, see the Systems Definitions chapter in the **SecureSphere Administration Guide**.

Automatically Adjusting the Web Profile

When SecureSphere recommends an adjustment in the **Recommended Adjustments** pane as shown in Reviewing Optimization Issues on page 269, it often displays a GO button at the right end of the line. If you click the GO button, a window will be opened in which you can make the recommended adjustment.

For example, if you click the GO button for the second adjustment in , the following **Apply Adjustment** window will be displayed.

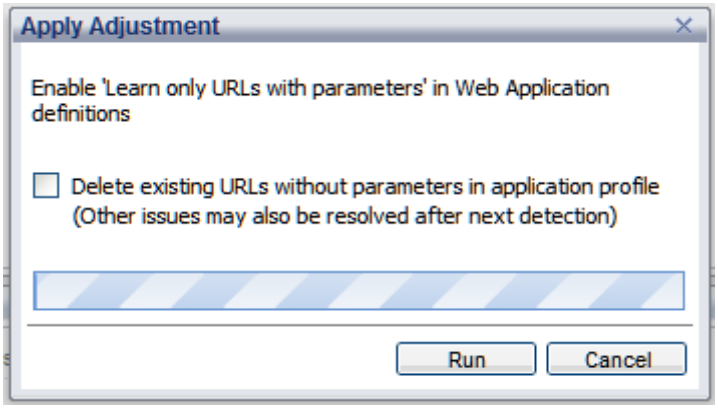


Figure 39: Apply Adjustment Window

If you click **Run**, this will enable the parameter **Learn only URLs with parameters** in the application's **Definitions** tab, exactly as if you had done this directly.

You can also delete from the profile all URLs without parameters by checking **Delete existing URLs without parameters**.

In addition to recommending adjustments, when an issue can be resolved only through a manual, possibly complex procedure that SecureSphere does not perform automatically, SecureSphere offers a "Tip" describing how the user can resolve the issue.

Ignoring Optimization Issues

It may happen that SecureSphere has identified as a profile violation behavior which is in fact normal for the application, or that a profile has been manually tuned and certain outstanding issues are no longer relevant. In these cases, you can ignore an issue by selecting it in the list and, in the **Recommended Adjustments** pane, clicking Ignore this issue.

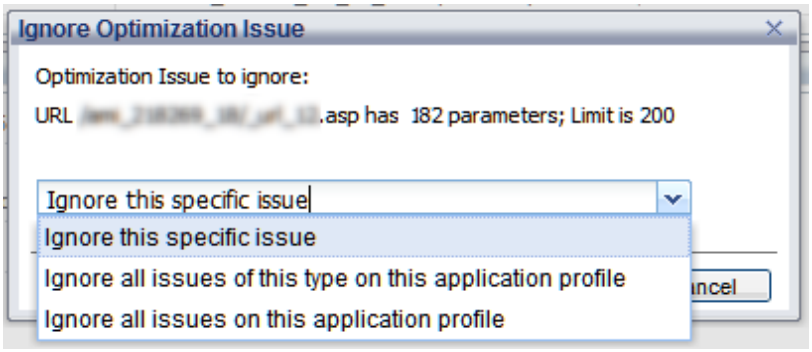


Figure 40: Ignore Optimization Issue Window

The **Ignore Optimization Issue** window is displayed, in which you can choose to:

- Ignore this specific issue

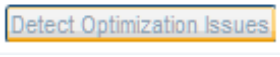
- Ignore all issues of this type on this application profile
- Ignore all issues on this application profile

You can view the profile's ignored issues by clicking 

above the list. The **Ignored Optimization Issues** window is displayed, in which the ignored issues are listed.

In the **Ignored Optimization Issues** window, you can select an issue and remove it from the list of ignored issues by selecting it and clicking **Undo Ignore**. The issue is removed from the list in the **Ignored Optimization Issues** window, but its **Status** in the **Optimization** pane as shown in Web Profile Optimization Overview on page 267 remains "Ignored".

Regenerating the List of Optimization Issues

The list of issues can be regenerated by clicking 

. It is also regenerated by the automatic detection SecureSphere runs overnight. Clicking the browser's **Refresh** button does not regenerate the list.

If an issue has been resolved manually, its status will be updated only after the list is regenerated.

Web Profile Policy

Each Web application has exactly one Web profile associated with it, and that profile has no more than one profile policy associated with it, which is applied only to the profile URLs which are in Protect mode. You can apply additional policies (which respond to different kinds of events, not to profile deviation events) to the application, but you cannot apply more than one profile policy to a profile.

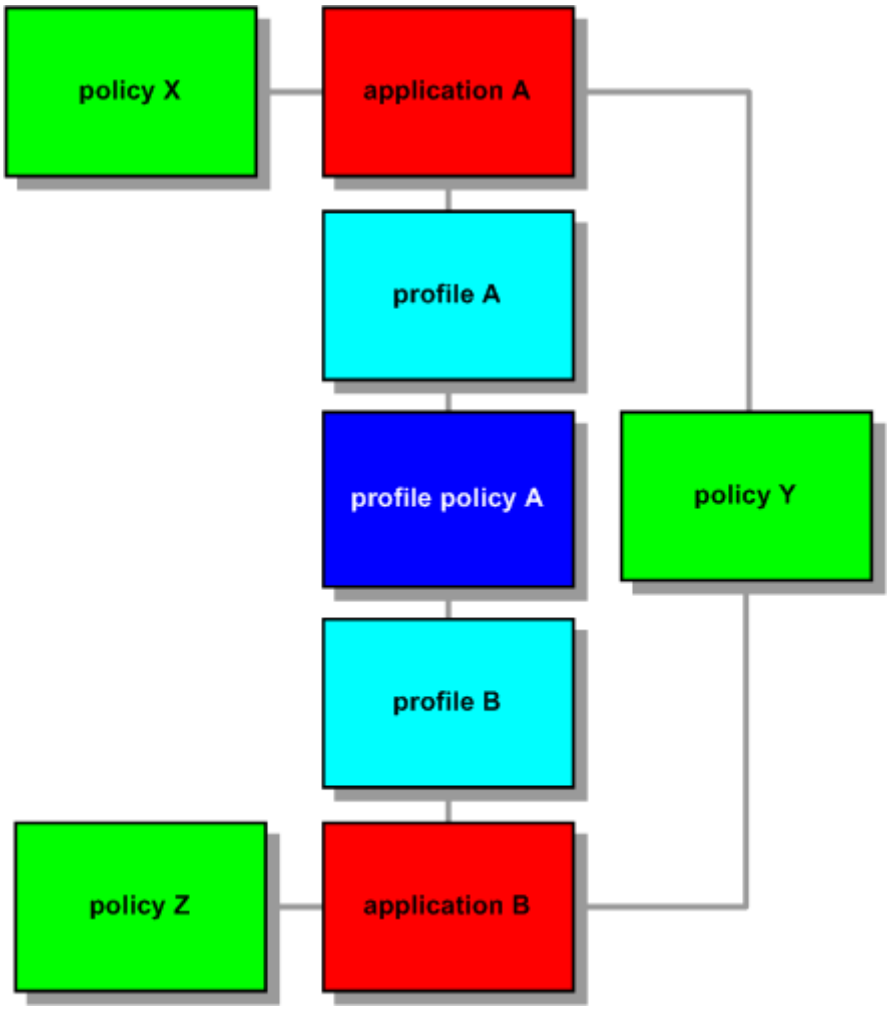


Figure 41: Web Application - Web Profile - Policy Hierarchy

Application A has one profile (profile A), which in turn has one profile policy (profile policy A). In addition, application A has two more policies (policy X and policy Y).

Similarly, application B has one profile (profile B), which in turn has one profile policy (profile policy A). In addition, application B has two more policies (policy Y and policy Z).

Note: It is possible to "unapply" a profile policy from a profile, so that the profile has no profile policy associated with it. In this case the profile becomes essentially useless.

Tuning a Web Profile Policy

You can tune a Web Profile policy by performing any of the following actions:

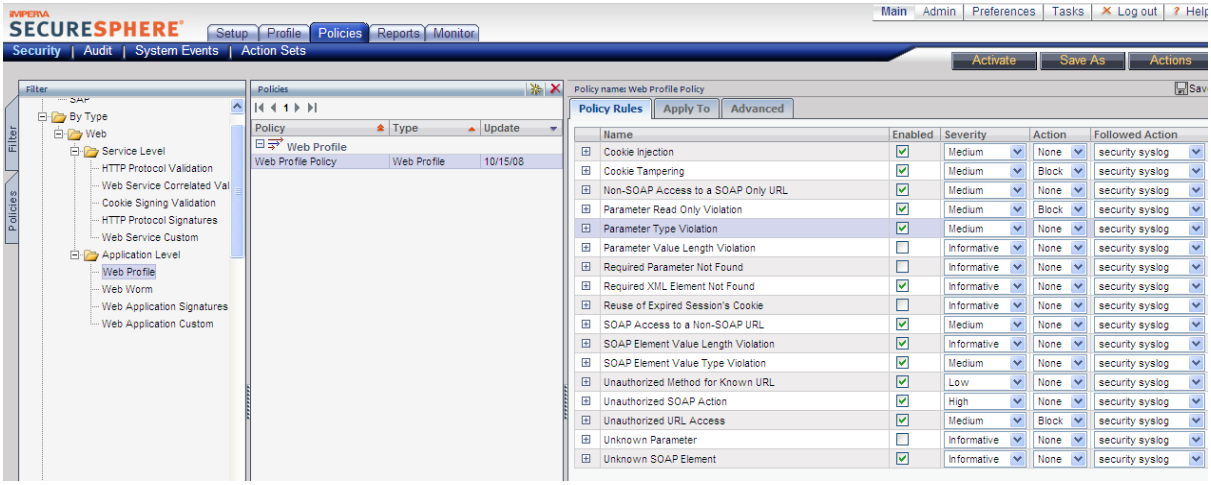
- View a policy, see Viewing a Web Profile Policy on page 276
- Edit a policy, see Modifying Policies in the Security Window on page 176
- Create a new policy, see Adding New Security Policies on page 179
- Change policy exceptions, see Security Policy Exceptions on page 173
- Apply to an application, see Applying Policies on page 318
- Disabling Automatic Web Profiling on page 278

Note: For information on when configuration changes take effect, see Activating Settings on page 75.

Viewing a Web Profile Policy

To view or edit a Web profile policy:

1. In the **Main** workspace, select **Policies** > **Security** and under **Web**, select **Application Level** > **Web Profile**. The Web Profile Policy appears in the Policies pane.



   Alternatively, browse to the Web application through **Main** > **Setup** > **Sites** and click on the Edit icon ( ) next to the Web Profile Policy in the **Applied Policies** tab. In this case, the policy is displayed in the **Policy Details** window.

2. In the middle pane, select a profile to display its policy rules and their parameters in the right pane.

   Alternatively, browse to the Web application through **Main** > **Setup** > **Sites** and click on the Edit icon ( ) next to the Web Profile Policy in the **Applied Policies** tab. In this case, the policy is displayed in the **Policy Details** window and the **Apply To** tab is Read Only.

3. For each policy rule, edit the parameters as required, as shown in the table below.

Note: These violations are invoked only for URLs and cookies which are in Protect mode.

Web Policy Rules

Parameter	Description
Cookie Injection	For information about when SecureSphere invokes a Cookie Injection violation, see Cookie Injection on page 255. **Defaults**: Enabled; Severity - Medium; Action - None.
Cookie Tampering	For information about when SecureSphere invokes a Cookie Tampering violation, see Cookie Status on page 254. **Defaults**: Enabled; Severity - Medium; Action - None.
Non-SOAP Access to a SOAP Only URL	This violation is invoked if an HTTP request to a SOAP URL does not include a SOAP message. **Defaults**: Enabled; Severity - Medium; Action - None.
Parameter Read Only Violation	This violation is invoked if the client altered a read-only parameter. For more information, see table **HTTP Tab Parameters** shown in HTTP Methods and Parameters on page 247. This rule must be enabled during Learning mode, otherwise SecureSphere does not learn which parameters are read-only. The following additional parameters are defined for this rule: **Issue Anomaly For Correlated Parameter Tampering**: Invoke a violation if the input does not conform to the learned pattern of combinations of read-only parameters. This violation is useful for applications in which only specific combinations of the read-only parameters are allowed. **Issue Anomaly For Requests Without Session**: Invoke a violation if a client accesses a URL with read-only parameters without first opening a valid session. **Issue Anomaly For Response Evasion**: Invoke a violation if SecureSphere detects that a read-only parameter was sent by a URL other than that detected at the beginning of the session. **Defaults**: Enabled; Severity - Informative; Action - None.
Parameter Type Violation	This violation is invoked if a parameter does not match its learned type, for example, its learned type is numeric but it contains alphabetic information. This violation may indicate an SQL Injection or XSS attack. See Editing a URL Parameter Value Type on page 260 for information on how to protect against these kinds of attacks. **Defaults**: Enabled; Severity - Medium; Action - None.
Parameter Value Length Violation	This violation is invoked if a parameter's length does not conform to its profiled length (that is, it is either too long or too short). **Defaults**: Disabled; Severity - Informative; Action - None.
Required Parameter Not Found	This violation is invoked if a required parameter is not present. **Defaults**: Disabled; Severity - Informative; Action - None.
Required XML Element Not Found	This violation is invoked if a required XML element is not present. **Defaults**: Enabled; Severity - Informative; Action - None.
Reuse of Expired Session's Cookie	This violation is invoked if the client sends a protected cookie whose session has expired. **Defaults**: Disabled; Severity - Informative; Action - None.

Parameter	Description
SOAP Access to a Non-SOAP URL	This violation is invoked if an HTTP request includes a SOAP message for a URL which is not SOAP-enabled. **Defaults**: Enabled; Severity - Medium; Action - None.
SOAP Element Value Length Violation	This violation is invoked if an XML element's length does not conform to its profiled length (that is, it is either too long or too short). **Defaults**: Enabled; Severity - Informative; Action - None.
SOAP Element Value Type Violation	This violation is invoked if an XML element does not match its learned type, for example, its learned type is numeric but it contains alphabetic information. **Defaults**: Enabled; Severity - Medium; Action - None.
Unauthorized Method for Known URL	This violation is invoked if a URL is accessed with a method which does not conform to the profile. **Defaults**: Enabled; Severity - Low; Action - None.
Unauthorized SOAP Action	This violation is invoked if a SOAP-enabled URL is accessed with a SOAP action which does not conform to the profile. **Defaults**: Enabled; Severity - High; Action - None.
Unauthorized URL Access	This violation is invoked if an unprofiled URL (that is, a URL which is not in the profile) in a locked directory is accessed. **Defaults**: Enabled; Severity - Medium; Action - None.
Unknown Parameter	This violation is invoked if a URL is accessed with a parameter which is not in the profile. **Defaults**: Disabled; Severity - Informative; Action - None.
Unknown SOAP Element	This violation is invoked if a SOAP URL is accessed with an element which is not in the profile. **Defaults**: Enabled; Severity - Informative; Action - None.

Disabling Automatic Web Profiling

You can disable further learning for all Web applications to which the Web profile policy applies. When learning is disabled, the profile is "frozen," that is, there is no more learning, but the profile continues to be enforced. Any elements of the profile which are in Protect mode continue to be protected.

If you want to stop enforcement of the profile altogether, in addition to disabling learning, then "unapply" the profile policy from the Web application.

To disable automatic profiling:

1. In the **Main** workspace, select **Policies** > **Security**.
2. In the left pane, the filter pane, under **Web**, select **Application Level** > **Web Profile**.
3. Select the policy in the middle pane of the **Policies** window as described in Viewing a Web Profile Policy on page 276.
4. In the **Advanced** tab, check **Disable learning engine**.
5. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Web APU (Automatic Profile Update)

Applications change with time and a profile that does not change to reflect the application's new behavior soon becomes outdated and its value diminishes as it generates an increasing number of false positives. Requiring administrators to manually update the profile is expensive and error-prone. Instead, SecureSphere implements the only practical solution: automatic updates based on application traffic.

The following topics provide information about working with the Web Automatic Profile Update:

- Understanding Web APU Profiling Dynamics on page 279
- Preparing for Change on page 281
- Tuning a Web Profile APU Policy on page 281

Understanding Web APU Profiling Dynamics

Application changes are initially perceived by SecureSphere as deviations from the profile. If these deviations are regular and uniform, that is, if they occur many times, over a relatively long period of time, and originate from multiple sources, then SecureSphere assumes that these events are not violations but rather reflect the changed application and should be incorporated into the profile.

The Web APU (Automatic Profile Update) feature dynamically modifies the protected elements of a Web profile so that the profile remains up-to-date even as the Web application changes. The Web Automatic Profile Update policy rules are part of the profile's policy, and are displayed in the **Advanced** tab of the **Profile Policy** window, see Tuning a Web Profile APU Policy on page 281.



Note: Administrators should periodically monitor the SecureSphere logs and profile alerts, and tune the profile and its policy accordingly.

The administrator has full control over the profile updating process, and can even disable the feature. Sensitive URLs such as login pages can be locked so that they are not changed. Individual violations can be disabled so that some profile parameters are updated while others are not.

You can view a rule's parameters by expanding the rule, that is, by clicking on the plus sign to the left of the rule.

If one of the violations listed in table **Web APU Profile Rules** below occurs **N** times over a period of **H** consecutive hours, and originates from **S** IP addresses, where **N**, **H** and **S** are individually defined for each violation listed in this table, then SecureSphere concludes that the application has changed and updates the profile according to the new information.



Note: S can be either an IP address or an application session identifier.
If any of these violations occurs several times during a single hour, this counts as 1 hour of the **H** total consecutive hours.

The default values for **N**, **H** and **S** are 50, 12 and 50, respectively. So the default threshold is as follows:

If one of the violations listed in table **Web APU Profile Rules** below occurs at least 50 times over a period of 12 consecutive hours, and originates from at least 50 IP addresses, then the application is considered to have changed, the profile is updated accordingly, and a system event is generated.



Note: If the Web application server is behind a proxy, then SecureSphere sees all traffic as coming from the proxy's IP address, so the default threshold of 50 source IP addresses is never reached. In this case, you should change the default to 1.
For information on how to configure these parameters for connections from a load balancer, see the **SecureSphere Administration Guide**.

Changes to the profile take effect as soon as they are downloaded to the SecureSphere gateway.

The following table shows how SecureSphere changes a profile in response to specific violations.

Web APU Profile Rules

Violation	Profile Change
Cookie Injection	SecureSphere stops generating Cookie Injection violations if this cookie is seen from at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled
Cookie Tampering	SecureSphere moves a cookie from the protected list to the ignored list if the cookie is seen with a changed value in traffic from at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled
Parameter Read-Only Violation	SecureSphere clears a parameter's read-only attribute if the parameter's value is changed by at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled
Parameter Type Violation	SecureSphere adds a character group to a parameter if the character group is seen in the parameter's value from at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled
Parameter Unknown	SecureSphere adds a parameter to a URL if the parameter is seen in traffic from at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled
Parameter Value Length Violation	SecureSphere changes a parameter's minimum or maximum length if the parameter is seen with a longer or shorter value from at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled
Required Parameter Not Found	SecureSphere clears a parameter's Required attribute if the parameter is not present in traffic from at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled
Required XML Element Not Found	SecureSphere clears an XML element's Required attribute if the element is not present in traffic from at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled
Reuse of Expired Session's Cookie	SecureSphere moves a cookie from the protected list to the ignored list if the cookie is seen as a re-use of an expired session cookie in traffic from at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled
Unauthorized Method for Known URL	SecureSphere adds an HTTP method to a URL if the method is seen in traffic from at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled

Violation	Profile Change
Unauthorized SOAP Action	SecureSphere adds a SOAP action to a URL if the SOAP action is seen in traffic from at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled
Unknown XML Element	SecureSphere adds an XML element to a SOAP action if the XML element is seen in traffic from at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled
XML Element Value Length Violation	SecureSphere changes an XML element's minimum or maximum length if an XML element is seen with a longer or shorter value in traffic from at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled
XML Element Value Type Violation	SecureSphere adds a character group to an XML element if the character group is seen in the XML element's value in traffic from at least S different sources at least N times during a period of H consecutive hours. **Default**: enabled

Preparing for Change

An administrator has two strategies available to simplify the profile update process in response to change. Just before the change is scheduled to take effect, the administrator can:

- Switch the profile to Learning mode.

  To do this, display the URL tree in tree mode, right-click the root directory and in the **Switch Directory Status** window, select **All URLs in this directory and in all subdirectories** and click **Switch**.

  The new application behavior will be learned, and the profile will return to Protect mode in due course.
- Temporarily reduce the values of the parameters in the **Advanced** tab of the **Profile Policy** window.

For more information see Viewing a Web Profile Policy on page 276.

This will accelerate the profile update process. The administrator should take care to restore the previous values once the profile has been updated.

Tuning a Web Profile APU Policy

You can tune a Web Profile APU policy by performing any of the following actions:



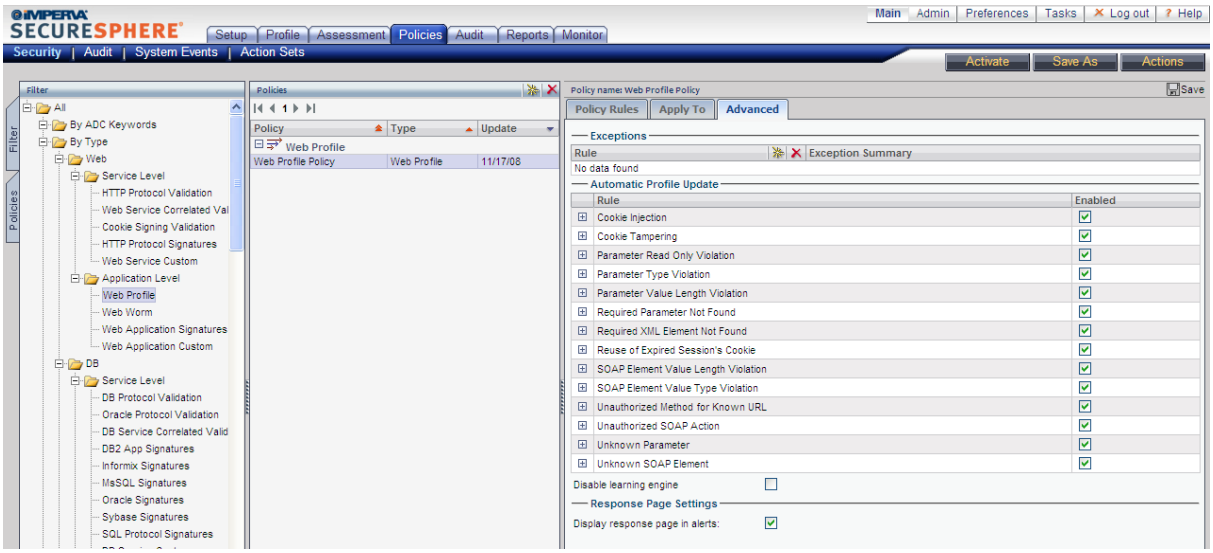
Note: For information on when configuration changes take effect, see Activating Settings on page 75.

Viewing a Web Profile Automatic Update Policy 282
Editing a Web Profile APU Policy Rule 282

Viewing a Web Profile Automatic Update Policy

To view or edit a Web profile's Automatic Update policy:

1. In the **Main** workspace, select **Policies** > **Security**.
2. In the left pane, the filter pane, under **Web**, select **Application Level** > **Web Profile**.
3. In the middle pane, select a policy to display in the right pane.
4. Click the **Advanced** tab.



5. To view a rule's parameters, expand the rule, that is, click on the plus sign to the left of the rule.



Note: The **Exceptions** section in the **Advanced** tab relates to the profile policy, not to the APU policy.

Editing a Web Profile APU Policy Rule

To edit a Web profile APU policy rule:

1. In the **Main** workspace, select **Policies** > **Security**.
2. In the left pane, the filter pane, under **Web**, select **Application Level** > **Web Profile**.
3. Select the policy in the middle pane of the **Policies** window.
4. In the **Advanced** tab, click the plus sign to the left of the rule and edit its parameters, listed in table Web APU Profile Rules shown in Understanding Web APU Profiling Dynamics on page 279.

   You can control the thresholds that cause different APU rules to be executed by changing a rule's parameters. Lowering the values of the rule's parameters, even to zero, causes the profile to be changed more quickly. Raising the values causes the profile to be changed more slowly.
5. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Web Profile System Definitions

There are groups of relevant properties regarding web profiles that can be configured from the **Admin** workspace, as follows:

- Protect Mode Thresholds (Web)**:** Located under **Admin > System Definitions > Switching to Protect mode Thresholds (Web)**
- Profile Size Limits**:** Located under **Admin** > **System Definitions** > **Profile Size Limits**

For information about these parameters, see **SecureSphere Administration Guide**.



Note: For information on when configuration changes take effect, see Activating Settings on page 75.

CHAPTER 10

Universal User Tracking

One of the primary requirements of any application or database audit process is validating appropriate user accountability. This is often a difficult task when users access databases via an application, because connections are pooled by the application server into a single connection to the database. Typical database auditing systems do not consistently link database activity with specific users when connection pooling is in use because only the application's login name is recorded.

SecureSphere's Universal User Tracking (UUT) feature overcomes these difficulties and enhances security and auditing by analyzing traffic to accurately associate application front end users with applications and database queries. SecureSphere tracks individual user connections, not just application logins, to provide full database audit accountability.

SecureSphere's Universal User Tracking technology makes it possible to hold individual users accountable for their actions.

User identification is provided in alerts and audit reports, and security policies to restrict access to sensitive databases can be enabled on a per-user basis. UUT provides out-of-the-box support for enterprise applications such as Oracle E-Business Suite, SAP, PeopleSoft and others, as well as custom and internally developed applications.

This section describes the Universal User Tracking feature, including:

Understanding Universal User Tracking 285
Web Application User Tracking 287
Web to Database User Tracking 295
SQL Connection User Tracking 300
Direct User Tracking 305

Understanding Universal User Tracking

Universal user tracking involves a number of aspects that come together to enable you to track a user from application front end user to application and database. The following topics provide information regarding general UUT components, architecture and more:

- Understanding UUT Components on page 285
- Universal User Tracking Architecture on page 286
- DB Profiling and UUT on page 286

Understanding UUT Components

The following table lists the various components used in Universal User Tracking.

To see how the various components work together, see Universal User Tracking Architecture on page 286.

Universal User Tracking components

component	function	type of applications
Application User Tracking (AUT)	AUT tracks users who connect indirectly to the database through an application.	
Web Application User Tracking	Identifies the user logging in to the Web application and tracks the user, session and IP address.	All HTTP-based and database applications.
Web to Database User Tracking	Tracks Web application users' interactions with the application through to their database queries.	Working together, Web Application User Tracking and Web to Database User Tracking track HTTP-based application users and correlates the URLs they access with their database queries. Because SecureSphere uniquely monitors both HTTP and SQL traffic, it is able to correlate front-end application users with their queries *even when the front end user identity is not present in the SQL traffic*.
SQL Connection User Tracking	Tracks application users and their database queries when SecureSphere is monitoring only the SQL traffic, even in the event of connection pooling.	• Multi-tiered non-HTTP based applications. • HTTP-based applications when the SecureSphere gateway is monitoring only SQL traffic.
Direct User Tracking	Tracks all users connecting directly to the database and their database queries.	

Universal User Tracking Architecture

The figure below shows the various scenarios in which users can access applications and databases, and the SecureSphere UUT components which, in all these scenarios, enable tracking the users through the application to the database transactions.

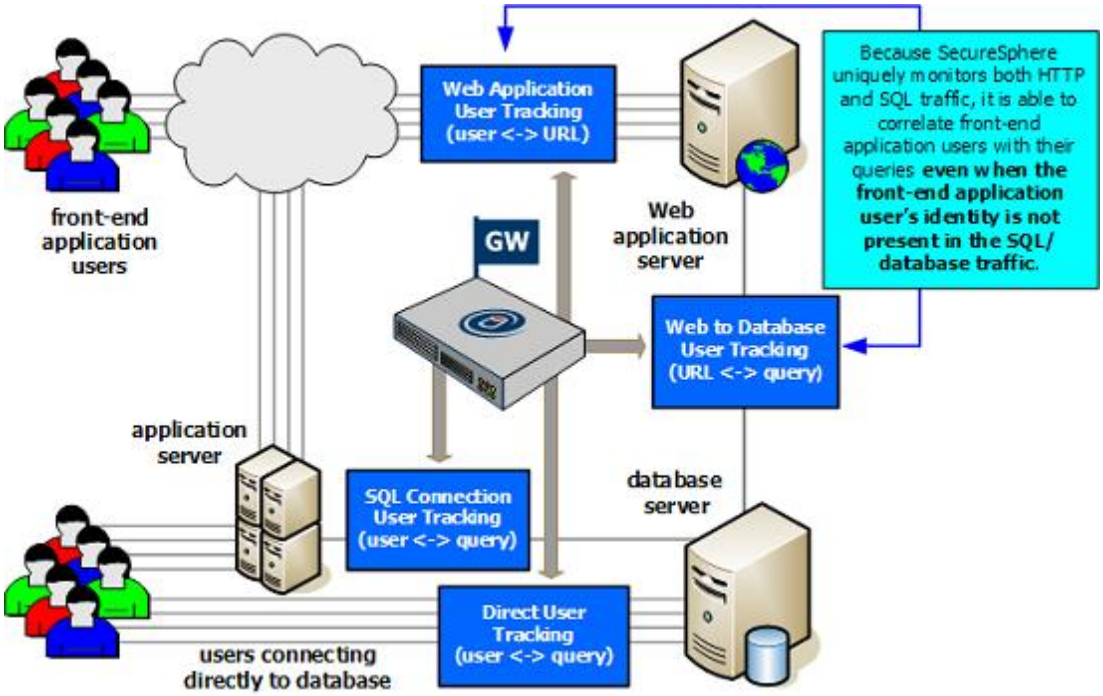


Figure 42: Universal User Tracking

Each of these scenarios is explained in detail in the following sections:

- Web Application User Tracking on page 287
- Web to Database User Tracking on page 295
- SQL Connection User Tracking on page 300
- Direct User Tracking on page 305

DB Profiling and UUT

UUT is an integral part of database profiling. The following topics provide an overview of how UUT works with database profiling:

- Learn Individual Queries on page 286
- Hashed Users in Database Profiling and UUT on page 287

Learn Individual Queries

In order that Universal User Tracking be able to extract user names from queries, the **Learn Individual Queries** parameter must be enabled in the **Learning Preferences Tab.** For more information, see Learn Individual Queries.

Hashed Users in Database Profiling and UUT

Some databases use an encrypted authentication scheme which prevents SecureSphere from extracting the user names properly. In these cases, SecureSphere aggregates all the unrecognized user names under a special user called "Hashed User". All queries and behaviors for these users are assigned to Hashed User in the DB profile.



Note: Another situation in which users are classified as "Hashed User" is when the MSSQL service is not properly configured. The MSSQL service must be configured to log on as administrator or as another domain user, and not as a built in user (such as LocalSystem). Otherwise, remote connections will be audited as hashed users is some scenarios.

However, SecureSphere is able to extract the encrypted user name from encrypted traffic when:

- It is encrypted in SSL, either as part of the authentication process or in the database connection, if SecureSphere has been provided with the SSL certificate and private key used by the server.
- Kerberos authentication is used with Microsoft SQL Server, and SecureSphere has been provided with the password.

You can prevent aggregating users under the Hashed User by making the encryption keys and Kerberos passwords available to the SecureSphere gateway.

Web Application User Tracking

Web Application User Tracking learns the Web application's login URLs as part of the process of building the Web application profile. When a Web application user successfully authenticates to the application, SecureSphere associates the Web application user name with an HTTP session and IP address, and tracks the user throughout the duration of the session.



Note: Web Application User Tracking supports both form-based and certificate-based user authentication.

The user information acquired by Web Application User Tracking is used in audit logs and alerts.



Note: SecureSphere does not authenticate application users, but only tracks successful logins.

The following topics provide information regarding web application user tracking and how to configure it:

Form-Based User Authentication 287
Certificate-Based User Authentication 288
Configuring Web Application User Tracking 288

Form-Based User Authentication

Action URLs are URLs of the forms in which users login to the Web application. SecureSphere learns the action URLs and the rules which determine whether the login was successful, based on what the application does after the attempted login.

In the **Action URL Details** pane, **Login Attempts Decision Rules** specify the behavior of **Action URL**, that is, what SecureSphere has observed happening when the user entered **Field1** and, optionally, **Field2**.

The learned **Login Result** is used as follows:

- If **Login Result** is **Successful**, the username is tracked throughout the HTTP session.
- If **Login Result** is **Failed**, the username is not tracked through the remainder of the HTTP session.

You can define a policy to alert on login failures, or on brute force attacks that are characterized by a large number of login failures in a short period of time.

- If **Login Result** is **Can't Tell**, the username is not tracked through the remainder of the HTTP session.

Learning Action URLs

Although this behavior is learned, it can happen that not enough information is available for SecureSphere to correctly model the behavior. For example, if the Web application returns a response code of 200 for both successful and unsuccessful logins, SecureSphere would be unable to distinguish between the two results. Or if the field names of the user name or other identifying fields were non-standard, then SecureSphere would be unable to know which of the form's fields represents the user name and so on.

If an action URL has more than one form, SecureSphere will check the forms with both **Field1** and **Field2** before checking forms with only **Field1**.

In any case, the administrator can manually modify the rules.

Certificate-Based User Authentication

When a user authenticates to a Web server using a client certificate, SecureSphere extracts the user name from the handshake traffic between the client and the Web server. If the traffic is encrypted, SecureSphere can read it provided that the SSL keys have been previously loaded into the SecureSphere gateway.

Because the user login occurs outside the context of HTTP, SecureSphere is unaware of unsuccessful logins. If the Web server accepts events on the SSL connection following the SSL handshake, SecureSphere assumes that the login was successful. The user is assumed to be the same user for the duration of the SSL connection.

SecureSphere supports the following certificate encoding methods:

- ASN.1/BER

Configuring Web Application User Tracking

You can configure Web Application User Tracking by performing any of the following actions:

- Specifying Session Identifiers and Authentication Methods on page 288
- Viewing, Manually Adding and Editing Action URLs on page 290
- Creating a Web Application User Tracking Decision Rule on page 292

  See also: **Example - Creating a Set of Decision Rules for an Action URL** as shown in Creating a Web Application User Tracking Decision Rule on page 292.
- Deleting a Web Application User Tracking Decision Rule on page 294

Specifying Session Identifiers and Authentication Methods

To specify the session identifiers and authentication method to be used for a Web service:

1. In the **Main** workspace, select **Setup** > **Sites**.
2. Select a Web service.

3. Click the **Operation** tab.

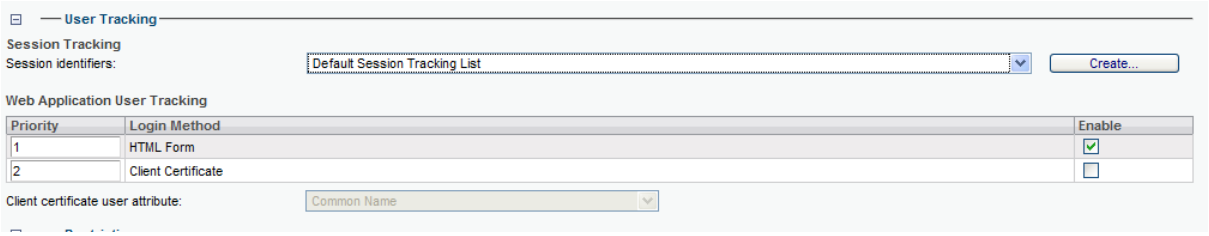


Figure 43: Web Application User Tracking

4. Open the **User Tracking** section.
5. Under **Session identifiers**, accept the **Default Session Tracking** List or create a new Session Identifier Group by clicking **Create**.

   Session Identifier Groups are global objects. If you choose to create a new Session Identifier Group, you must define its members. For more information, see Configuring Session Tracking on page 298.
6. Enable at least one of the listed **Login Methods** and specify its parameters.

Parameter	Description	
Priority	Determines the order in which items are checked.	
Login Method	**HTML Form, Client Certificate**,**Kerberos** and **NTLM** are available.	To support Kerberos operation, either the Kerberos password must be typed in on the **Definitions** tabs of the web service, or support must first be configured as described in Configuring Kerberos Settings To support, you must configure it as describe in Configuring NTLM Support on page 289
Enable	Select to enable the **Login Method**.	
Client certificate user attribute	If the **Client Certificate method is enabled**, select the user attribute to be used as the user's name.	Select **Common Name** (CN) or **e-mail**.

Configuring NTLM Support

SecureSphere supports NTLM authentication, and the identifying of users as part of the NTLM authentication process. When using NTLM in your network and wanting to identify users in audits and logs, you need to enable this feature in as described in Specifying Session Identifiers and Authentication Methods on page 288. Once enabled, SecureSphere can then track failed logins and generate events and alerts based on failed logins.

To configure NTLM support:

1. In the **Main** workspace, select Setup > Sites.
2. In the **Sites Tree**, expand the relevant site and server group, then click on the web service you want to configure. Web service options appear in the main window.
3. In the **Sites** window, click the **Operations** tab. Operations tab options are displayed.
4. Expand the **NTLM Authentication** section on the bottom of the page by clicking the plus (+) sign.
5. Click **New** , a new rule row is added.



Note: When using more than one rule, the order of rules is important. If the result of rules contradict each other (1 is matched, a second is not), SecureSphere goes by the first matched rule.

6. Configure the parameters as follows:
   - **Type**: Select the Type of identifier you want to use to determine the status of a login. Options include Pattern, Redirect, or Response.
   - **Value**: Type a value used together with the identifier. Values may include the following:
   - **Pattern**: Value can be a regular expression or use wildcards. For example, you may want to take the value of a jsessionid like jsession_id=123456
   - **Redirect**: Value would be a URL. For example, http://error/error.html
   - **Response**: Value would be an HTTP response code such as 401 or 402.
   - **Login Result**: Select an option to determine if the type and value combination represent a **Success**, or **Failed Login Attempt**.



Note: The recommended settings for standard use involves two rules:
Type: Response | **Value**: 401 | **Login Result**: Failed Login Attempt
Type: Response | **Value**: 403 | **Login Result**: Failed Login Attempt

7. Under **Default Decision**, select the desired option to determine the default action if no rules are matched. Options include **Successful Login** or a **Failed Login Attempt**.
8. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

EXAMPLE: CONFIGURING NTLM AUTHENTICATION AS FAILED WHEN 401 HTTP CODE RETURNED

In this example, we configure NTLM with a Response Code which when matched, identifies the user as having failed logging in. This would result in an event being created in SecureSphere, and if so configured, this information being audited.

To configure NTLM:

1. In the **Operations** tab of the desired service, click **New** and add a new row.
2. Under **Type**, click **Response Code**.
3. Under **Value**, type **401**.
4. Under **Response**, select **Failed Login Attempt**.
5. Under **Default Decision**, select **Failed Login Attempt**.
6. Click **Save**. When a user accessing the IIS server receives a response code of 401, they are identified by SecureSphere as having failed their login attempt and an event is created. If their response code does not match the one received, they will be identified as having failed the login attempt.

Viewing, Manually Adding and Editing Action URLs

To view a Web application's Action URLs:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, *or*
   Click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. Click Display (in the upper right corner of the screen).
4. From the menu, select **Web Application User Tracking**.
5. In the **Web Application User Tracking** pane, select an Action URL.

6. To manually add an Action URL, click , and in the **Create Action URL** window, enter the parameters, as described in Web Application User Tracking Pane Parameters on page 291.



Note: If you manually add an Action URL, its **Login URL** is never learned and cannot be manually defined.

7. To delete an Action URL, select it and click .



Note: If you delete an Action URL, it will never be relearned.

8. To edit an Action URL, select it and in the **Action URLs Details** pane, edit its parameters, as described in the Web Application User Tracking Pane Parameters on page 291 and table Action URL Details Pane Parameters on page 292, and click **Save**.

Web Application User Tracking Pane Parameters

Web Application User Tracking Pane Parameters

Parameter	Description	
Action URL	A URL from which the user logs in to the application.	This is not the login form. For example, the user accesses `login.asp`, issues a GET request, and the username and password fields are displayed. When the user clicks **Submit**, a POST request to `dologin.asp` is issued with these parameters. In this example, the Action URL is `dologin.asp`, not `login.asp`.
Field1	The first field of interest in the URL form.	This field is typically the primary identifying field, for example, username.
Field2	The second field of interest in the URL form.	This field is typically the some additional identifying information, for example, an account number.

Action URL Details Pane Parameters

Action URL Details Pane Parameters

Parameter	Description	
Action URL Status	**In Learning**: SecureSphere is still learning this Action URL. **Active**: SecureSphere has learned this Action URL and is enforcing the decision rules. **Manual Editing**: The decision rules are not yet being enforced.	
Type	The entries under **Type** are rules, specifying what happens when the user enters data, and are tested sequentially. **Redirect**: The user was directed to another URL, given by **Value**, with the specified **Login Result**. This behavior can be learned. **Response Code**: The user was sent the response code given by **Value**, with the given **Login Result**. This behavior can be learned. **Pattern**: The pattern given by **Value** was detected in the HTML BODY of the response page URL, which indicates the given **Login Result**. This behavior is not learned and must be defined by the administrator. **Note:** HTML HEADER patterns are not matched.	**Redirect** and **Response Code** rules are learned, but can also be edited and created by the administrator. **Pattern** rules are not learned and must be created by the administrator.
Default Decision	In Protect mode, SecureSphere will take this decision if none of the decision rules applies. Select one of: **Succeeded**: SecureSphere associates the user with the session. **Failed**: SecureSphere does not associate the user with the session. **Unclassified Login Attempt**: SecureSphere associates the user with the session.	If none of the decision rules applies, SecureSphere applies **Default Decision**.
Login URLs	The URL from which the user reached **Action URL**. This parameter is only learned.	

Creating a Web Application User Tracking Decision Rule

To define a Web Application User Tracking Decision Rule:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, *or*
   Click the application, for example, Default Web Application.

If you do not see the application, either expand the server group or click **Change Application**.

3. Click Display.
4. From the menu, select **Web Application User Tracking**.
5. In the **Web Application User Tracking** pane, select an **Action URL**.
6. In the **Action URL Details** pane, click .
7. Under **Type**, select one of the following: **Redirect**, **Response Code** or **Pattern**.
   See table Action URL Details Pane Parameters on page 292 for an explanation of these choices.
8. Specify **Value**.
9. Select **Login Result**.
10. Under **Action URL Status**, select **Active**.
11. Click **Save**.

EXAMPLE: CREATING A SET OF DECISION RULES FOR AN ACTION URL

In this example, an action URL is defined from scratch. In practice, SecureSphere is usually able to identify action URLs automatically, though the parameters may require some editing for reasons discussed above, see Viewing, Manually Adding and Editing Action URLs on page 290.

The following decision rules for the action URL will be manually created:

- If the user is redirected to **login/homepage.asp**, the login was successful.
- If the user is redirected back to the action URL, the login was unsuccessful.
- If the response code is **404** or **500**, the result of the login cannot be classified by SecureSphere.
- If the response includes the string **failed**, the result of the login was unsuccessful.
- In all other cases, the result of the login cannot be classified by SecureSphere.

To define the Web Application User Tracking Decision Rules for the Example:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, *or*
   Click the application, for example, Default Web Application.
   If you do not see the application, either expand the server group or click **Change Application**.
3. Click Display.
4. From the menu, select **Web Application User Tracking**.
5. In the **Web Application User Tracking** pane, click .
6. In the **Create New Action URL** window, enter the following:
   - **Action URL**: for example, **/mycompany/quack/login**
   - **Field1**: for example, **username**
   - **Field2**: for example, **usernum**

   Note: The values for Field1 and Field2 are the names of the fields from the page's HTML source which represent the user's identifying information, for example, username and employee_num.
7. In the **Action URL Details** pane, click .
8. Under **Type**, select **Redirect**.
9. Under **Value**, enter **login/homepage.asp**.
10. Under **Login Result,** select **Successful Login**.
    This completes the definition of the first decision rule. The meaning of the rule is that if the user is redirected to **login/homepage.asp**, then this considered a successful login. This rule is placed first, because the rules are searched sequentially, and the most common result is a successful login.

Next, define a rule for an unsuccessful login.

11. In the **Action URL Details** pane, click .
12. Under **Type**, select **Redirect**.
13. Under **Value**, enter **/mycompany/quack/login**, that is, the action URL.
14. Under **Login Result,** select **Unsuccessful Login**.

    This completes the definition of the second decision rule. The meaning of the rule is that if the user is redirected back to the same page, this considered a unsuccessful login. This rule is placed second because unsuccessful logins are expected to be relatively rare.

    Next, define rules for unsuccessful logins based on the response code.

15. In the **Action URL Details** pane, click .
16. Under **Type**, select **Response Code**.
17. Under **Value**, enter **404**.
18. Under **Login Result,** select **Unclassified Login Attempt**.
19. In the **Action URL Details** pane, click .
20. Under **Type**, select **Response Code**.
21. Under **Value**, enter **500**.
22. Under **Login Result,** select **Unclassified Login Attempt**.

    Next, define a rule for unsuccessful logins based on a pattern in the response page.

23. In the **Action URL Details** pane, click .
24. Under **Type**, select **Pattern**.
25. Under **Value**, enter **failed**.
26. Under **Login Result,** select **Unsuccessful Login**.

    Next, define a default rule, that is, a rule that defines the result when none of the other rules applies.

27. Under **Default Decision**, select **Unclassified Login Attempt**.
28. Under **Action URL Status**, select **Active**. The **Action URL Details** window for this example is shown in the figure below.
29. Click **Save**.

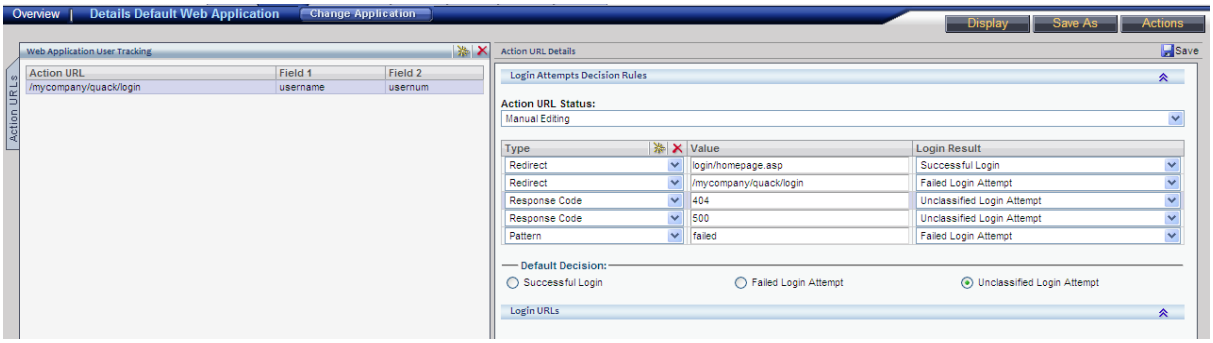


Figure 44: Action URL Example

Deleting a Web Application User Tracking Decision Rule

To delete a Web Application User Tracking Decision Rule:

1. In the **Main** workspace, select Profile.
2. Click **Details** and select the application, *or*

Click the application, for example, Default Web Application.

If you do not see the application, either expand the server group or click **Change Application**.

3. Click Display.
4. From the menu, select **Web Application User Tracking**.
5. In the **Web Application User Tracking** pane, select an **Action URL**.
6. In the **Action URL Details** pane, select a rule.
7. Click .
8. Click **Save**.

Authentication Parameters in POST/XML

When authentication occurs as an HTTP POST/XML transaction, for example, when the username and password are transferred as XML parameters, Web Application User Tracking must be manually configured.

To configure Web Application User Tracking for POST/XML authentication parameters:

1. In the **Main** workspace, select **Setup > Sites**.
2. In the **Sites Tree**, expand the relevant site and server group.
3. Click the Web service you want to configure.
4. In the **Sites** window, click the **Operation** tab.
5. Under **XML Parsing**, deselect **Parse SOAP XML Only**.
6. Under **XML Parsing**, deselect **Ignore Attributes**.
7. Identify the action URL, that is, the URL in the request where the XML is POSTed to the server, and add it as an Action URL to your application's profile.
8. Identify the username parameter (and optionally one other parameter) in the XML the client POSTs to the server, that uniquely identifies the user.
   - If the XML parameter is a tag, for example,
     `<FormName...>...<UserName=xxxxx/>...</FormName>`,
     the parameter name, as it appears in SecureSphere, will be of the form "FormName/UserName".
   - If the XML parameter is an attribute, for example,
     `<FormName...>....<Credentials ...UserName=xxxxx .../>...</FormName>`
     the parameter name, as it appears in SecureSphere will be of the form "FormName/Credentials/@UserName".
9. Add the URL and the user identifying parameter (or parameters) as an Action URL to the profile (see Viewing, Manually Adding and Editing Action URLs on page 290).
10. Configure the Login Decision Rule criteria for successful or failed authentication detection (see Creating a Web Application User Tracking Decision Rule on page 292).
11. Set **Action URL Status** to **Active**.

Web to Database User Tracking

This section reviews subjects related to Web to Database User Tracking. It reviews the following subjects:

Overview 296
Web to Database User Tracking Configuration 297
Topology Examples 299

Overview

Web to Database User Tracking accurately associates URLs with queries, by monitoring both the Web traffic and the resulting database queries. The correlation between URLs and queries is learned over time as part of the Web application profile building process.

Note: Web to Database User Tracking is only available when licenses for both SecureSphere Web Application **and** DAM or DBF are installed. It can only be conducted if traffic is being monitored on a SecureSphere gateway, and does not support tracking through database agents.

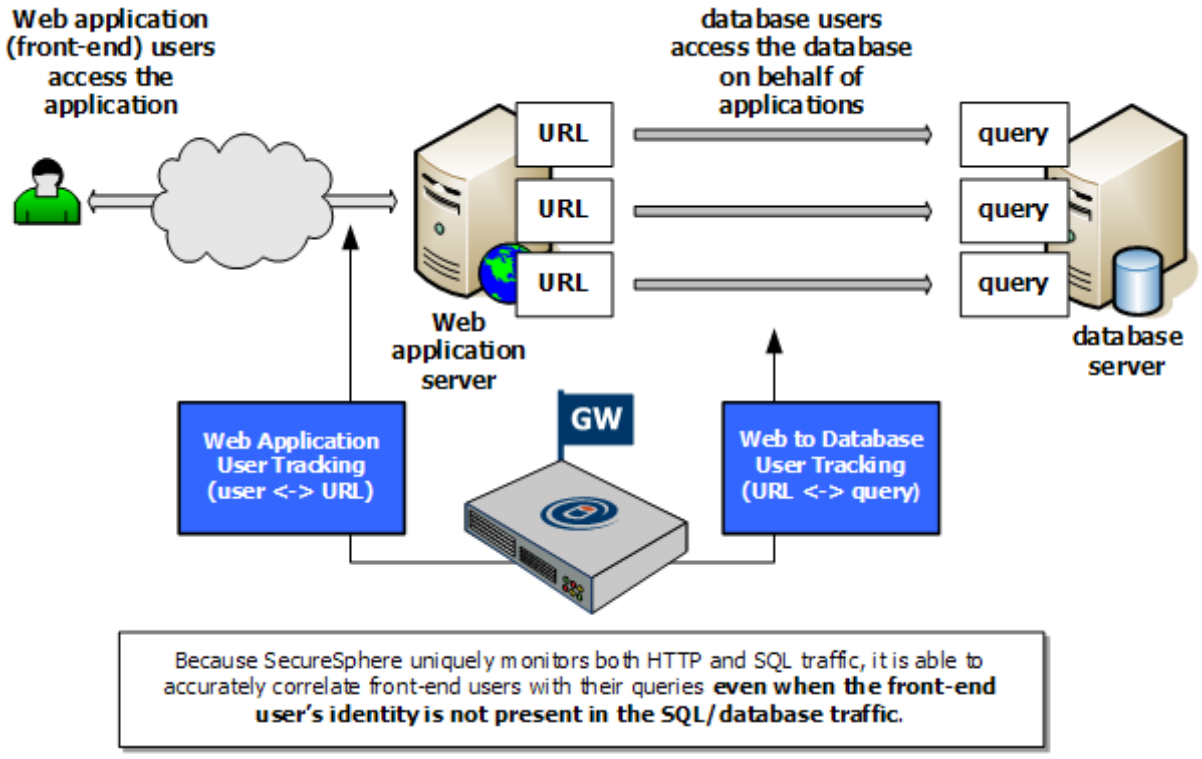


Figure 45: Web to Database User Tracking

The user information acquired by Web to Database User Tracking is used in audit logs and alerts. It is not part of the Web application profile.

The administrator can see:

- **In the Web application profile**: the queries associated with each URL
- **In the DB profile**: the source URL associated with each correlated query
- **In the DB audit**: the web source URL, the session ID and the web application user

In order to correlate HTTP-based application users with their SQL queries, both of the following must be true:

- The same SecureSphere gateway must be monitoring both the web application traffic and the SQL traffic. If the same gateway cannot see both SQL and HTTP, or when the application traffic is non-HTTP, SQL Connection User Tracking can be used to correlate front-end application users with their database activity.
- The SecureSphere gateway cannot be configured in Reverse Proxy mode.

Web to Database User Tracking is learned as part of the profile building process, but you must configure SecureSphere so that it monitors the correct Web server and database server pairs.

Web to Database User Tracking Configuration

The table below lists the tasks that you must perform to configure Web to Database User Tracking.

Web to Database User Tracking Tasks

Step	Action	For more information, see
1	Configure Web Application User Tracking for the Web applications you will be associating with the database service in the Web application profile.	Web Application User Tracking on page 287
2	Associate a Web service with the database service.	Associating a Web Service with a Database Service on page 297
3	Configure session tracking in order to associate multiple sessions with a single user.	Configuring Session Tracking on page 298
4	Define the Web to Database User Tracking Settings.	Defining Web to Database User Tracking Settings on page 298
5	After a sufficient amount of traffic has been monitored, you will be able to see the queries associated with each URL in the Web application profile, in the URL's **Correlated Queries** tab.	Correlated Queries Tab on page 250

Associating a Web Service with a Database Service

To associate a Web application service with a database service:

1. In the **Main** workspace, select **Setup** > **Sites**.
2. In the **Sites** tree, select the server group and click on the database service with which you want to associate the Web service.
3. In the **Operation** tab, under **Application User Tracking**, under **Web to Database User Tracking**, click .
4. In **Service Name**, select a Web service.
5. Check **Enabled**.
6. Click **Save**.
7. Click the **+** sign next to the Web service name to expand it.
8. Under **IP Group**, click .
9. Select an **IP Group** (a Global Object) or **Any**.

   Selecting **Any** is not recommended. It is recommended to track connections from specific IP groups, in order to improve the accuracy of the correlation
10. Click **Save**.
11. Once again, click the **+** sign next to the Web service name to expand it (saving it caused it to be collapsed in the GUI).
12. Click **Any** under **Users** (at the right end of the line).
13. In the **User Defined** window, either select one of the users under **Users**, or define a user under **User Defined**.

You must associate a specific database user with the application activity. Specify the database users who query the database on behalf of applications.

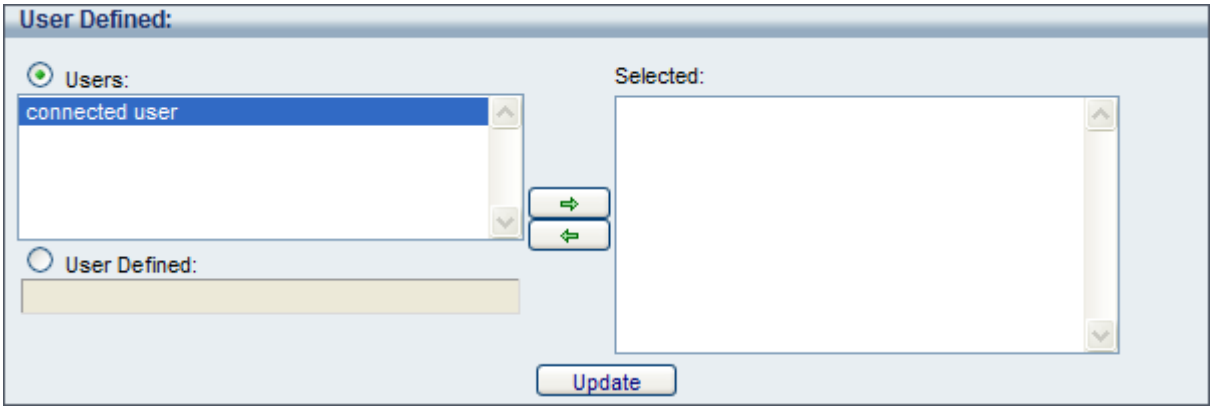


Figure 46: User Defined Window

14. Move the user to the **Selected** box using the arrows.
15. Click **Update**.

Configuring Session Tracking

1. In the **Main** workspace, select **Setup** > **Sites**.
2. Select the Web service associated with the database service.
3. Click the **Operation** tab.

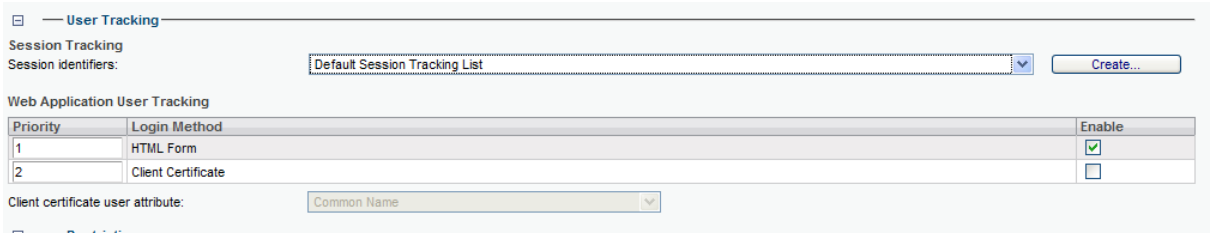


Figure 47: Web Application User Tracking

4. Open the **User Tracking** section.
5. Under **Session identifiers**, accept the **Default Session Tracking List** or create a new Session Identifier Group by clicking **Create**.

   Session Identifier Groups are global objects. If you choose to create a new Session Identifier Group, you must define its members. For more information, see Configuring Session Tracking on page 298.

Defining Web to Database User Tracking Settings

To define Web to Database User Tracking Settings:

1. In the **Admin** workspace, select **System Definitions**.
2. In the **System Definitions** pane, under Gateway Settings, select **Web to Database User Tracking Settings**.
3. Specify the parameters.

   For information about these parameters, see the System Definitions chapter in the **SecureSphere Administration Guide**.

Topology Examples

In this section, a number of topologies are presented which illustrate how to correctly configure SecureSphere to achieve UUT.

In the topology depicted in **One SecureSphere Gateway** below, the SecureSphere gateway sees both Web traffic and traffic from the Web application to the database.

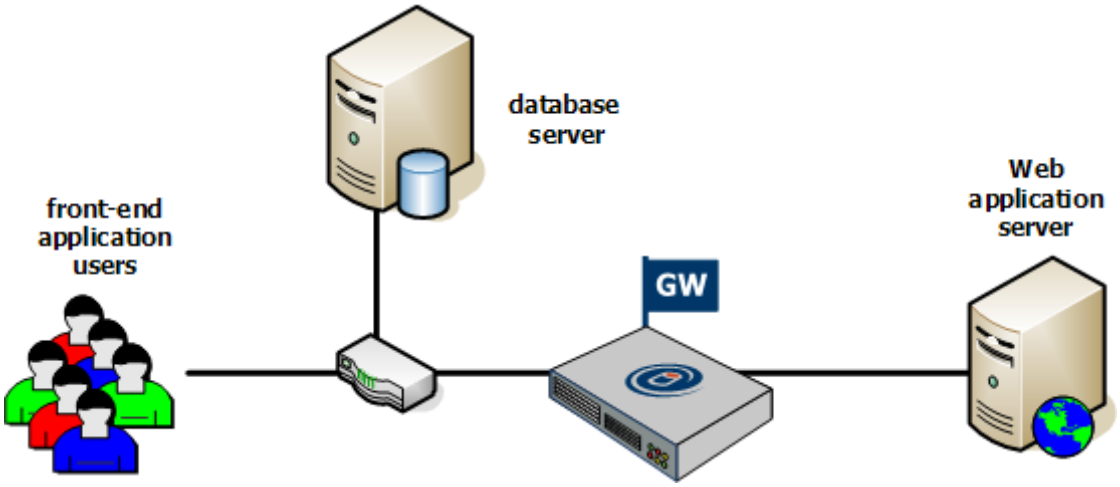


Figure 48: One SecureSphere Gateway

In the topology depicted in **Two SecureSphere Gateways** below, the SecureSphere DAM + WAF gateway sees both Web traffic and database traffic (including traffic from the Web application to the database), while the SecureSphere DAM gateway sees only database traffic. Only the SecureSphere DAM + WAF gateway can achieve Web to Database User Tracking.

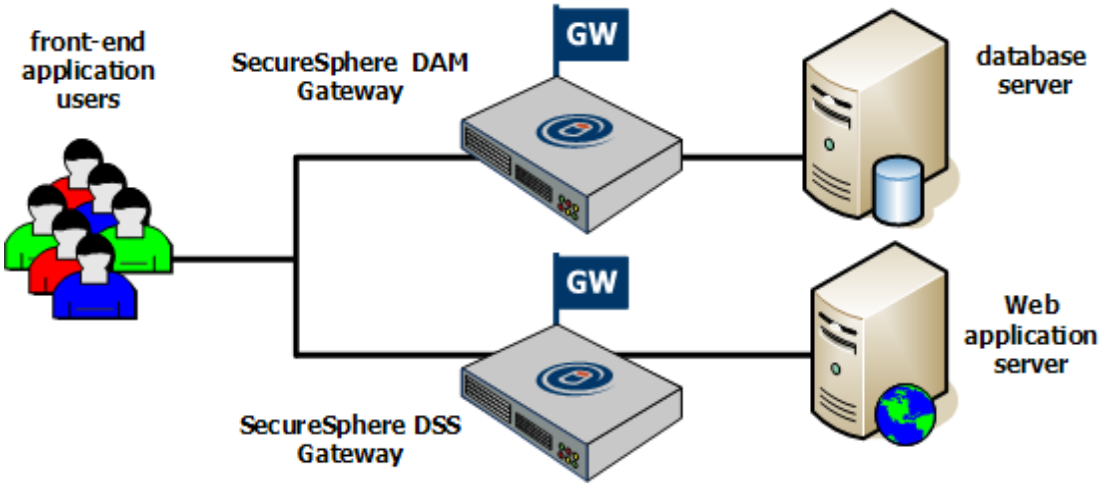


Figure 49: Two SecureSphere Gateways

In the topology depicted in **One SecureSphere Gateway Sees Only Some Traffic** below, the SecureSphere gateway sees Web traffic but does not see traffic from the Web application to the database, so Web to Database User Tracking cannot be achieved.

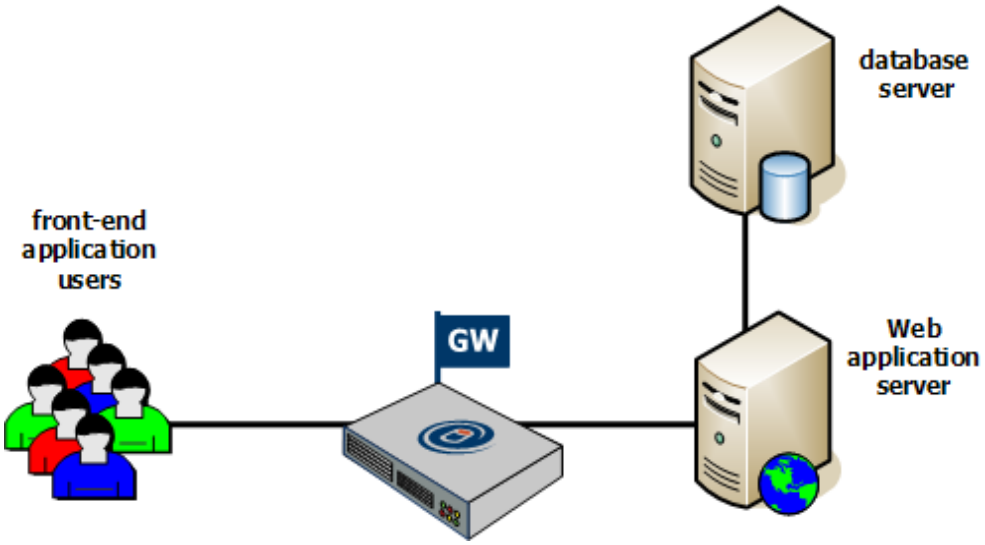


Figure 50: One SecureSphere Gateway Sees Only Some Traffic

SQL Connection User Tracking

SQL Connection User Tracking extracts front-end user names from SQL queries, regardless of the authentication method, to either a Web or an application server. Some business applications use pooled connections and identify the front-end user in specific queries preceding the user queries. Other applications use separate connection-per-user queries.

The following topics provide an overview of SQL Connection Tracking and its configuration:

- SQL Connection User Tracking Architecture on page 301

- Configuring SQL Connection User Tracking on page 302

SQL Connection User Tracking Architecture

SecureSphere supports two methods for tracking SQL Connections, these include:

- Built-in support for known applications such as SAP, EBS and PeopleSoft, as defined by ADC content
- A generic mechanism to support application behavior not predefined by ADC content

The figure below **SQL Connection Tracking** illustrates how SecureSphere manages SQL tracking.

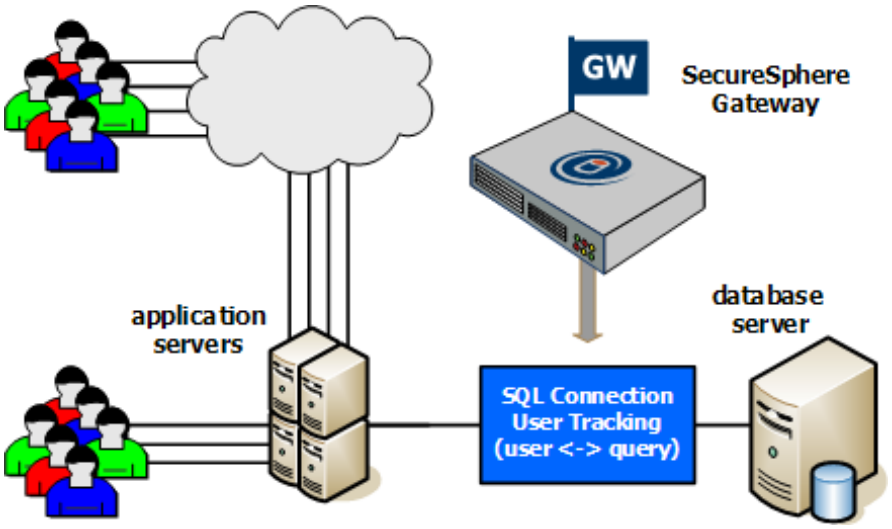


Figure 51: SQL Connection Tracking

SQL Connection User Tracking Overview

The table below lists the tasks that you must perform to configure SQL Connection User Tracking.

SQL Connection User Tracking Tasks

Step	Action	For more information, see
1	Specify an appropriate SQL User Tracking Group, which defines the initialization and termination SQL queries that bracket pooled queries and contain the end-user data that must be extracted in order to associate queries with end-users.	Configuring an SQL User Tracking Group on page 302
2	Associate the SQL User Tracking Group with a database service.	Associating an SQL User Tracking Group with a Database Service on page 305
3	Define an audit policy report that will enable you to track the end-users connections to the database.	Auditing

Configuring SQL Connection User Tracking

Note: SQL Connection Tracking is a learning process. While SecureSphere is still learning, user names appear in audit records within parentheses — for example, **(dbuser1)** — to indicate that the learning process is not yet complete and that there remains some measure of uncertainty about the association of the user with the query.

You can configure SQL Connection User Tracking by performing any of the following actions:

Configuring an SQL User Tracking Group 302
Associating an SQL User Tracking Group with a Database Service 305
Deleting an SQL User Tracking Group - Database Service Association 305

Configuring an SQL User Tracking Group

SQL User Tracking Groups are configured in **Main** > **Setup** > **Global Objects**, under **DB-Related Objects** > **SQL User Tracking Groups**. A number of widely-used database applications are pre-defined under **Built in**, and you can add proprietary applications.

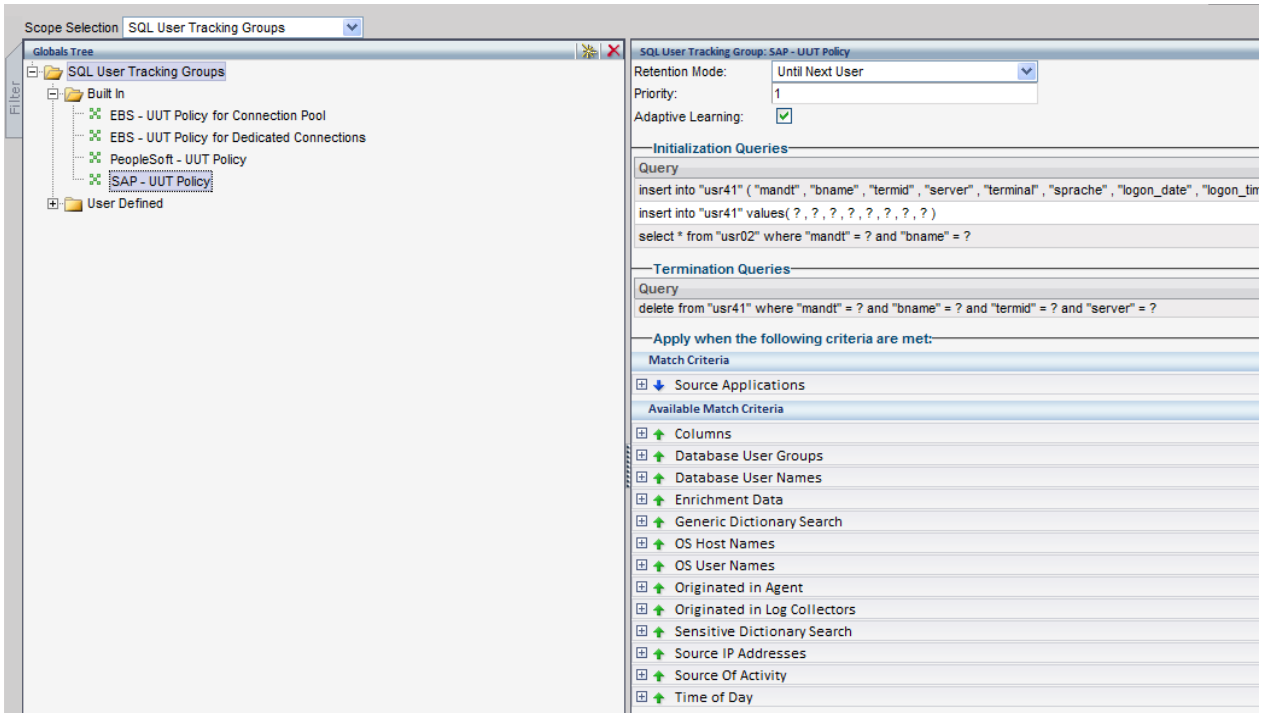


Figure 52: SQL User Tracking Groups Window

To configure an existing SQL User Tracking Group**:**

1. In the **Main** workspace, select **Setup** > **Global Objects**.
2. Under **DB-Related Objects** > **SQL User Tracking Groups**, select one of the UUT Groups.
3. Edit the UUT Group's parameters, as listed in table **SQL User Tracking Group Parameters** below.

To create a new SQL User Tracking Group**:**

1. In the **Main** workspace, select **Setup** > **Global Objects**.
2. Under **DB-Related Objects** > **SQL User Tracking Groups**, right-click on **User Defined**.
3. In the **Create SQL User Tracking Group** window, enter a **Name**.

4. Select either:
   - **From Scratch**, *or*
   - **Use Existing** and select an existing UUT Group.
5. Enter the parameters for the new SQL User Tracking Group, as listed in table **SQL User Tracking Group Parameters** below.

SQL User Tracking Group Parameters

Parameter	Description
Retention Mode	This parameter specifies how to associate user data in the initialization query with queries. Select one of the following: **Query**: User data in the initialization query relates only to the query immediately following. **Stream**: User data in the initialization query relates to all following queries, until the query stream for the user is terminated (see the Termination Queries parameter below). **Stream Until Next User**: User data in the initialization query relates to all following queries, until the next user data in the stream. **Default**: **Stream**
Priority	When more than one UUT Group is associated with a database service, **Priority** specifies the sequence in which the UUT Groups are applied. The highest priority is 1.
Adaptive Learning	If **Retention Mode** is **Stream**, **Adaptive Learning** cannot be enabled. SQL Connection Tracking extracts user IDs from the initialization queries defined in this window. It is possible that as time passes, the application and the initialization queries change, so SQL Connection Tracking is no longer able to extract the user IDs because it does not identify the initialization queries. If **Adaptive Learning** is enabled, SecureSphere attempts to extract user IDs (of users who were previously identified based on the parameters in this window) from other queries. Adaptive learning starts again from scratch when the SecureSphere gateway is restarted. **Note:** This capability is available only for users who have already been extracted from the initialization queries as defined in this window. **Default**: disabled
Initialization Queries	The queries which contain the user data. These are represented in their parsed forms, with **?** replacing parameter values. There can be multiple initialization queries, and they are searched for in the sequence in which they are defined in this window.
Parameter Index	The sequential number of the initialization query parameter which specifies the user name. For example, in the **SQL Tracking Group Parameters** window shown above, Parameter Index is 2, meaning that the user name is given by the value of the parameter bname.

Parameter	Description
Termination Queries	The queries which terminate the stream of queries related to a specific user. There can be multiple termination queries, and they are searched for in the sequence in which they are defined in this window. Termination queries are optional: a query stream can also be terminated by an initialization query. The scope of the user data is also determined by **Retention Mode**.
Apply when the following criteria are met	You can specify that the extraction of user data from queries take place only when the matching criteria are met, so that only relevant queries are scanned.

Associating an SQL User Tracking Group with a Database Service

To associate an SQL User Tracking Group **with a database service:**

1. In the **Main** workspace, select **Setup** > **Sites**.
2. In the **Sites** tree, select the server group and click on the database service with which you want to associate the UUT Group.
3. In the **Operation** tab, under **SQL Connection User Tracking**, click .
4. In **UUT Group**, select an SQL User Tracking Group.



Note: You can associate multiple SQL User Tracking Groups with a database service. The SQL User Tracking Group Groups are applied in the sequence specified by their respective **Priority** fields. As defined in the SQL User Tracking Group Window, see as described in Configuring an SQL User Tracking Group on page 302.

5. Click **Save**.

Deleting an SQL User Tracking Group - Database Service Association

To delete an association between an SQL User Tracking Group **and a database service:**

1. In the **Main** workspace, select **Setup** > **Sites**.
2. In the **Sites** tree, select the server group and click on the database service from which you want to delete an application.
3. Click the **Web Application User Tracking** tab.
4. Under **SQL Connection User Tracking**, select the SQL User Tracking Group.
5. Click .
6. Click **Save**.

Direct User Tracking

Direct User Tracking relies on the native database authentication mechanisms. Direct User Tracking records the database user name, hostname, IP address, client application name and operating system name of the user directly accessing the database.

Notes:

- For Oracle and DB2, also tracks OS user name and OS host name.
- The availability of data other than DB user name is protocol dependent.

Direct User Tracking is part of SecureSphere's dynamic database profile mechanism. For more information, see Understanding DB Profiles in SecureSphere.

The following topics provide information regarding direct user tracking:

Direct User Tracking Architecture 306
Configuring Direct User Tracking 306

Direct User Tracking Architecture

Direct User Tracking is primarily used to track all users directly connected to the database, either over the network locally. SecureSphere associates the DB user name, IP address, OS host name, OS user name and client application name with all subsequent SQL transactions performed by the user.

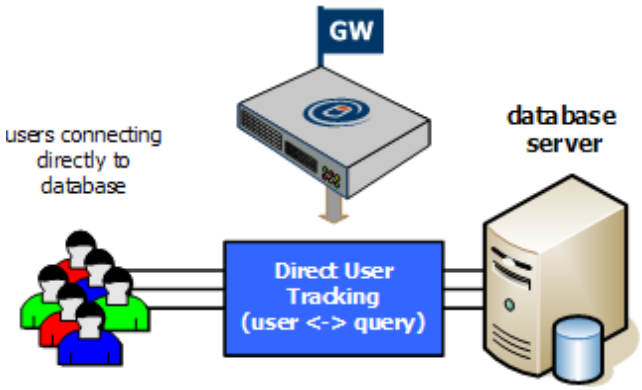


Figure 53: Direct User Tracking

Note: Users connecting to the database over the LAN are monitored by the SecureSphere gateway. To monitor users connecting directly to the database server, you must install a SecureSphere database agent on the database server.

Configuring Direct User Tracking

There is no need to configure Direct User Tracking. Once you configure the database service on the server group, SecureSphere begins to dynamically build the database application profiles, monitoring the SQL traffic and acquiring the user information.

Note: You can disable dynamic profiling. For more information, see Disabling Automatic Web Profiling on page 278.

CHAPTER 11

Signatures

Signatures are text strings which identify known attacks. Imperva's Application Defense Center (ADC) maintains signatures of several thousand attacks, some of them originally provided by Snort® and others provided by ADC. These signatures can be downloaded to the SecureSphere MX automatically or manually, depending on how the MX Server is configured.

This section describes signatures and dictionaries, including:

Working with Signatures 307
Working with Dictionaries 310
Standard Dictionaries 314

Working with Signatures

Signatures are organized in dictionaries, each of which is essentially a filter on the signatures database. You can define new signatures and new dictionaries. You can enable or disable signatures in individual dictionaries or in all dictionaries, and you can apply dictionaries to policies.

The following topics provide information regarding signatures and their management:

- Signatures Workflow on page 308
- Viewing and Filtering Signatures on page 309
- Viewing a Signature's Details on page 309
- Editing a Signature on page 309
- Enabling and Disabling a Signature on page 310

Signatures Workflow

SecureSphere includes thousands of signatures provided by Imperva's ADC. These signatures are regularly updated, and the updated signatures can be automatically downloaded to the SecureSphere. In addition, you can define custom dictionaries and custom signatures and apply the dictionaries to policies.



Note: Defining a custom dictionary or a custom signature has no effect unless you apply the dictionary and its signatures to a policy.

Signatures Not Applied by Default

The following ADC signatures are by default not applied to any policy:

- Anti-Google Hacking
- Application Code Leakage
- Credit Card Number Insertion
- Data Leakage - Korean Resident Registration Numbers
- Developer Comments Leakage
- Directory Browsing Detection
- Full Width - Half Width Unicode Encoding Blocking
- Personal Information Leakage
- Sensitive Error Message Leakage
- Web Worm - Defacement

Signature Tasks

The SecureSphere administrator must perform the following tasks:

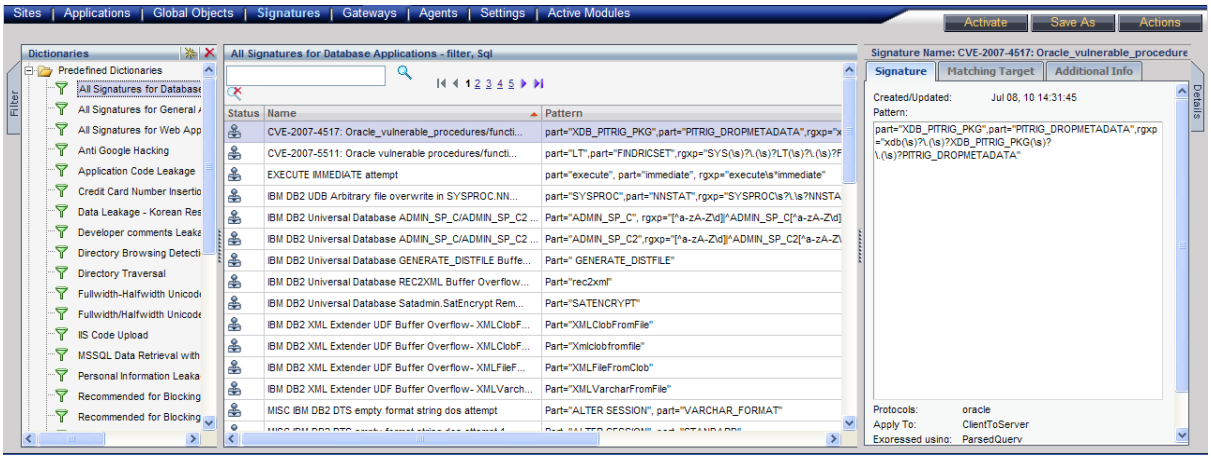
Signature Tasks

	Action	For more information, see
1	Specify whether to download signatures from ADC automatically or manually.	For more information, see **SecureSphere Administration Guide**.
2	Define custom dictionaries and signatures as required, and apply the custom dictionaries to policies.	Working with Dictionaries on page 310 Working with Signatures on page 307

Viewing and Filtering Signatures

To view signatures:

1. In the **Main** workspace, select **Setup** > **Signatures**.



   The dictionaries are displayed in the left pane.

   The signatures included in the dictionary selected in the left pane are displayed in the middle pane.

   The details of the signature selected in the middle pane are displayed in the right pane. See Viewing a Signature's Details on page 309 for more information.

2. To filter the list of signatures in the middle pane, enter text in the box in the upper left corner of the middle pane and click .

Viewing a Signature's Details

To view a signature's details:

1. In the **Main** workspace, select **Setup** > **Signatures**.
2. In the **Dictionaries** pane, select a dictionary.
3. In the middle pane, select a signature.

   The signature's details are displayed in the right pane, in three tabs:

   - **Signature**: Created and Updated dates, pattern, protocols, and so on
   - **Matching Target**: Class, attack complexity, risk, details
   - **Additional Information**: affected systems, references, accuracy.



Note: For predefined signatures, these fields cannot be changed.

Editing a Signature

To edit a signature:

1. In the **Main** workspace, select **Setup** > **Signatures**.

2. In the **Dictionaries** pane, select the dictionary to which the signature you want to edit belongs.



Note: You can edit signatures only in Manual dictionaries.

3. In the middle pane, select the signature you want to edit.
4. In the rightmost pane, edit the signature's parameters.
5. Click **Save**.

Enabling and Disabling a Signature

You can disable a signature from all dictionaries or from a specific dictionary, and you can enable a disabled signature in all dictionaries or in a specific dictionary. A signature's status is indicated by the icon to its left:

Signature Status

Icon	meaning
	enabled
	disabled in this dictionary
	disable d in all dictionaries

To enable a signature:

1. In the **Main** workspace, select **Setup** > **Signatures**.
2. In the **Dictionaries** pane, select a dictionary.
3. In the middle pane, right-click a signature and select **Enable Signature**.

To disable a signature:

1. In the **Main** workspace, select **Setup** > **Signatures**.
2. In the **Dictionaries** pane, select a dictionary.
3. In the middle pane, right-click a signature and select one of:
   - Disable Signature From This Dictionary Only
   - Disable Signature From All Dictionaries

Working with Dictionaries

A dictionary is a filter on signatures. The following topics provide information regarding dictionaries and their management:

- Creating a New Dictionary on page 311
- Adding a Signature to a Dictionary on page 313
- Applying a Dictionary to a Policy on page 314

Creating a New Dictionary

To create a new dictionary:

1. In the **Main** workspace, select **Setup** > **Signatures**.
2. In the **Dictionaries** pane, click .

3. In the **Create New Dictionary** window, enter the parameters. For a detailed description of the options in this window, see the table below **Create New Dictionary Options**.

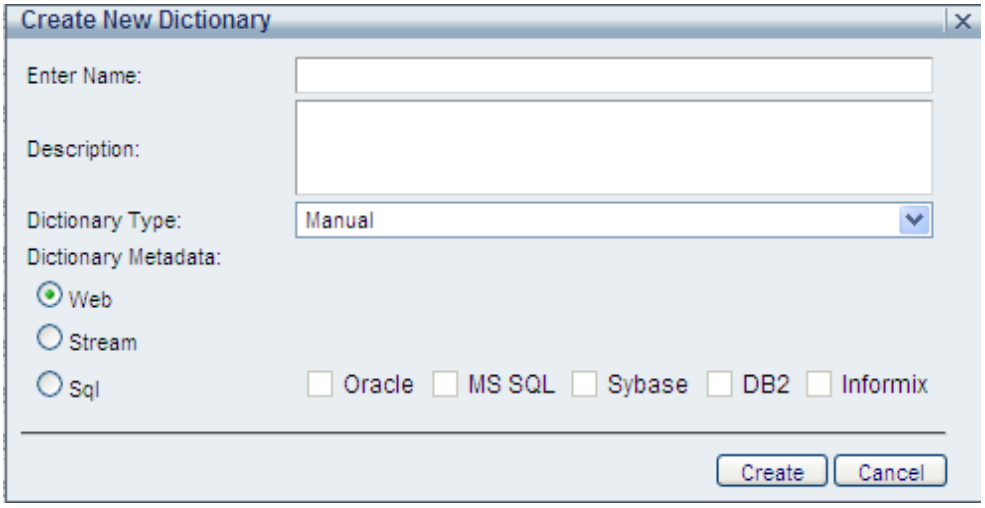


Figure 54: Create New Dictionary Window

4. Click **Create**.

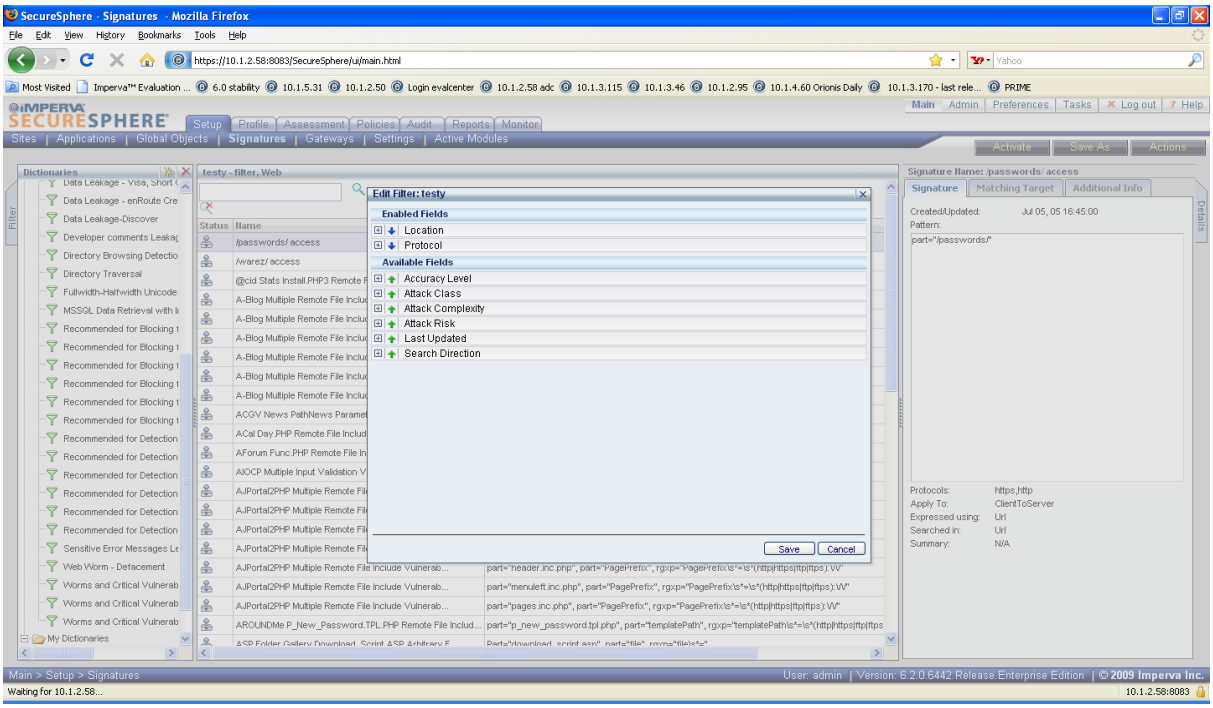


Figure 55: Selecting Dictionaries

Create New Dictionary - Options

Parameter	Description
Name	The dictionary's name.
Description	A description of the dictionary's function.

Parameter	Description
Dictionary Type	Select one of the following: • **Manual**: Define the signatures manually. • **Filter**: Select the signatures from among the existing signatures (see the figure below).

•

Adding a Signature to a Dictionary

To create a signature, you must first select a dictionary to which the signature will be added.

To add a signature to a dictionary:

1. In the **Main** workspace, select **Setup** > **Signatures**.
2. In the **Dictionaries** pane, select the dictionary to which to add the signature.



Note: You can edit signatures only in Manual dictionaries.

3. In the middle pane, click .

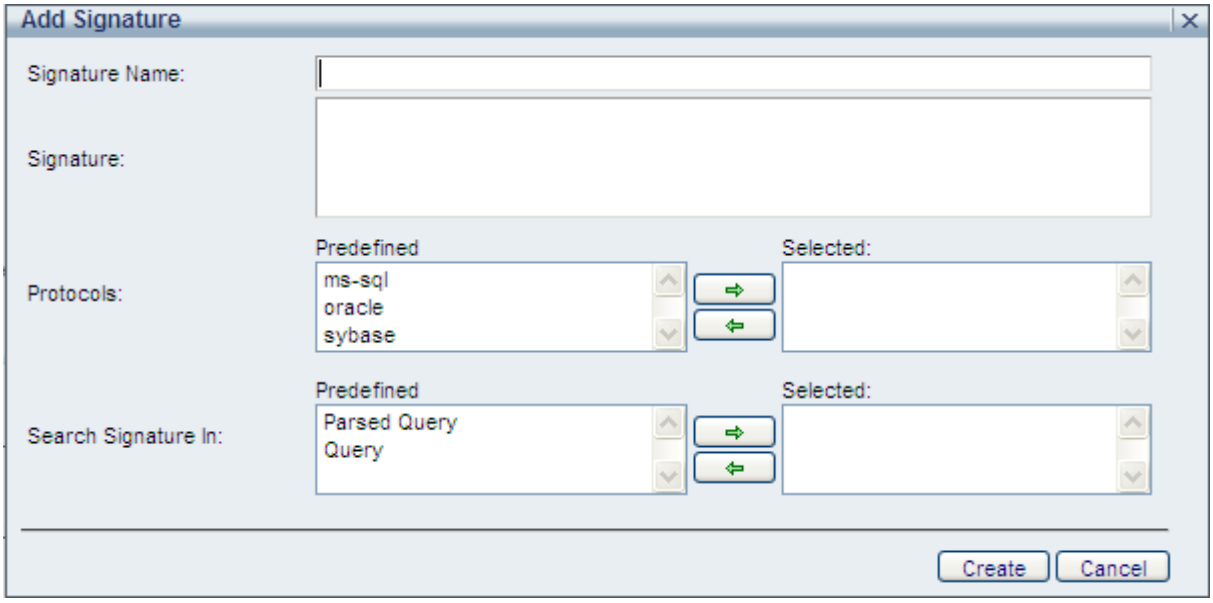


Figure 56: Add Signature Window

4. In the **Add Signature** window, enter the parameters.

Parameter	Description
Signature Name	The Signature's name.
Signature	The syntax for signatures is described in Writing Signatures and Regular Expressions on page 473.
Protocols	Which protocols to search. The list of available protocols depends on the **Dictionary Type** in the **Create New Dictionary** window. • **Stream**: TCP protocols • **SQL**: The SQL protocols selected when the dictionary was created.
Search Signature In	Where to search for the signature string. The list of options depends on the **Dictionary Type** in the **Create New Dictionary** window. • **Stream**: For stream only, select protocols from the list. • **SQL**: Parsed Query or Query
Apply To	Select **Client to Server**, **Server to Client** or **Both Directions**.

Applying a Dictionary to a Policy

You cannot apply individual signatures to a policy, only entire dictionaries, so you should group your custom signatures into custom dictionaries accordingly.

For information on how to apply dictionaries to policies, see Introducing Security Policies on page 166.

Standard Dictionaries

The following table lists and describes the predefined dictionaries. For a description of the attacks against which the individual signatures defend, see the description in the signature's **Matching Target** tab.

Predefined Dictionaries

Dictionary Name	Description
All Signatures for Database Applications	
All Signatures for General Applications	
All Signatures for Web Applications	
Anti Google Hacking	Hackers can exploit search engines such as Google to find sensitive data on web servers.
Application Code Leakage	The code of the application give an attacker great insight into the inner workings of an application. These signatures identify responses that contain source code of any application.
Credit Card Number Insertion	Insertion of a credit card number into parameter value, possibly violating the privacy policy of the web application.
Data Leakage - American Express	Hackers can exploit a large variety of application vulnerabilities to

Dictionary Name	Description
Data Leakage - American Social Security Number	extract credit card numbers from databases
Data Leakage - Credit Card Numbers	
Data Leakage - Diner's Club / Carte Blanche Credit Card Numbers	
Data Leakage - JCB Credit Card Numbers	
Data Leakage - Korean Resident Registration Numbers	
Data Leakage - MasterCard	
Data Leakage - Visa, Long Credit Card Numbers	
Data Leakage - Visa, Short Credit Card Numbers	
Data Leakage - enRoute Credit Card Numbers	
Data Leakage - Discover	
Developer Comments Leakage	Developer comments give an attacker great insight into the inner workings of an application. This attack contains signatures which catch responses containing developer comments.
Directory Browsing Detection	The Directory Browsing attack is used to determine possible vulnerable directories (such as cgi-bin, _vti_bin) and find directory listings that may contain sensitive secret information or list of files to be used as hotlinking, which consumes huge bandwidth. It is also used to determine targeted web site directory structures in order to gain an understanding of the web application infrastructure.
Directory Traversal	Remote "Directory Traversal" and "Remote Execution" vulnerability exist in some firewall appliances.
Fullwidth-Halfwidth Unicode Decoding Blocking	Requests that contain characters decoded on Fullwidth/Halfwidth Unicode Decoding.
MSSQL Data Retrieval with Implicit Conversion Errors	An attacker can retrieve information from an MSSQL database through data type conversion error messages. When an MSSQL database performs an illegal conversion between two data types an error message is returned. The error message contains the actual data that is being converted.
Recommended for Blocking for Database Applications	An attacker may exploit these issues by supplying a large string value to the application. This payload may contain replacement memory addresses and shell code designed to redirect process execution. If successful, these issues may result in a denial of service condition or allow for arbitrary code execution. As a result the attacker may gain unauthorized access to a vulnerable computer.
Recommended for Blocking for Database Applications - Legacy	
Recommended for Blocking for General Applications	
Recommended for Blocking for General Applications - Legacy	

Dictionary Name	Description
Recommended for Blocking for Web Applications	This rule looks for requests of a page called "passwords".
Recommended for Blocking for Web Applications - Legacy	
Recommended for Detection for Database Applications	Using the EXECUTE IMMEDIATE option may allow the attacker to use some text manipulations in order to sneak his code into the database server.
Recommended for Detection for Database Applications - Legacy	
Recommended for Detection for General Applications	This attack is generated when an attacker attempts to activate the :$data variable. Activating this dll might cause malicious activities to the server.
Recommended for Detection for General Applications - Legacy	
Recommended for Detection for Web Applications	This attack is generated when an attacker attempts to access the ftp initialization executable. This enables him to alter the ftp variables at a later time, and to change its normal behavior, leading to possible data theft and to internal component access.
Recommended for Detection for Web Applications - Legacy	
Sensitive Error Messages Leakage	Error messages can give an attacker great insight into the inner workings of an application. This attack contains signatures that catch error responses from a web server.
Web Worm - Defacement	SQL Injection attacks and new malware.
Worms and Critical Vulnerabilities for Database Applications	
Worms and Critical Vulnerabilities for General Applications	
Worms and Critical Vulnerabilities for Web Applications	These issues may be exploited to steal cookie based authentication credentials, map the application root directory of the affected application, execute arbitrary commands and include arbitrary files. Other attacks are also possible. This dictionary contains a large number of signatures identifying a wide variety of specific attacks.

CHAPTER 12

Advanced SecureSphere Configuration

Advanced configuration of SecureSphere enables you to configure a number of advanced features to assist in protecting data and streamlining SecureSphere operation. Configuring advanced SecureSphere features may include any number of elements and steps.

This section reviews the various advanced configuration options available in SecureSphere and include the following:

Advanced Configuration Overview 318
Applying Policies 318
Managing Archive and Key Settings 324
Working with Action Sets and Followed Actions 326
Working with Global Objects 342
Configuring Sensitive Data Protection 350
Working with Data Types 357
Configuring Windows Domains 358
Generic Dictionary Groups 361
Protecting and Auditing Custom Information with Lookup Data 363
Enriching Data 372
Introducing Active Modules 383
Change Tracking - External Objects 383
Application Structure Modeling 387
Working with Tasks 387
Configuring Preferences 392
Configuring Integration with Third Party Management Systems 394

System Event Policies 411
Configuring Scheduling 413
Customizing How SecureSphere Handles Web Traffic 415
Malware Detection Support 439

Advanced Configuration Overview

In addition to Basic Configuration, and the configuration of main SecureSphere components such as Profiles and Policies, you can customize the operation of SecureSphere to match requirements based on your network components. For example, if your organization deals with credit card information or other sensitive data such as social security numbers, you can configure SecureSphere to automatically identify sensitive data with pre-defined dictionaries, then further configure error pages to automatically mask this sensitive data to prevent the data from being exposed to unauthorized personnel.

Identifying Advanced Configuration Requirements

SecureSphere offers a wide range of advanced functionality. To assist in determining the best method to protecting your data, try to identify the components in your system and examine which SecureSphere features may be required. Major functionality that SecureSphere advanced configuration covers includes:

Tasks: Enables you to create and manage tasks both manually and automatically to provide an additional layer of supervision to the SecureSphere workflow. For more information, see Working with Tasks on page 387.

Configuring User Preferences: Customizes the SecureSphere GUI and configures it with personal details that may appear in alerts. For more information, see Configuring Preferences on page 392.

Applying Policies: Enables applying policies to the objects in the site tree (server groups, services, applications) or in the applications tree. For more information see Applying Policies on page 318.

Protecting Sensitive Data: Configures SecureSphere to protect sensitive data including credit and social security numbers. For more information see Using Sensitive Data Dictionary Groups on page 350.

Supporting Load Balancers: Depending on implementation, you may require SecureSphere to support load balancing such as rewriting URLs. For more information, see Working with Tasks on page 387.

Web Traffic Support: Configures advanced support for features that may be used in URLs such as sensitive URLs or Microsoft Encoding. For more information, see Configuring Web Service Restrictions on page 416

Webworm Protection: Configures protection against zero-day worms that utilize unknown signatures. For more information, see Configuring Advanced Web Service Settings on page 424.

Session Tracking: Customizes SecureSphere to identify and work with custom token names as part of session identifiers. For more information, see Configuring Session Tracking on page 298

Applying Policies

Policies are activated by applying them to the objects in Sites/Applications Tree. Once a policy is applied to an object and the configuration is saved and activated, the object is immediately protected by this policy.

Policies of any domain level (server group/ service/application) can be applied to the Sites Tree, which is the default SecureSphere mode. In addition to the Sites Tree mode, you can switch to the Applications Tree mode in which you can work with the logical structure of the applications in your organization.

The differences between correlation rules that participate in the policies on the service level and the rules that participate in the policies on the application level are as follows:

- The policies that are assigned to the Application level are enforced much later than the Service level policies, as the process of identifying the application is more time consuming than identifying the port.
- The enforcement of the application policies is more accurate and therefore more effective, as the policy is assigned to the specific application.

Therefore, any policy can be applied to the Site Tree. In big organizations, in order to achieve more accurate and effective results, it is recommended to apply application level policies to the Application Tree in addition to the policies applied to the Site Tree.

Applying Security Policies to the Objects in the Sites Tree

A Sites Tree contains a graphical representation of the domain object hierarchy for each site defined on SecureSphere. Assigning security policies to all the domain levels in the Sites Tree is very important, as each level is related to a different aspect of protection, see the figure below.

When the new traffic arrives, SecureSphere inspects the traffic as follows:

1. First, the traffic is inspected on the Server Group level, meaning that the traffic is secured using policies assigned to server group already when the first packet arrives, for example, the Firewall policy.
2. The next step in attack detection is reviewing the protocol using policies applied to services. These policies inspect the port of the incoming traffic. For example, the HTTP Protocol Validation policy.
3. On the Application level SecureSphere reviews the request itself and applies the policies assigned to this level, for example, identifying the parameter type and length.

Depending on a policy type, a policy can be associated with multiple server groups/services/applications. In some cases, multiple policies are associated with a single domain object. For example, the administrator can define a server group with applied policies and assign different user group roles, which allows only managing the service or the application under this server group. This enables the administrator to force a basic security on new services and applications created under this server group, if and when created.

In cases where there are multiple children, and you apply a policy only to some of them, higher-level object(s) appear as partially applied (greyed-out node), and the relevant objects as selected



Note: With Assessment Policies, when the parent objected is selected, the assessment policy is applied to all children even though all child objects remain unmarked.

If after applying a policy to a server group, you create a child service, the child service does not inherit the policy.

To apply a policy in the site tree:

1. In the relevant policy, click the **Apply To** tab. **Apply To** options are displayed.

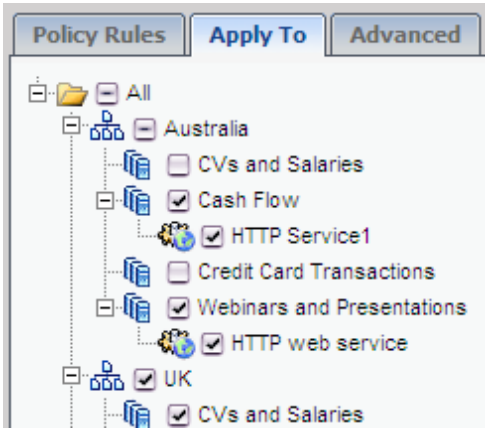


Figure 57: The Apply To pane Showing the Site Tree

2. Select each individual object to which you want to apply the policy.

Applying Security Policies to the Objects in the Applications Tree

In addition to the Site tree option, you can apply the application level policies to the Applications tree in which each level is represented by an application group.

An **application group** is a logical entity that includes one or more applications that share the same preferences and permissions. Using applications tree you can apply a policy to multiple applications in one click. For more information on working with application groups, see Working with Application Groups on page 322.

You can apply application level policies to the application groups and work with application groups as either physical representation of sites, or as logical representations of the protected domain, which enables them to be mapped in both, the physical representation and according to the company hierarchy with its business units.

Application groups can include applications that run on different servers, in different physical locations and provide different services. For example, in a big organization with many branches all over the world you can define a web-based mail system interface in all the branches as an application group, so that you can set all the preferences only once for the whole group and it is also very easy to modify your definitions in all the sites at one click.

EXAMPLE: USING AN APPLICATION GROUP

An example of a Sites Tree is presented in the following figure.

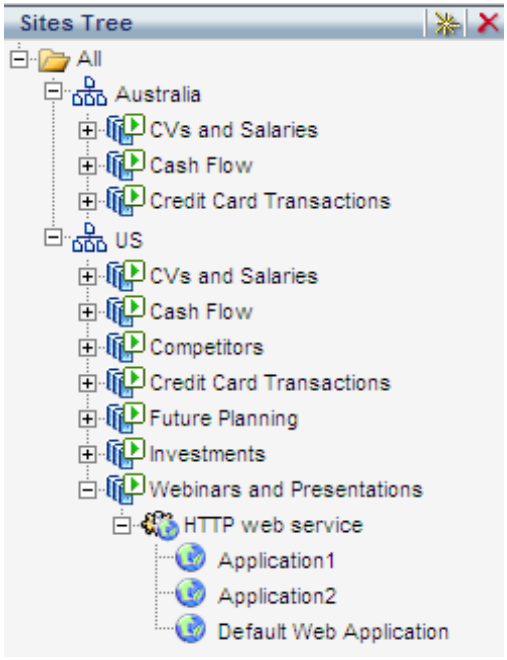


Figure 58: Organization1 Site Tree

In this example we can see a site tree of a company called Organization1. This company has headquarters in US and offices in UK and Australia. The server group Webinars and Presentations with the same applications, Application1 and Application2 is used in each one of the company offices, as shown in the following figure.

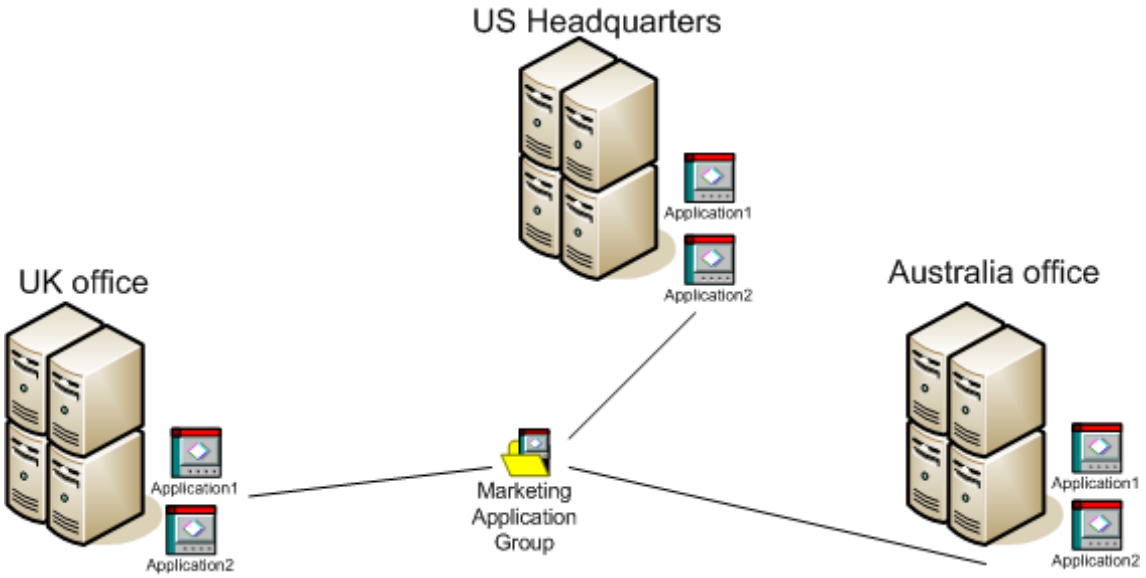


Figure 59: Organization1 Offices

The following figure shows the application tree of Organization1, in which the application groups are defined according to the company departments.

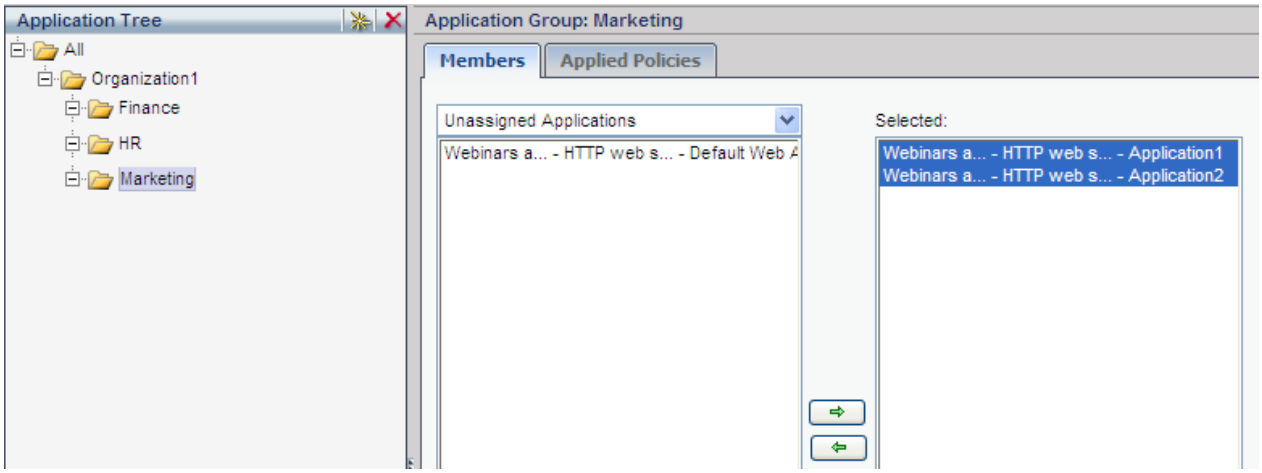


Figure 60: Marketing Application Group

The applications that are selected to be the **Members** of the **Marketing** application group are **Application1** and **Application2**, so that these applications are managed and protected as the same entity. This means that they share the same preferences and application level policies can be applied to the entire group at once.

Working with Application Groups

An application group is a grouping of applications to which you can apply audit policies. An application group can be comprised of applications from any part of the system and from additional application groups which are composed of applications and application groups. The applications in a group do not have to be of the same type or from the same server group.

Application groups can be hierarchically organized. You can create application groups that are part of another application group and organize them how you wish.



Note: If you apply a web related policy to an application group with SQL applications, the policy is not saved and no error message appears. Make sure that only DB related policies are applied to DB applications.

Setting Up Application Groups

Before applying application level policies to the application groups, first you need to define application groups in the **Applications** window and build the hierarchy between them.

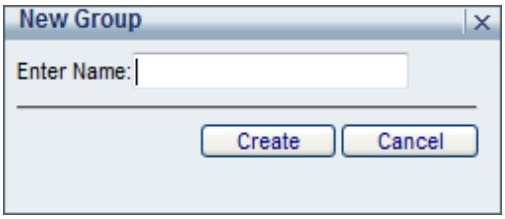
The tree structure of application groups is flexible:

- The tree structure can be any depth you require.
- Applications can appear at any level of the tree.
- Many applications can be grouped in a single application group.

To create an application group:

1. In the **Main** workspace, select **Setup** > **Applications**. The **Applications Tree** appears.
2. Select the root application group under which you want to create the new application group.

3. Click the  button. The **New Group** window opens.



4. Enter the name of the new application group and click **Create**. The application group is created under the selected root.



Notes:

- An application can only appear in one application group, otherwise there would be a conflict between policies applied to different application groups containing the same application.
- You can also create an application group using the Open API. For further information, see the Imperva SecureSphere API document.

Enabling Application Group Settings

To be able to apply application level policies to the application groups, you have to enable this ability.

To enable applying application level policies to the application groups:

1. In the **Admin** workspace, select **System Definitions** > **Application Groups Settings** and then select **Apply application-level policy objects using application group hierarchy**.
2. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Applying Policies to the Application Tree

You can apply policies at a high level in the application tree, even though the policy may not be applicable at that level. SecureSphere applies the policy only to the relevant objects in the tree. You can then tailor the policy to suit individual child objects. When you do this, only the modified policy for the child object is changed and the policy remains unchanged for the remainder of the tree.

Locking Application Groups

Locking a policy on an application group ensures the following:

- The policy is applied to all the application group members. To unapply the policy from one of the applications in the group, you must first unlock the group.
- When a new application is added to the locked group, the policy applied to the applications in the group is also applied to the new application.

The following figure presents a locked application group (Organization 1).

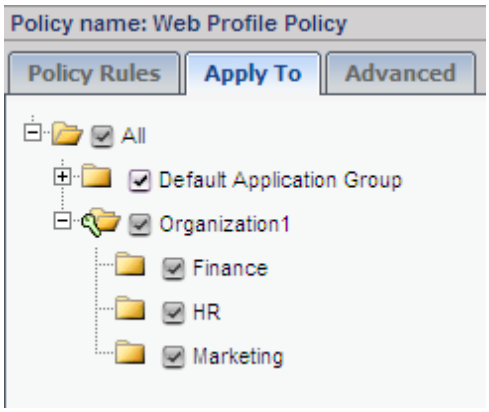


Figure 61: Locked Application Group

To lock an application group:

1. In the **Main** workspace, select **Policies** > **Security** > **Apply To**.
2. Right-click the group that you want to lock and select **Lock Group**. The lock icon appears next to the locked group.

Managing Archive and Key Settings

Archive and key settings in SecureSphere assist you in configuring settings and managing keys used for the archiving of system events, and reports. SecureSphere comes with default settings. However, you can modify these defaults to match your working environment. This section describes how to modify these defaults. Configuring archive and keys settings involve the main tasks listed below.

Managing Archive and Key Settings - Main Tasks

	Action	Description	For more information, see...
1	Importing keys.	Enables you to import and export keys in SecureSphere to used for encryption and signing.	Managing Keys on page 324.
2	Define archive settings.	Enables you to configure archive including keys and filename formats.	Defining Archive Settings on page 325.
3	Configure and apply archive settings.	Configure the archive setting for the required SecureSphere item.	Configuring and Applying Archive Settings on page 325.

Managing Keys

SecureSphere Key management is used to manage keys required for encryption and data signing purposes. SecureSphere comes by default with two keys, one for encryption and the second for signature. It additionally allows you to configure a third set of keys, and import keys for use. Key options include the following:

- **Encryption**: Used to encrypt archives.
- **Signature**: Used to ensure data integrity.
- **Keys**: Located below the encryption and signature options. Enables you to add other keys for archiving actions.

To upload a key:

1. In the **Main** workspace, select **Setup > Settings**. **Settings** options appear.
2. In the **Settings** tree, select **Key Management.** The **Key Management** window appears.
3. In the **Default Keys** pane of the **Key Management** window, click **Upload**. The upload dialog box appears.
4. Under **Private Key**, click **Browse**, navigate to the location of the private key, select it, then click **Open**.
5. Under Public Key, click **Browse**, navigate to the location of the public key, select it, then click **Open**.
6. Enter the **Password** required to access the key. If you do not have a password consult with your system administrator.
7. Click **Upload**. The keys are uploaded into SecureSphere and ready for use.



Note: To export an existing key, click Export. SecureSphere will export the key as a text file.

Defining Archive Settings

SecureSphere comes with default archive settings. It additionally enables you to configure customized archive settings which can utilize your own keys for encryption and signature, and customize archive file name format.

To configure archive settings:

1. In the **Main** workspace, select **Setup > Settings**. **Settings** options appear.
2. In the **Settings** tree, select **Archive Settings.** The **Archive Settings** window appears.
3. In the table header row of the **Archive Settings** table, click  **New**. A new row is added to the table.
4. Type a **Name** for the archive settings.
5. Select an **Encryption Key**.
6. Select a **Signature Key**.
7. Configure an **Archive Name File Format** for the archive. This determines the naming convention for the archive file created by SecureSphere. Default message includes a number of placeholders and reads as follows: `SecureSphere_ Audit_${policyName}_${startDate}_${endDate}.mprv.`
8. Configure an **AM Archive Name File Format** for the AM archive. This determines the naming convention for Active Module (ODM) archive files created by SecureSphere. Default message includes a number of placeholders and reads as follows: `${am}_from${startDate}_to${endDate}_policy=${policyName}`



Note: Only placeholders listed above can be used for these names. The format can however be modified and placeholders can be reorganized or removed.

9. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Configuring and Applying Archive Settings

This procedure describes how to configure the archiving tab for various features and subsequently activate archiving and apply settings after they have been configured. For an overview on how to configure archive settings see Managing Archive and Key Settings on page 324.

To configure the archiving tab:

1. Access archiving from the relevant location, as follows:
   - **Auditing**: Under **Main > Policies > Audit.** Then select the audit policy you want to configure.

- **Track Value Changes**: Under **Main > Setup > Active Module**. Then select the Track Value Changes policy you want to configure.
- **Change Tracking**: **Under Main > Setup > Active Module**. Then select the Change Tracking policy you want to configure.



Note: For information on configuring archiving for System Events and Reports, see the SecureSphere Admin Guide.

2. Select the **Archiving** tab. Archiving options are displayed.
3. Configure an **Archiving Action**. Select an Action Set to be used when the data produced by the policy is successfully archived. For information about action sets, see Configuring Action Sets and Followed Actions on page 75.
4. Configure a **Schedule**: Configure scheduling to determine when the policy should be run. For detailed instructions on how to configure scheduling, see Configuring Scheduling **on page 413.**



Note: To run the archiving immediately, click **Actions** > **Run Now**.

5. If you are configuring Audit Archiving, configure the following, otherwise go to step 6.
   - **Archiving**: Specify whether to archive report data.
   - **Include audit response data in archiving process**: Enable this option to archive response data as well.
   - **Purge Records Older Than:** Specify which archive records to purge.



Note: Archive data will not be purged from gateways to which the current policy is not applied, even if the policy was applied to that gateway in the past.

6. Select an **Archive Settings** from the menu. SecureSphere enables you to choose a default archive setting, or configure custom settings. For information on archive settings, see Managing Archive and Key Settings on page 324.



Note: The **Default Archive Settings** does not by default encrypt the archive file. To change these settings, in the Main workspace, select **Setup** > **Settings** and then select **Archive Settings** from the **Settings** pane, select **Default Archive Settings** and enter encryption keys.

7. Click **Save**. The policy is saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.



Note: Additional archive functions include Archive Now, Purge Now, and Import are available from the Actions button in the top right-hand corner of the SecureSphere window.

Working with Action Sets and Followed Actions

This section reviews the various aspects related to working with Action Sets and Followed Actions. It reviews the following subjects:

Action Sets and Followed Actions Overview 327
Action Set Main Tasks 328
Working with Default Action Sets 329
Creating Custom Action Sets 330
Action Interface Types 330

Configuring Action Interface Parameters 332
Assigning Followed Actions 341

Action Sets and Followed Actions Overview

Action sets and followed actions are components used to define the actions taken by SecureSphere when specific conditions are met. Action sets are configured, then attached as followed actions to various SecureSphere items such as security policies, system events, reports, tasks and so on. The use of action sets provides a flexible means of customizing how SecureSphere responds to events taking place in your system, and plays a key role in enabling you to schedule tasks such as automatically generating reports, and much more.

To view the Action Sets window:

- From the **Main** workspace, select **Policies > Action Sets**. The **Action Sets** window appears.

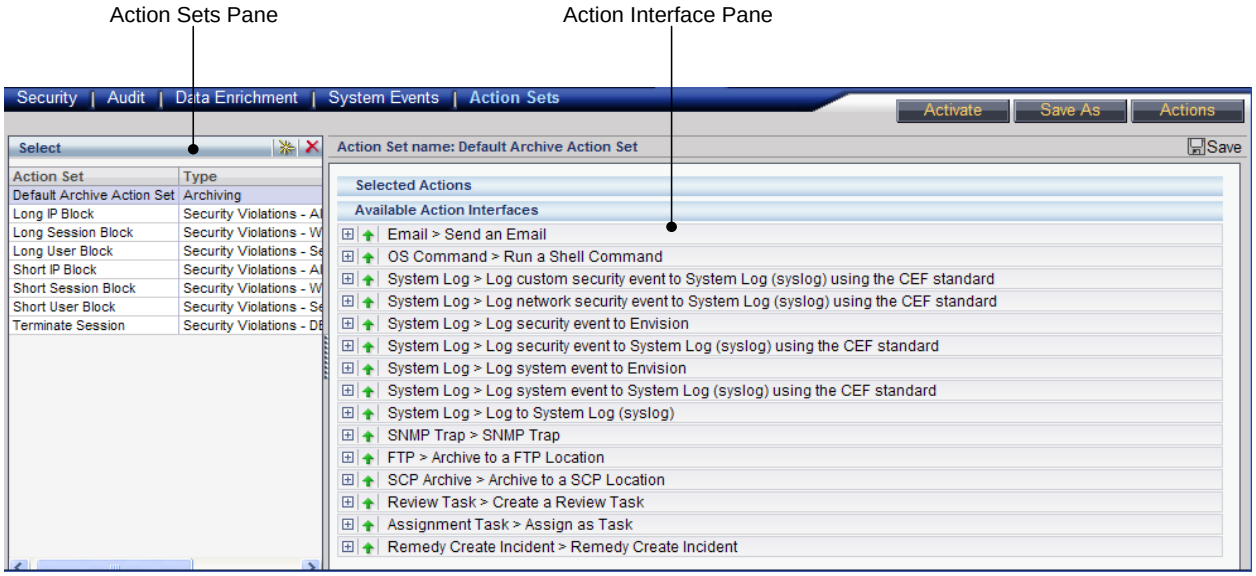


Figure 62: Action Sets Window

The **Action Sets** window has two primary sections:

- **Action Sets**: Lists action sets that are already configured in SecureSphere. For information on how to create custom action sets, see Creating Custom Action Sets on page 330.
- **Action Interfaces**: Lists action interfaces that are available for the action set selected in the Action Sets pane. For information on configuring action interface parameters, see Configuring Action Interface Parameters on page 332.

The following table lists definitions of the different components of action sets and followed actions.

Action Set Definitions

Parameter	Description
Action	An immediate response to an event in your network. For example, you can configure a policy with a rule that determines if SecureSphere detects an attempt to hijack a session, the source connection is immediately blocked, (and if the client is a web client, an error page can also be provided). Actions are not part of action sets, they are configured as a separate part of security policies. For information on configuring policies, see Modifying Security Policies on page 177.
Action Set	One or more pre-defined actions (in the form of action interfaces) grouped together and used to determine the course of action taken when specific conditions are met. Action sets are attached to objects as followed actions. For example, in addition to immediately blocking the source of the hijack session attack, a policy might be configured with a followed action that blocks any additional attempts to connect to the server for 30 minutes. SecureSphere comes with a number of pre-defined action sets. For information on configuring custom action sets, see Creating Custom Action Sets on page 330.
Followed Action	An action set that has been attached to an object and that is invoked when the conditions surrounding this object are met. For information on attaching a followed actions to various SecureSphere components, see Assigning Followed Actions on page 341.
Action Interface	The method by which a followed action is conducted. SecureSphere comes with a number of action interfaces, including sending an e-mail, running a shell command, logging into a system (syslog), and so on. For example, if sending an e-mail is an action interface configured as part of a security policy followed action, an e-mail is generated when the conditions of that security rule are met. For an overview of available action interfaces and their parameters, see Configuring Action Interface Parameters on page 332

Action Set Main Tasks

Integrating action sets into the SecureSphere workflow includes the following basic steps:

Action Sets - Task Overview

	Action	Description	For more information, see...

	Action	Description	For more information, see...
1	**Create an action set.**	Create action set to be assigned as a followed action. You can also use default action sets. For an overview of default action sets, see Working with Default Action Sets on page 329.	Creating Custom Action Sets on page 330.
2	**Configure action interfaces.**	Add and configure action interfaces to the action set that determines exactly the actions that take place.	Configuring Action Interface Parameters on page 332.
3	**Assign as followed action.**	Assign the action set as a followed action to a policy rule, report or task.	Assigning Followed Actions on page 341.

Working with Default Action Sets

SecureSphere comes with a set of default action sets that can be used as is or configured with additional action interfaces, then assigned as followed actions. The following table lists the default Action Sets available in SecureSphere.

Note: Default Action Sets can be customized. Additional interfaces can also be created and configured.

Default Action Sets

Parameter	Description
Default Archive Action Set	Not configured by default. This action set is empty until configured. When a new database service is created and selected, you are prompted to configure its settings. For information on available action interfaces and their configuration for populating this action set, see Configuring Action Interface Parameters on page 332.
Long IP Block	Configured by default with action interface that blocks offending IP address for 3600 seconds.
Long Session Block	Configured by default with action interface that blocks offending session for 3600 seconds.
Long User Block	Configured by default with action interface that blocks offending user for 3600 seconds.
Short IP Block	Configured by default with action interface that blocks offending IP address for 180 seconds.
Short Session Block	Configured by default with action interface that blocks offending session for 180 seconds.
Short User Block	Configured by default with action interface that blocks offending user for 180 seconds.

Creating Custom Action Sets

Custom Action Sets enable you to create an action set for a special purpose, assign it a user friendly name, and configure it with the required action interfaces.

To create a custom action set:

1. In the **Main** workspace, select **Policies > Action Sets**. The **Action Sets** window appears.
2. In the **Action Sets** pane, click **New**. The **Action Set** dialog box appears.
3. Type a name for the action set.
4. In the **Apply to event type** drop-down menu, select an event type to which to apply the action set, then click **Create**. The new action set is created and appears in the **Action Sets** pane. For a list of standard event types and their parameters, see List of SecureSphere Events.



Note: Available action interfaces depend on the event type selected. Not all action interfaces are available for all event types.

5. Add an action interface you want to be part of the action set by clicking . The action interface is added to the **Selected Actions** pane.
6. Configure the action interface added in step 5. For a list of available parameters per interface see Configuring Action Interface Parameters on page 332.
7. Repeat steps 5 and 6 for each action interface you want to be part of the action set.
8. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

EXAMPLE: CONFIGURING AN ACTION SET

Suppose you want to configure SecureSphere with an action set that users can easily identify as being for all general security violations that do not block traffic. You can create a Web Application Layer new action set named WebSecurityNoBlock, assign an action interface that generates a syslog event and creates a review task.

Action Interface Types

An Action Interface's **Type** determines the Action Set event types in which it can be used.

Action Interface Type - Action Set Event Type Correlation

Action Interface Type	Action Set Event Type													
		Security Violation												
	Any event type	All	Service level	DB Service level	Web Service level	Application level	DB application level	Web application level	System Events	Archiving	Audit	Agent	Assessments	Vulnerability

Action Interface Type	Action Set Event Type													
		Security Violation												
	Any event type	All	Service level	DB Service level	Web Service level	Application level	DB application level	Web application level	System Events	Archiving	Audit	Agent	Assessments	Vulnerability
Assignment Task	V	V	V	V	V	V	V	V	V	V			V	V
Email	V	V	V	V	V	V	V	V	V	V			V	V
FTP Archive										V				
SCP Archive										V				
NFS Archive										V				
OS Command	V	V	V	V	V	V	V	V	V	V			V	V
Remedy - Create Incident	V	V	V	V	V	V	V	V	V	V			V	V
Review Task	V	V	V	V	V	V	V	V	V	V			V	V
SNMP Trap	V	V	V	V	V	V	V	V	V	V		V	V	V
System Log	V	V	V	V	V	V	V	V	V	V		V	V	V
Syslog CEF Audit Event											V			
Syslog CEF Gateway Audit Event											V			
Syslog CEF Custom Security Event	V	V	V	V	V				V	V			V	V
Syslog CEF Custom Network Event	V	V	V	V	V				V	V			V	V
Syslog Security Event to Envision	V	V	V	V	V				V	V			V	V
Syslog CEF Security Event	V	V	V	V	V				V	V			V	V
Syslog System Event to Envision	V	V	V	V	V				V	V			V	V
Syslog CEF System Event	V	V	V	V	V				V	V			V	V
Terminate Session							V							

Action Interface Type	Action Set Event Type													
		Security Violation												
	Any event type	All	Service level	DB Service level	Web Service level	Application level	DB application level	Web application level	System Events	Archiving	Audit	Agent	Assessments	Vulnerability
IP Block		V	V	V	V	V	V	V						
IP Monitor		V	V	V	V	V	V	V						
Long Session Block					V									
Session Monitor					V									
User Block			V	V	V	V	V	V						
User Monitor			V	V	V	V	V	V						

Note: There are no pre-defined Action Interfaces of Type HTTP Archive, HTTPS Archive or NFS Archive, but administrators can create new Action Interfaces of these Types. For more information, see the Systems Definitions chapter of **SecureSphere Administration Guide**.

Configuring Action Interface Parameters

Action interfaces are the items that are used by an action set, and subsequently the method by which a followed action is conducted. SecureSphere comes with a number of built-in action interfaces. If they have been set as user configurable by the SecureSphere administrator, then they are customizable in the action interface window.

To configure an action interface's parameters:

1. Select an existing action set or create a new one as described in Creating Custom Action Sets on page 330.
2. Click the green arrow of the **Action Interface** you want to configure. The action interface is moved up to the **Selected Actions** pane.
3. Expand the action interface and configure its parameters as required, and using the items described in the table below.
4. From the upper right-hand corner of SecureSphere, click **Save**.

The following is a list of all action interfaces available by default in SecureSphere and a description of their parameters.

Note: Action interfaces are context sensitive, meaning the action interfaces available depend on the type of action set selected.

Action Interfaces

Action Interface Name	Parameters
Archive to FTP Location	Archives data to a pre-configured FTP location. Only available with archiving events. Only FTP active mode is supported. Configurable parameters include: **Name**: Descriptive name for the action. **Host**: Hostname or IPv4 or IPv6 address of the target server where information should be archived. **Port**: Port used for archiving information **Remote Directory**: Directory on the remote server where information should be archived. **Username**: Username required to gain access to the remote server. **Password**: Password required to gain access to the remote server. **Verify Password**: Required to verify the password already given.
Archive to SCP Location	Archives data to a pre-configured location using SCP (Secure Copy Protocol). Only available with archiving events. Configurable parameters include: **Name**: Descriptive name for the action. **Host**: Hostname or IPv4 or IPv6 address of the target server where information should be archived. **Port**: Port used for archiving information **Remote Directory**: Directory on the remote server where information should be archived. **Username**: Username required to gain access to the remote server. **Password**: Password required to gain access to the remote server. **Use Authentication Key**: Used when an authentication key is required for archiving with Secure Copy. Options are yes and no. Yes means trust has been set up between machines using the server's public key and you do not require authentication. No means you need to provide a username and password. **Authentication Key Path**: When using an authentication key, indicates the path to the management server private authentication key. The default path to the management server authentication key is: • Private Key /opt/SecureSphere/server/.ssh/id_dsa • Public Key /opt/SecureSphere/server/.ssh/id_dsa.pub **Authentication Key Passphrase**: A user configured password that may be required to access the authentication Key.

Action Interface Name	Parameters
Assignment Task	Generates an assignment task. Assignment task parameters include: **Name**: Name for the action interface. **(Parameter) Name**: Name of the task as you want it to appear in the Tasks List. **Due After** (days): Number of days after which the task should be completed. **Priority**: Priority of the task. Options include Low, Medium, High and Urgent. **Followed action on status change**: Conducts attached followed action when status is changed. **Followed action on assignee change**: Conducts attached followed action when assignee is changed. **Followed action on overdue**: Conducts attached followed action when task is overdue. **Add the task owner to email notifications**: Sends e-mail notification to task owner. **Assignee**: Individual responsible for completing the assigned task. Selected from users configured in SecureSphere. For more information on tasks, see Working with Tasks on page 387.
Create a Review Task	Generates a review task. Review task parameters include: **Name**: Name for the action interface. **(Parameter) Name**: Name of the task as you want it to appear in the Tasks List. **Due After** (days): Number of days after which the task should be completed. **Priority**: Priority of the task. Options include Low, Medium, High and Urgent. **Followed action on status change**: Conducts attached followed action when status is changed. **Followed action on assignee change**: Conducts attached followed action when assignee is changed. **Followed action on overdue**: Conducts attached followed action when task is overdue. **Add the task owner to email notifications**: Sends e-mail notification to task owner. **Reviewers**: Reviewers responsible for reviewing the task. Available reviewers include those available in SecureSphere. For more information on tasks, see Working with Tasks on page 387.

Action Interface Name	Parameters
Log Audit Events to System Log **(Gateway Syslog)** **Log Audit Events to System Log (ArcSight CEF)**	Sends audit information to an external SIEM. Can alternatively configure SIEM interface directly from a Gateway Group. For more information on configuring a gateway group with External Logger see the Administration Manual. Audit syslog parameters include: **Name**: Descriptive name for the action. **Protocol**: Protocol used to send the information. If using a system with failover capabilities, select TCP. **Primary Host**: Type the IP address or host name of the primary SIEM logger. **Primary Port**: Type the port number of the primary SIEM logger. **Secondary Host**: Type the IP address or host name of the secondary SIEM logger if available. **Secondary Port**: Type the port number of the secondary SIEM logger if applicable. **Note:** Secondary connections are configured when in an environment with failover capabilities. When configuring secondary options the action interface protocol should be configured as TCP. **Syslog Log Level**: Type of syslog event. Options include Info, Warn, Debug and Error. **Message**: Actual body of the syslog event. This may contain information formatted to be assimilated by third party systems. Can use predefined placeholders as described in Working with Placeholders on page 494. **Note:** A separate Action Interface for ArcSight systems with a predefined message in CEF format is available. **Facility**: Select the relevant facility (your company's categorization for the source of the specific event.)
IP Block **User Block HTTP Session Block**	Three different blocking interfaces may be available, based on IP address, User or HTTP Session. Blocks an IP address for a designated period of time. IP Block parameters include: **Name**: Descriptive name for the action. **Duration** (seconds): Period of time for which to block the item. **Trusted IPs**: IP addresses that must not be blocked, a white list (only available with IP Block). **Note:** The SecureSphere Gateway which encountered the event immediately implements the block and then notifies the MX, which sends this information to other Gateways so that they will implement the block as well.

Action Interface Name	Parameters
Server System Log > Log <...> event to System Log (syslog) using the CEF standard	Sends the indicated type of ArcSight Event message. Use the relevant System Log action interface when creating action sets for the various types of events. If you use for example a security action interface for a system event, the event will be created in ArcSight, but its information will be irrelevant. ArcSight parameters include: **Name**: Type a name for the syslog event. **Syslog Host**: Type the IP address or host name of the ArcSight server. **Syslog Log level**: Select the desired syslog log level from the dropdown list (info, warn, debug or error). **Message**: Type a message with placeholder information to be used by syslog to create a message readable by ArcSight. This message must follow CEF guidelines. For a description of the CEF syslog message, including syntax and available placeholders, see Examples of Syslog Events for ArcSight on page 536. The message field may be locked by your SecureSphere administrator. In this case the message is predefined and can only be modified by the administrator. **Facility**: Select the required facility (type of authorization required by your installation of Arcsight). **Run on every event**: Select this checkbox if you want to send a syslog event to Arcsight for every event. For more information, see Run on Every Event on page 341.
Server System Log > Log to System Log (syslog)	Sends a syslog message from the MX. Syslog parameters include: **Name**: Descriptive name for the action. **Syslog Host**: IP address or hostname of the computer hosting the target syslog application. **Syslog Log Level**: Type of syslog event. Options include Info, Warn, Debug and Error. **Message**: Actual body of the syslog event. This may contain information formatted to be assimilated by third party systems. Can use predefined placeholders as described in Working with Placeholders on page 494. **Facility**: Select the relevant facility (your company's categorization for the source of the specific event). **Run on Every Event**: Sends a syslog as configured for every event. For more information, see Run on Every Event on page 341.
Gateway Security System Log > Log <...> event to System Log (Gateway syslog)	These action interfaces are similar to the **Server System Log** > events, except that the syslog is sent from the SecureSphere Gateway rather than from the MX. **Note:** When a placeholder is added to a Gateway Security System Log action interface or to an action that includes a **Gateway Security System Log** action interface, SecureSphere validates that the placeholder is accessible to the Gateway. If it is not, SecureSphere does not allow the action or the interface to be saved.
IP Monitor >Monitor an IP **User Monitor > Monitor a User** **Session Monitor > Monitor an HTTP Session**	Begins monitoring all activity regarding the selected item, for all practical purposes auditing that activity. Resulting activity is then displayed in the relevant alert in SecureSphere. Three different monitoring interfaces may be available, based in IP address, HTTP session or user. Defines the parameters for monitoring including: **Name**: Descriptive name for the action. **Duration**: Period of time to monitor the item.

Action Interface Name	Parameters
Remedy Create Incident	Sends a pre-formatted message to a Remedy server. Remedy parameters include: **Name**: Type a name for the Syslog event. **XML Soap Request**: A pre-configured soap request used pass information to the Remedy server. **Host**: Host name of the web-server. Can be an IP address or web host name. **Remedy Protocol**: Protocol used to communicate with Remedy. Options include HTTP and HTTPS. **Port**: Port number used to communicate with Remedy. **Remedy Server**: Name of the Remedy server that runs on the web server. Type a valid HTTP name. **Web Service**: Web service name. This field is defined by the SecureSphere administrator and locked for editing. **First Name**: First name of the Remedy ITSM user which will be the owner of the incident. **Last Name**: Last name of the Remedy ITSM user which will be the owner of the incident. **Note:** The first and last name fields represent the incident reporter (owner) in Remedy, and must match an existing Remedy user. **Impact**: Impact level for the incident. Options include 1-Extensive/Widespread, 2-Significant/Large, 3-Moderate/Limited, 4-Minor/Localized. **Urgency**: Urgency level of the incident. **Incident Summary Template**: Summary of the incident (title). This corresponds to the $(summary) parameter in the XML Soap Request. You can specify one or more placeholders (see Working with Placeholders on page 494 for more information). **Incident Notes Template**: Notes of the incident (detailed description). This corresponds to the $(notes) parameter in the XML Soap Request. You can specify one or more placeholders (see Working with Placeholders on page 494 for more information). **CI Name:** (Configuration Item) - optional. For example, name of the server in Remedy linked to a particular DB service). Together with Lookup Keyword (below), this represents a configuration item that is pre-configured on the Remedy server. The CI is associated to all incidents opened by this action interface. **Lookup Keyword**: Optional. This is the class type of the CI specified in the setting above. This field is required when CI is defined.
Server System Log > Log <...> event to System Log (syslog) using the CEF standard	Sends the indicated type of ArcSight Event message. Use the relevant System Log action interface when creating action sets for the various types of events. If you use for example a security action interface for a system event, the event will be created in ArcSight, but its information will be irrelevant. ArcSight parameters include: **Name**: Type a name for the syslog event. **Syslog Host**: Type the IP address or host name of the ArcSight server. **Syslog Log level**: Select the desired syslog log level from the dropdown list (info, warn, debug or error). **Message**: Type a message with placeholder information to be used by syslog to create a message readable by ArcSight. This message must follow CEF guidelines. For a description of the CEF syslog message, including syntax and available placeholders, see Examples of Syslog Events for ArcSight on page 536. The message field may be locked by your SecureSphere administrator. In this case the message is predefined and can only be modified by the administrator. **Facility**: Select the required facility (type of authorization required by your installation of Arcsight). **Run on every event**: Select this checkbox if you want to send a syslog event to Arcsight

Action Interface Name	Parameters
	for every event. For more information, see Run on Every Event on page 341.

Action Interface Name	Parameters
Run a Shell Command	Sends a shell command. You can use this for example, to automatically send an SNMPv3 trap to alert a monitoring system outside of SecureSphere of a critical event such as a Gateway being disconnected. To invoke an OS command, the management server user must have execute permissions (+x) on the executable binary or script file. Configurable parameters include: **Command:** The command you want to execute. **Name**: Descriptive name for the action. **Arguments**: Arguments used with the command. **Working Dir**: File system location of the command. **Run on Every Event**: Runs the specified Shell Command with every event. For more information, see Run on Every Event on page 341. **Note:** The OS command is run using permissions available to the mxserver user, which does not have rights to create directories under the root directory.
Send an E-mail	Sends an e-mail. Configurable parameters include: **Name**: Descriptive name for the action. **SMTP Address**: SMTP address of the mail server. **Source Email Address**: Address of the account being used to send the e-mail. You can enter multiple e-mail addresses by separating them by commas. **Destination Address**: Address to which to send the e-mail. **Note:** To enter multiple email addresses, separate them with a comma (""). **CC Address**: Additional address if desired. **Note: To add multiple email addresses, separate each individual address with a comma (",").** **Email Subject**: Subject you want to assign to e-mails sent using this action interface. You can type in a subject or populate the subject based on job type and format. You can optionally use a default e-mail subject provided by SecureSphere by enabling the **Use Default** option. **Email Content**: Body of the e-mail you want to send using this action interface. You can type in content, or populate the content based on job type and format. You can optionally use default e-mail content provided by SecureSphere by enabling the **Use Default** option. **Email Format**: E-mail format, options include HTML or text. **Run on Every Event**: Sends an e-mail as configured for every event. For more information, see Run on Every Event on page 341. **Note:** When attaching the **Send an E-mail** action interface to a report, a copy of that was report is attached to the e-mail. You can alternatively have your SecureSphere Administrator configure SecureSphere to send a link instead of an attachment under **Admin** > **System Definitions** > **Report Settings**. **Note:** When sending a link instead of an attachment, you need to additionally configure a URL in this same location.

Action Interface Name	Parameters
SNMP Trap	Sends an SNMP trap. SNMP trap parameters include: **Name**: Descriptive name for the action. **SNMP Host**: Host of the SNMP Agent. **SNMP Port**: Port used to send the SNMP trap. **Run on Every Event**: Send an SNMP trap as configured for every event, see Run on Every Event on page 341. **SNMP Community String**: Community string (authentication string) used to authenticate the sending system up against the agent. The parameter value must match the parameter set defined on the SNMP server to which the notification is sent to. The mib is in the following location: `/opt/SecureSphere/server/SecureSphere/jakarta-tomcat-secsph/webapps/SecureSphere/WEB-INF/conf/imperva-mib-2.txt`.
Terminate Session	Terminates the database session in which the event occurred. Can be used to block traffic which cannot be blocked over the network (for example, when SecureSphere is deployed in sniffing mode or traffic originated in DB agents). Currently available for Oracle and MSSQL. **Note:** This action interface is only available when creating a security policy in the DB service or DB application level. When this followed action is invoked, the SecureSphere Management Server connects to the relevant database, and instructs it using SQL to terminate the session. SecureSphere takes the database name and IP address of the event, searches for the direct access information of the database connection for the relevant service (service configuration > Direct Access Information), then uses these credentials to connect directly to the relevant database and terminate the session. In order to directly access the database for the relevant service and terminate the session, there must be credentials that have been configured specifically for this IP address and database name. Additionally, the service name on which it is located must be unique in SecureSphere. For more information on configuring database direct access, see Configuring Database Direct Access Information. **Note:** In order to have this action work only on traffic originating in DB agents, create a custom policy, add the Originated in Agent criteria, and set it to yes. The user that has been configured for direct database access requires the following database permissions: • **Oracle**: create session, alter system, and select on sys.v_$session • **MSSQL**: must be a member of sysadmin or processadmin server roles Terminate Session parameters include: **Name**: Descriptive name for the action. **Lock User Account**: Locks the user account in the database. The user will not be able to connect to the database until the account is manually unlocked by a DBA. **Note: Lock User Account** cannot be used for MSSQL 2000. **Run on Every Event**: Send the terminate session message for every event, see Run on Every Event on page 341.

Run on Every Event

By default, alerts are generated for up to the first 30 events which are added to the relevant alert aggregation. This number is configurable by the SecureSphere administrator. Once SecureSphere reaches this threshold, it no longer generates alerts for subsequent events. You can configure action interfaces to generate an alert for every event.



Note: When violations are aggregated, followed actions are only applied to the first event of the aggregation, unless the action interface connected with the alert has been configured to Run on Every Event.

Assigning Followed Actions

Followed actions are another name for action sets that is used when an action set is assigned to a SecureSphere object, and that are invoked when specified conditions are met. Action sets can be added as followed actions to a number of elements in SecureSphere. These include security policies, system events, audit policies, reports, active modules, and tasks. The following table lists the various SecureSphere objects to which action sets can be assigned as followed actions.

Followed Actions in Various SecureSphere Modules

SecureSphere Module	Description
Security Policy	Select **Policies > Security**. Action set can be added as followed action that takes place following a security event. For more information, see Modifying Security Policies on page 177.
System Event	Select **Policies > System Events**, then click the **Followed Action** tab. Action set can be added as followed action that takes place following a System Event. For more information, see System Event Policies on page 411.
Report	Select **Reports > Manage Reports**, then select a report. Action set can be added as followed action to a report from the **General Details** tab. For more information, see Configuring Reports With Followed Actions on page 236.
Task	Action set can be added as followed action to Assignment Tasks to enable the configuring of events following actions taken on tasks such as a status change, or if an event is overdue. For more information, see Working with Tasks on page 387.
Assessment	Action set can be added to an assessment job in the Job Details tab of an assessment. For more information see Creating and Configuring a Web Scanner Integration Policy on page 138.
Maintenance	Administrative maintenance tasks involving archiving also involve configuring action sets as part of the archive export including Definitions archiving, Reports archiving and System Event Archiving. For more information see the SecureSphere Administration Guide.

For information on configuring custom Action Sets to be assigned as followed actions, see Configuring Action Interface Parameters on page 332.

Directly Assigning Followed Actions from the Security Policy Window

You can directly assign a followed action to one or more policies in the main policies window. This enables you to apply a single followed action to a number of policies at once, eliminating the need to go into each policy separately and configure it with the desired followed action.

This can help, for example to configure a followed action to send all events generated by a policy to a syslog server. You can configure an action set that contains an action interface and the required parameters, then directly assign that action set to the required policies using right-click functionality.

For more information on right-click functionality in policies, see Directly Configuring Policies from the Security Policies Window on page 171.

To directly assign a followed action from the Security Policies window:

1. Configure a followed action for applying to multiple policies, for example a syslog followed action. For more information on configuring followed actions, see Action Set Main Tasks on page 328.
2. In the **Main** workspace, select **Policies > Security Policies**. The **Security Policies** window appears.
3. Configure a filter to display only those policies that are relevant. For more information on filtering policies, see Filtering Policies on page 181.
4. Select each of the policies you want to apply the filter to by clicking on them once.
   - To select a series of policies, click the first policy in the series, then shift + click the last policy of the series.
   - To select a number of policies not in a series, ctrl + click each policy you want to select.
5. Once you have selected all of the desired policies, right-click on one of the selected policies, then choose **Set Followed Action**. The **Set Followed Actions for All Selected Policies** window opens.



Note: Followed actions that appear in the dropdown menu include all the followed actions available for the selected policies.

6. From the Followed Actions dropdown menu, select the followed action configured in step 1.
7. Click Save, the followed action is applied to the selected policies and the changes are saved.

Working with Global Objects

Global objects are objects created once and maintained in a central location that are used globally with a number of SecureSphere features. Global objects provide a wide variety of functions. For example, they can provide an additional level of granularity that assists in focusing on a specific set of IP addresses, or advanced functionality such as rules that guide the rewriting of URLs for use with load balancers.

For example, one of the applications SecureSphere is protecting is a Web server. You decided that traffic from a particular group of hosts in the company network are not a threat and do not need to be monitored when accessing this server. SecureSphere enables you to configure a global object called an IP group with a list of IP addresses whose traffic is excluded from monitoring (for example, private address range 10.0.0.0), then configure the profile for web traffic to exclude monitoring on this IP Group. This would exclude all traffic between the private addresses and your web server from being monitored, reducing the amount of traffic that has to be protected and the processing and alert management overhead.

For a list of available Global Objects and a short description of each, see Summary of Global Objects on page 343.



Note: Global Objects are typically not stand alone features. They are usually configured as part of a larger configuration task.

To view global objects:

- From the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.

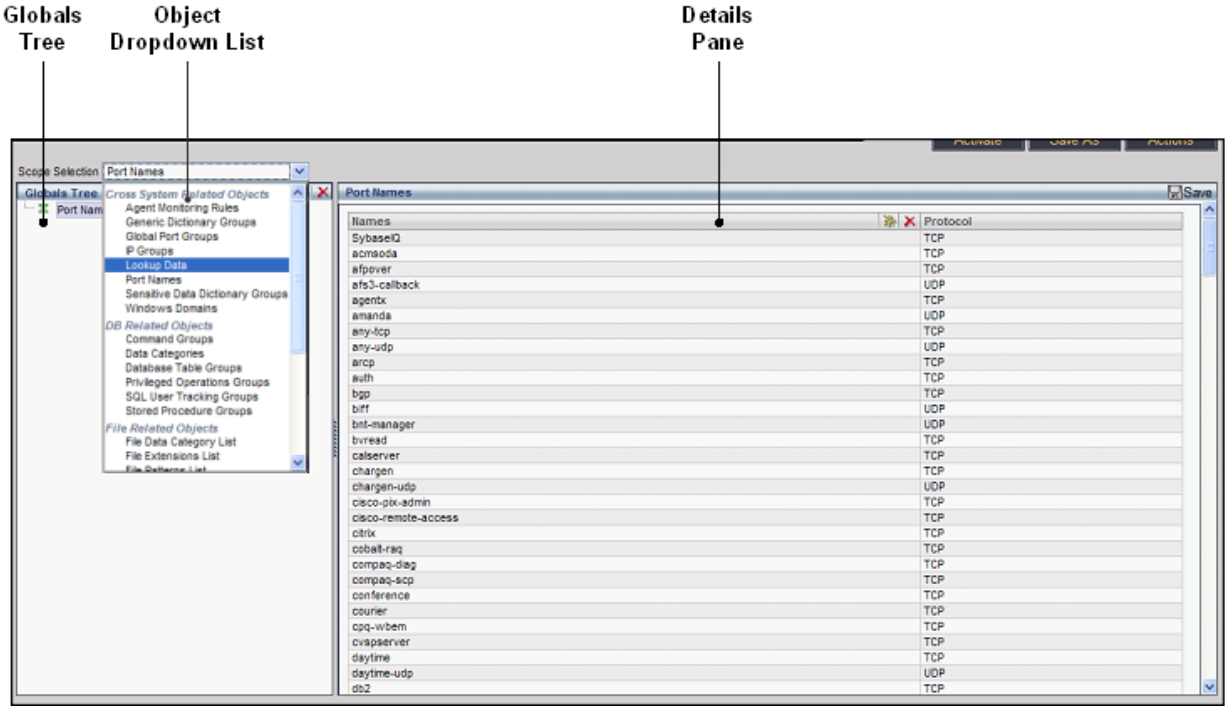


Figure 63: Global Objects Window With Port Names Object Selected

The Global Objects window has two primary sections:

- **Object Dropdown List:** Lists available global objects by category. For a list of available global objects, see Summary of Global Objects on page 343
- **Globals Tree**: Lists the available global objects for the category selected in the **Object Dropdown List**.
- **Configuration Pane**: Displays configuration options for the currently selected global object.

Summary of Global Objects

The following table provides an overview of the Global Objects available in SecureSphere and the features in which they are used. Instructions on configuring these objects can be found in their relevant configuration guidelines, as indicated by the link for each object.

General Global Objects

Parameter Name	Description
IP Groups	Creates a group of IP addresses, then used to assist SecureSphere identifying traffic for inclusion or exclusion from SecureSphere operation. For information on configuring IP Groups, see Configuring IP Groups on page 347.
Lookup Data	Enables you to create custom groups that can be used to protect and enrich information and events with data imported from external systems like LDAP and ticketing systems. For more information, see Protecting and Auditing Custom Information with Lookup Data on page 363.
Global Ports Group	Stores the Default Port List used to map application protocols to ports and help SecureSphere identify traffic running in your network. For more information on configuring Global Ports Groups, see Working with Global Ports Groups on page 347.
Windows Domains	Defines the various aspects of Windows Domain connectivity for domain controllers, active directory, and DNS for use when configuring external systems, Kerberos compatibility and more. For more information, see Configuring Windows Domains on page 358 and Configuring Kerberos Support.
Generic Dictionary Group	Defines potentially malicious patterns that may pose a risk to your data. For more information, see Generic Dictionary Groups on page 361.
Sensitive Data Dictionary Groups	Defines sensitive data patterns that need to be protected by SecureSphere. For more information, see Using Sensitive Data Dictionary Groups on page 350.

Web Global Objects

Parameter Name	Description
CA Group	Used to group a number of security certificates together then be selected in a Web service's Reverse Proxy settings. For more information, see Configuring Certificate Authority Groups on page 437.
Certificate Authorities	Enables the import of certificates from a certificate authority (such as Verisign), which are then used for validation. For more information, see Configuring Certificate Authorities on page 436.
Certificate Revocation Lists (CRLs)	Enables the import of Certificate Revocation Lists from a valid certificate authority which can then be used during certificate validation in SecureSphere. For more information, see Configuring Certificate Revocation List (CRL) on page 437.
Client Certificate Parameter Set	When validating certificates, determines which parameter will be taken from the certificate received from the client and passed on to the web server. For more information, see Configuring Client Authentication Rules on page 438.
Data Masking Group	Defines parameters that are determined to be sensitive (for example, password). Enables you to group these sensitive parameters then attach this group to a policy or service for supervision. For more information, see Configuring Sensitive Data Protection on page 350.
File-Type Group	Defines lists of file types determined as not posing a threat, for exclusion from SecureSphere operation. For information on configuring file type restrictions, see Configuring Static File Lists on page 417.
Gateway HTTP Proxy	Defines a list of HTTP proxies for use with gateway groups in the **Main** > **Gateways** window. For information, see Configuring Gateway HTTP Proxies on page 426.
Parameter Group	Defines lists of static parameters determined as not posing a threat for exclusion from SecureSphere protection. For information on configuring parameter restrictions, see Configuring Parameter Groups on page 416.
Parameter Type Configuration	Defines lists of parameters and enables the creation of rules to enforce a complex format when received as part of web traffic. For more information see Configuring Parameter Type Configuration Objects on page 417.
Soap Elements Group	Defines lists of SOAP elements determined as not posing a threat for exclusion from SecureSphere protection. For information on configuring SOAP element restrictions, see Configuring SOAP Elements on page 417.
Session Identifiers Group	Defines a list of identifiers used to track sessions and block traffic when a policy is violated. For information on configuring session identifiers, see [onfiguring session tracking xxx]
Susceptible Directory Group	Defines a list of directories which have been identified as needing to be protected in the event of a zero-day webworm attack. For information on configuring web worm protection, see Configuring Susceptible Directories on page 426. Default: Default Susceptible Folder List
URL Prefixes/Dictionary Group	Defines a list of URL prefixes or directories which can be attached to a web application and used to restrict monitoring of a specific list of URLs, or alternatively exclude URLs or directories that do not need to be monitored. For more information, see Configuring URL Prefixes / Directory Based Restrictions on page 420.

Parameter Name	Description
URL Rewrite Groups	Defines the rules used to rewrite URLs in Transparent Reverse Proxy mode, when load balancers are part of your network topology. For information on configuring rewrite, see URL Rewrite on page 428.
Web Error Pages Group	Defines policies when errors are encountered with web traffic and enables you to create customized error pages to display to users. For information on configuring web error page policies, see Basic Web Service Configuration Tasks on page 59.

Creating New Global Objects

New global objects can be created for any of the existing global object categories. While the procedure of creating new objects is standard for most objects, there are some that require special configuration. This procedure describes how to create Global Objects that do not require special configuration.



Note:

- In any SecureSphere window where the  icon is displayed to the right of a global object, you can create a new global object or edit an existing one by clicking the icon. For example, when configuring source restrictions on a server group you can create an IP group from the **Definitions** tab of the Server group by clicking the  icon displayed to the right of **Ignore IP Group**. After you configure the global object, it will appear in the dropdown list from where you started.
- For information on creating IP Groups, see Configuring IP Groups on page 347.

To create a Global Object:

1. In the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Objects** dropdown list, select the category of global object you want to create by clicking on it once.
3. In the **Globals Tree**, select the desired object.
4. Click  **New**. The **Create Object** dialog box appears for the object you want to create.
5. Enter a name for the Global Object.
6. Select the option to define the basis of your new object. You have two options:
   - **From Scratch**: Creates a new global object for the category selected.
   - **Use existing**: Enables you to select an existing global object to be used as the basis for the new object. If you click this radio button, select the existing global object to be used as the basis for the new object from the **Use existing** drop-down list.
7. Click **Create**. The new global object is created and added to the relevant category.
8. Configure the new global object as required.

To create a Global Object "on the fly":

In any SecureSphere window where the icon is displayed to the right of a global object, you can create a new global object or edit an existing one by clicking the icon. For example, when configuring source restrictions on a server group you can create an IP group from the **Definitions** tab of the Server group by clicking the icon displayed to the right of **Ignore IP Group**. After you configure the global object, it will appear in the dropdown list from where you started.

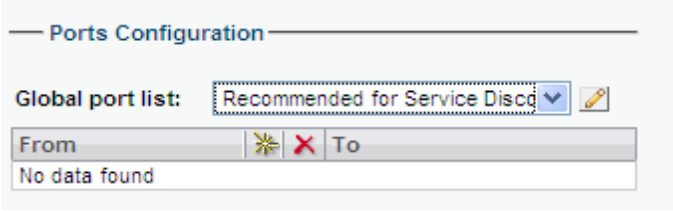


Figure 64: Creating a new Global Object "on the fly"

Working with Global Ports Groups

The Global Ports Group houses the location of the Default Port List, used to map application protocols to ports and help SecureSphere identify traffic running in your network, and to determine the ports used during discovery when applicable. The default list determines all ports that are identified by SecureSphere.



Note: You cannot modify the Default Global Ports list, but you can create a new list based on the default and containing all of its objects.

To add ports to a custom Port List:

1. In the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Objects** dropdown list, select **Global Ports Groups**. Existing **Global Ports Groups** are displayed.
3. In the **Globals Tree**, select an existing Global Ports group or create a new group as described in Creating New Global Objects on page 346. The selected IP Group appears in the pane on the right-hand side of the screen.
4. From the **Global Ports Group** pane, click **New.** A new line is added to the **Protocol** table.
5. From the Protocol drop-down list, select the desired protocol.
6. In the **Ports** field, type the port or range of ports to map to the protocol selected in step 5.
7. Repeat steps 4 - 6 as required.
8. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Configuring IP Groups

IP Groups are groups of IP addresses with some form of relationship, used to assist SecureSphere identifying traffic for inclusion or exclusion from SecureSphere monitoring.



Note: You can enter both IPv4 and IPv6 addresses in any of the three IP Group types (single, range, and network), though a range can consist of addresses of only one type.

To modify an IP Group:

1. In in **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Objects** dropdown list, select **IP Groups**. Existing **IP Groups** are displayed.

3. In the **Globals Tree**, you can create a new IP Groups either by:
   - Importing them from a CSV file (Importing Protected IP Addresses from a CSV File)
   - Adding them individually (Creating an IP Group Individually on page 349)

Importing an IP Group from a CSV File

To import an IP Group from a CSV file:

1. Click **Upload from CSV**.
2. In the **CSV Upload** window, in the **General** tab, enter the name of the CSV file in **Select file to upload** or click **Browse** and browse to the file.

   The format of the CSV file should be as follows:
   - The first line may contain only headers, in which case you should check **Use first line as header** in the **Advanced** tab.
   - Each line should contain up to 4 comma-separated fields, in one of the following formats:

     SINGLE, <IP Address>

     RANGE, <IP Address>, <IP Address>

     NETWORK,,, <IP Address>/<number of network bits>

     NETWORK,,,123.124.125.126/24
   - Each IP Address can be in either IPv4 or IPv6 format. For RANGE, both must be in the same format.

     Examples:

     SINGLE, 2001:db8:0:0:1:0:0:1

     RANGE,10.1.2.3,10.1.2.12
   - NETWORK must be in the CIDR format, where the number of network bits is between 1 and 32 inclusive for IPv4 addresses and between 1 and 128 inclusive for IPv6 addresses. In the example below, it is 24.

     Example:

     NETWORK,,,123.124.125.126/24

   Note: A maximum of 5,000 IP groups can be uploaded at one time. You should allow a few minutes for 5,000 IP groups to be uploaded, and a proportionate amount of time for other numbers of IP groups.
3. In the **Advanced** tab, optionally specify the parameters listed in the following table.

   Note: You can specify defaults for the fields in the **Advanced** tab in **Admin** > **Systems Definitions** > **CSV Upload Default Settings**.
4. You can move the CSV upload job to the background by clicking **Move to Background**.

   Only one CSV upload job can be running at the same time. You can see previous CSV upload jobs in **Admin** > **Job Status**.

CSV Upload - Advanced Tab Parameters

Parameter Name	Description
Upload Settings	Select one or more of: **Case Sensitive** - The data in the CSV file are case sensitive, so if existing data match the data in the CSV file, they are not considered duplicates. **Note:** In the case of IP Groups, this parameter has no effect, since IP Addresses are numbers, including IPv6 addresses which can contain hexadecimal digits. SecureSphere does not distinguish between "a" and "A", for example, in IPv6 addresses. Also, SecureSphere does not distinguish between "RANGE" and "range" or "RanGe", for example. **Use first line as header** - The first line of the CSV file does not contain data to be imported, but column headings. **Override existing entries** - If the data in the CSV file matches existing data, then the data in the CSV file will overwrite the exiting data.
Format	Select **UNIX** or **Windows**.
Encoding	Select either **ASCII** or **UTF8**.

Creating an IP Group Individually

To create an IP Group individually:

1. From the **IP Group** pane, click  **New.** A new line is added to the **IP Groups** table.
2. Configure the IP Group's properties as follows:
   - **Type**: Select how you want to define the IP Group. The options are: a single IP address, an IP range, and a network address.
   - **Single IP Address:** Enter the IP address in **Start IP Address**.
   - **IP Range:** Enter the **Start IP Address** and the **End IP Address**.

     Note: You can enter either IPv4 or IPv6 addresses in **Start IP Address** and **End IP Address**, but you cannot mix IPv4 and IPv6 addresses in the same group.
   - **Network:** Enter the network IP address in CIDR notation.

     Both IPv4 and IPv6 addresses must be entered in the CIDR notation (see CIDR Notation on page 47). Network and subnet masks are no longer used.
3. Repeat steps 4 to 5 as required.
4. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

   You can now associate the IP group as required.

EXAMPLE: USING AN IP GROUP TO MONITOR WEB TRAFFIC

One of the applications SecureSphere is protecting is a Web server. You decided that traffic from hosts in the company network are not a threat and do not need to be monitored when accessing this server. SecureSphere enables you to configure a global object called an IP group with a list of IP addresses who's traffic is excluded from monitoring. You configure the IP group to represent the private address scheme of 10.0.0.0 and then configure this IP group as excluded in the profile for web traffic. This excludes all traffic between the private addresses and your web server from monitoring, reducing the amount of traffic that has to be protected and eliminating processing overhead.

Configuring Sensitive Data Protection

SecureSphere offers a number of ways that can be employed to prevent the leakage of sensitive data such as social security numbers, account balances, PINs, passwords, and more. Deciding on which of these features to use depends on your specific needs. These options include:

- **Sensitive Data Dictionaries**: Configurable dictionaries used to prevent the leakage of sensitive data by matching it to pre-defined patterns such as Social Security or Credit Card Numbers. These dictionaries can then be used by policies or directly attached to services. Data that matches configured patterns is automatically masked (hidden). For information on configuring sensitive data dictionaries, see Using Sensitive Data Dictionary Groups on page 350.
- **Masking Data in Elements**: Performs masking on web services according to element name by enabling you to identify specific parameters that are sensitive and that need to be masked (for example, a password). For information on creating and configuring data masking lists, see Masking Data in Elements on page 356.

When data is masked, it is replaced by a row of asterisks (*), while the original data is eliminated (and cannot be retrieved) from the SecureSphere log, the number of asterisks corresponds to the number of bytes hidden.



Notes:

- Data masking affects only what users see. SecureSphere always operates on the complete, unmasked data, when referring to tasks like matching signatures or building profiles.
- You can view or change only the dictionaries for which you have Edit permissions.

Using Sensitive Data Dictionary Groups

Sensitive Data Dictionary groups in SecureSphere are groups of data patterns used to identify sensitive data and prevent data leakage. Once configured, SecureSphere looks for these predefined patterns in specific locations within data.

Sensitive data dictionary groups are used to detect sensitive data such as telephone numbers, ID numbers, social security numbers, credit card numbers, then mask this data. Masking is conducted by the Gateway before data is sent to the management server. Once a sensitive dictionary has been configured, you can define security and audit policies that detect and prevent exposure of sensitive data to unauthorized users based on these groups, and configure SecureSphere services to mask data in alerts, audits and logs.

When sensitive data dictionary groups are attached directly to a service, the service parses all data searching for the patterns defined in the dictionary and subsequently masks it when detected.

- For a list of typical use cases for sensitive data dictionary groups, see Sensitive Data Dictionary Groups Use Cases on page 350.
- For Sensitive Data Dictionary Configuration Guidelines see Using Sensitive Data Dictionaries Configuration Guidelines on page 351.
- For instructions on creating a new dictionary groups, see Creating New Dictionary Groups on page 351.

Sensitive Data Dictionary Groups Use Cases

Sensitive Data Dictionary Groups can be used in various ways in SecureSphere. The following are examples:

- **Preventing data leakage in web responses**: Identifying sensitive data in HTTP responses and then blocking them or generating alerts about them. For example, preventing exposure of credit card numbers that someone tries to steal from a protected web site.
- **Masking sensitive data in alerts**: Masking sensitive data that appear in the SecureSphere Alerts table. For example, masking credit card numbers that appear in alerts in order to avoid a situation in which all SecureSphere users can view full credit card numbers.

Using Sensitive Data Dictionaries Configuration Guidelines

Configuring SecureSphere to use sensitive data dictionaries can involve a number of tasks, some of which are optional. For example, you can use an existing dictionary group as opposed to create a new one. Dictionary groups can be used by either a security or audit policy, they do not have to be used by both. The following table lists the main tasks that can be conducted to configure data protection using Sensitive Data Dictionaries.

Configuring Sensitive Data Task Overview

	Action	Description	For more information, see...
1	**Define a new Sensitive Data Dictionary Group.**	Define a new Sensitive Data Dictionary Group to list data patterns you want to protect. Alternatively, you can use an existing Sensitive Data Dictionary Group.	Creating New Dictionary Groups on page 351.
2	**Add data patterns to the Sensitive Data Dictionary Group.**	Add patterns of sensitive data for SecureSphere to protect and configure additional options.	Adding Data Pattern Definitions and Validation Method to Dictionary Groups on page 352.
3	**Assign Sensitive Data Dictionary to web security policy (optional).**	Assign sensitive data dictionary group to security policies to protect data by security policy.	Web Application Custom Policies on page 210: Example - Preventing Data Leakage in Web Responses
4	**Assign Sensitive Data Dictionary to database security policy (optional)**	Assign sensitive data dictionary group to database policies to protect data by security policy.	Web Application Custom Policies on page 210: Example - Violation Caused by Specific Users
5	**Assign Sensitive Data Dictionary to a service (optional).**	Attach sensitive data dictionary group to the desired service to protect data by service.	Attaching a Data Masking List to a Service on page 357
6	**Assign Sensitive Data Dictionary to audit policy (optional).**	Attach sensitive data dictionary group to audit policies to protect data by audit policy.	Working with Audit Policies



Note: When configuring policies using the **Sensitive Dictionary Search** match criteria, setting the **Search Mode** to **Exact** has no effect for the following Locations: Parameters, URLs and Parameters. The match is always performed as in **Contains** search mode. In addition, the **Limited Values** definitions within the dictionaries are meaningless for these locations.

Creating New Dictionary Groups

This procedure describes how to create new Dictionary Groups.



Note: For an example of how to configure SecureSphere with Sensitive Data Dictionaries to prevent data leakage, Web Application Custom Policies on page 210: Example - Preventing Data Leakage in Web Responses.

To create a new dictionary group:

1. In the **Main** workspace, click Setup > Global Objects. The Global Objects window appears.
2. In the **Objects** dropdown list, select one of the following:
   - Generic Dictionary Group: Used to identify and control suspicious behavior that may be posing a threat to your data or network. For more information see Generic Dictionary Groups on page 361.
   - **Sensitive Data Dictionary Group**: Used to identify sensitive data and prevent data leakage. For more information see Configuring Sensitive Data Protection on page 350.

   Existing Dictionary Groups are displayed for the item you selected in the **Globals Tree**.
3. In the **Globals Tree** pane, right-click the folder of the Dictionary Group you want to create, then click **Create**. The **Create Dictionary Group** dialog box appears.
4. In the **Name** text box, type the name of the new dictionary group.
5. Define a new group by doing one of the following:
   - To define a new group on the basis of the existing one, select **Use Existing** and browse to find the group that you want to use.
   - To define a new group from scratch, select **From Scratch** and set the parameters in the table **Dictionary Group Parameters** below.
6. Click **Create**. The **Create Dictionary Group** dialog box closes and the new group appears in the **Globals Tree** pane.
7. Add data pattern definitions to the dictionary group as described in Adding Data Pattern Definitions and Validation Method to Dictionary Groups on page 352.

Dictionary Group Parameters

Option	Description
Case Sensitive (check-box)	Enables pattern match only if the case of the pattern defined in the dictionary group is the same as the case of the pattern that is found by the search engine. **Default**: Not Selected.
Limited Values	Enables performing search only within the range of the predefined number of bytes. SecureSphere counts the number of bytes only after the UTF-8 conversion, which means that it counts the number of bytes and not the number of the letters, as in some languages a letter can be represented by a number of bytes. Configuring the range is important for narrowing down the search, and improving performance. For example, a dictionary of phone numbers should be limited to sizes 5-15, but note that this range (5-15) applies to the total length of the response, so if more than one phone number is returned, the limit will be exceeded. First, select **Min/Max** and then type the number in the text box. **Default**: No limit.

Adding Data Pattern Definitions and Validation Method to Dictionary Groups

SecureSphere enables you to add data patterns to new or existing Dictionary Groups. By identifying these patterns, you inform SecureSphere what data to mask. This procedure describes how to add data patterns to existing Dictionary Groups. For instructions on how to create a new Dictionary Group, see Creating New Dictionary Groups on page 351.



Notes:

- An alternate method for populating Dictionary Groups is by importing values from a CSV file. For more information, see Importing Data from a CSV File on page 355
- For an example of how to configure SecureSphere with Sensitive Data Dictionaries to prevent data leakage see Web Application Custom Policies on page 210: Example - Preventing Data Leakage in Web Responses.

To add data pattern definitions to a Dictionary group:

1. In the **Main** workspace, click Setup > Global Objects. The Global Objects window appears.
2. In the **Objects** dropdown list, select the type of dictionary group to which you want to add patterns. Options include Generic Dictionary Groups and **Sensitive Data Dictionary Groups**. Existing Dictionary Group**s** are displayed for the item you selected in the **Globals Tree**.
3. In the **Globals** Tree pane, select the dictionary group you want to edit. The selected dictionary group's details appear in the **Details** pane.
4. In the **Display Name** title bar, click , a new row is added to the bottom of the dictionary table.
5. Configure the desired parameters, as shown in the table **Sensitive Data Dictionary Group Options** below.
6. Select a **Validation Method** if desired. This implements an additional method of data validation on top of advanced pattern matching to prevent false positives. The **Luhn** method is commonly used to validate credit card numbers and other data such as social security numbers.
7. Configure other options including whether or not pattern matching should be case sensitive, and minimum and maximum values.
8. Click **Save**. Settings are saved and the new dictionary is ready to be added to policies or attached to a service. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Sensitive Data Dictionary Group Options

Option	Description
Display Name	Type a name to identify the pattern. Once there is a match, this name is displayed in the Details pane of the Violations table.
Type	Select a type of pattern to match. Options include: Simple: Enables you to define the exact string to find. When selected for Sensitive Data Dictionaries, the **Elements without Masking** field is disabled. Advanced: Enables you to specifically define on what part of the string to perform the search and can use regular expressions to define what to look for. When selected for Sensitive Data Dictionaries, the **Elements without Masking** field is enabled.
Pattern	When Simple is selected type: Type the exact string pattern you want to find. When Advanced is selected type: Type a pattern to determine exactly where to look for sensitive data. Patterns use syntax similar to **Signatures** syntax, which contains a part and regular expressions. The main difference between the Signatures and Dictionary groups is that signatures perform search on all the traffic, while dictionary groups search the predefined location only.
Elements without Masking	Configure partial masking if desired by defining elements that do not need to be masked. For example, last 4 digits of the credit card: xxxx-xxxx-xxxx-8211. Partial masking helps to identify the victim of the attempt to steal sensitive data, as it shows a part of the card number. This option is only available when configuring Sensitive Data Dictionary Groups. Configuring **Elements without Masking** is only available when using **Advanced type**. Simple data is always completely masked, except for the following separators: - , . _ / \. To see an example of masking, see Configuring Data Masking Options.

EXAMPLE: PREVENTING DATA LEAKAGE

This example describes how to configure a sensitive data dictionary to prevent the leakage of sensitive phone numbers in database responses. It results in masking this data in alerts, audits and logs.

To configure Sensitive Data Protection:

1. In the **Main** workspace, click Setup > Global Objects. The Global Objects window appears.
2. In the **Objects** dropdown list, select **Sensitive Data Dictionary Groups**.
3. In the Globals Tree pane, right-click **Sensitive** Dictionary Group, then click Create Sensitive Dictionary Group. The Create Sensitive Dictionary roup dialog box appears.
4. In the Name text box, type **Sensitive Phonenumbers**.
5. Click **Create**. The group **Sensitive Phonenumbers** is added to the list in the **Globals Tree**, and its details are displayed in the **Details** pane.
6. In the **Details** pane, click  twice. Two new rows are added to the bottom of the Sensitive Dictionary table.
7. In the first row, configure the following:
   - Under display name, type John Smith.
   - Select the type **Simple**.
   - Under pattern, type in 312.323.9876
8. In the second row, configure the following:
   - Under display name, type Jane Smith.
   - Select the type **Simple**.

- Under pattern, type in 312.323.9882

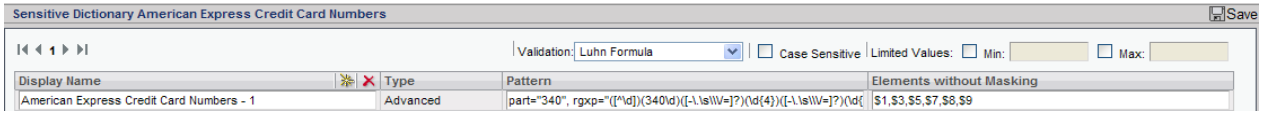
9. From the top of the window, click **Save**.
10. Configure the relevant policy as follows:

- To prevent data leakage, see Web Application Custom Policies on page 210: Example - Preventing Data Leakage in Web Responses.

EXAMPLE: CONFIGURING SENSITIVE DATA DICTIONARY GROUP MASKING

The following illustrates the configuration of a Sensitive Data Dictionary Group that is used to protect American Express Credit Card Numbers. The pattern Type is Advanced. The Pattern contains the part and the regular expression that together define what to look for:

part="340", rgxp="([^\d])(340\d)([-\.\s\\\/=]?)(\d{4})([-\.\s\\\/=]?)(\d{3})(\d)([-\.\s\\\/=]?)(\d{3}[^\d]{1})"



The pattern of any credit card number is usually the following: /xxxx-xxxx-xxxx-xxx/. Elements without Masking present which parts of the credit card number is eventually shown in Alerts, as follows:

/	xxxx	-	xxxx	-	xxxx	-	2415	/
$1	$2	$3	$4	$5	$6	$7	$8	$9
unmasked	masked	unmasked	masked	unmasked	masked	unmasked	unmasked	unmasked

Importing Data from a CSV File

When creating Dictionary Groups, you can import generic or sensitive data into the group you are creating. The data is uploaded from a CSV file and it overrides the existing table entries. The maximum number of entries in the CSV file that you can upload is 50000.



Note: The pattern used in such dictionary groups is always Simple.

To upload a CSV file to a Generic/Sensitive Dictionary Group:

1. Create a Generic or Sensitive Data Dictionary Group as described in Creating New Dictionary Groups on page 351.
2. In the Globals Tree pane, right-click the group that you have defined and click Upload File. The Upload File dialog box appears.
3. Type the location of the CSV file that you want to upload, or click Browse and navigate to the file location.
4. To ignore the table title when uploading tables from a CSV file, select **The File Includes Title**, which means that SecureSphere ignores the first row in the file.
5. Click Upload. The Upload File dialog box closes and the table entries from the CSV file appear in your dictionary group.

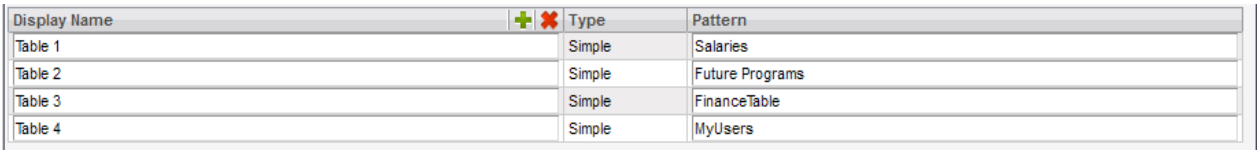
EXAMPLE: CSV DATA IMPORTED INTO SECURESPHERE

The format of data that appears in the CSV file is the pattern name followed by the pattern to match.

For example, the following CSV file data:

```
Table1,Sallaries
Table2,FuturePrograms
Table3,FinanceTable
Table4,MyUsers
```

Is presented in SecureSphere as follows:



Display Name	Type	Pattern
Table 1	Simple	Salaries
Table 2	Simple	Future Programs
Table 3	Simple	FinanceTable
Table 4	Simple	MyUsers

Meaning that from each line the information before the first comma is identified as the pattern's name and the rest of the line as the pattern.

Masking Data in Elements

Data Masking groups enable you to define lists of specific parameters you have determined to be sensitive, then attach this list to a service. SecureSphere then uses this list to identify these parameters on the configured service, and mask their value to prevent unauthorized personnel from seeing this data. For example, you could create a list and add the parameter name password, then attach this list to a service. Then any time one of these parameters is detected on this service, its value is masked. This is done using a data masking list in the global object named **Data Masking Groups**. The following table lists the main tasks that can be conducted to configure masking of elements.

Configuring Masking of Elements Tasks Overview

	Action	Description	For more information, see...
1	**Define a new Data Masking List (optional).**	Define a new Data Masking List that includes sensitive parameters you want to Mask in alerts, logs and reports. Alternatively, you can add parameters to an existing Data Masking Group.	Creating New Global Objects on page 346.
2	**Add parameters to the Data Masking List.**	Add parameters for SecureSphere to use to identify sensitive information that should be masked.	Adding Parameters to a Data Masking List on page 356.
3	**Attach a Data Masking list to a service.**	Attach a Data Masking list to a service so all matching parameters detected in traffic hosted by that service are masked.	Attaching a Data Masking List to a Service on page 357.

Adding Parameters to a Data Masking List

Data Masking Lists are lists of parameters that may appear in parameters, headers, or cookies, that are sensitive, and that are subsequently masked in alerts and logs by replacing their values with a series of asterisk. You can add parameters to an existing data masking list, or create a new data masking list.

To add parameters to an existing data masking list:

1. In the **Main** workspace, click Setup > Global Objects. The Global Objects window appears.
2. In the **Objects** dropdown list, select **Data Masking Group**. Existing **Data Masking Group** are displayed.
3. In the **Globals Tree**, select an existing **Data Masking Group** or create a new group as described in Creating New Global Objects on page 346. The selected **Data Masking Group** appears in the Configuration pane on the right-hand side of the screen.

4. From the top of the **Details** pane, click  **New**. A new row is added to the bottom of the **Parameter Table** displayed in the **Details** pane.
5. Type the name of the parameter as it appears in context.
6. Configure the following:
   - **Enable**: New parameters are enabled by default. To disable them deselect the enable checkbox.
   - **Match**: From the dropdown menu, select whether you want to match the full parameter, or if matching a prefix is sufficient to identify it for masking.
   - **Description**: Enables you to type a free text description of the parameter.
   - **Parameters**: Determines if this to mask this parameter if it appears in regular traffic.
   - **Headers**: Determines whether to mask this parameter if it appears in a packet header.
   - **Cookies**: Determines whether to mask this parameter if it appears in a cookie.
7. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Attaching a Data Masking List to a Service

Once a Data Masking List has been configured, it needs to be attached to the service to which you want to apply it.

To attach a Data Masking List to a service:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. Click the **Service** to which you want to associate the item. Service details appear in the **Site** window.
3. Click the **Operation** tab. Operation options are displayed.
4. Attach the required item, as follows:
   - **To attach a Sensitive Data Dictionary Group**: In the **Sensitive Dictionaries** pane, click **New**, then select the desired dictionary from the drop down list.
   - **To attach a Data Masking List**: Under **Mask Data in Elements**, select the list from the drop-down menu.
5. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Working with Data Types

Data types in SecureSphere are used to classify database data. Data types assigned to database data is further used as part of Risk Management that enables you to identify data in your network by category. This provides an extra layer of information regarding your data beyond the sensitive non-sensitive model and allows you to easily identify the specific types of information in your network.

SecureSphere comes with a number of built-in data types defined by Imperva ADC including Account Numbers, Bank Identifier Code, Drug Name, Payment Card CVV, and more. Database items for these types contain built in rules to identify table and column names and actual content, while you can manually configure rules for file type classification using patterns and wildcards, additionally assigning an owner name and owner department. These types are then used to classify database data when discovered, assist in filtering data in SecureSphere, and can be included in reports.

The following options enable you to create and work with data types in SecureSphere:

Creating a New Data Type 358

Creating a New Data Type

This procedure describes how to create a new data type.



Note: While this procedure describes how to create new data types from the Global Objects window, you can also create new data types and edit them directly from a data classification scan as described in Working with SecureSphere Scans on page 32.

To create new data types:

1. In the **Main** workspace, click Setup > Global Objects. The Global Objects window appears.
2. In the **Objects** dropdown list, select **Data Type List**. Existing **Data Types** are displayed.
3. In the **Globals Tree**, click  **New**. The create New Data Type dialog box appears.
4. Type a **Name** for the **Data Type**, then click **Create**. The new data type is added to the **Data Type List**. The data type by default is marked as sensitive. If you'd like to change this, deselect the sensitive check box.
5. Configure the Data Type as follows
   - For **DB Data Classification**, configure options as described in Configuring Database Data Types.
   - For **File Data Classification**, configure options as described in Configuring File Data Types.
6. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Configuring Windows Domains

In order to classify, monitor and manage writes for file and directories on file servers, the domain for these files servers must be configured. Domain credentials, Active Directory/LDAP, and user business information can be configured for each domain.



Note: You can configure SecureSphere to work with multiple windows domains.

They include the following:

Configuring Windows Domain Options

	Action	Description	For more information, see...

	Action	Description	For more information, see...
1	**Default Domain**	**Used for Classification, File Mapping and File URM.** Configures the default domain.	Configuring the Default Domain Name on page 359
2	**Default Domain Credentials**	**Used for Classification, File Mapping and File URM.** Configures a set of default credentials for exploring and classifying files and directories and for extracting file privileges for User Rights Management.	Configuring Default Domain Credentials on page 360
3	**Active Directory Configuration**	**Used for adding user organizational information.** Configures an external system that interfaces with LDAP/Active Directory for extracting the users' groups for User Rights Management.	Configuring Active Directory Integration for Data Enrichment on page 360
4	**User Organizational Information for Monitoring**	**Used for adding user organizational information.** Configures information used to determine users' business information like department, title, security groups and full name.	Configuring Active Directory Integration for Data Enrichment on page 360

Configuring the Default Domain Name

The default domain determines the domain used by SecureSphere by default when looking to extract Active Directory/LDAP details used to identify users, departments and security groups, regarding traffic taking place in your network. Once configured, the default domain it is used by default for each new file server created in the system without the need to reconfigure it.

To configure a default domain:

1. In the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Objects** dropdown list, select **Windows Domain**. Existing **Windows Domains** are displayed.
3. In the **Globals Tree**, click  **New.** The **Create Windows Domain** dialog box opens.
4. Type a name for the Windows Domain, then click **Create**. The new domain is created and its details are displayed in the right-hand side of the window.
5. At the top of the window under **Domain**, type the name of the domain you want to use as the default in SecureSphere.
6. Click **Save**. The settings are saved and the domain credentials have been configured. For information on configuring default domain credentials, see Configuring Default Domain Credentials on page 360.

Configuring Default Domain Credentials

Domain credentials are the credentials you require in order to access your company's Windows domain. This procedure is only required when working in a Windows domain.

To configure a domain credentials:

1. In the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Objects** dropdown list, select **Windows Domain**. Existing **Windows Domains** are displayed.
3. In the **Globals Tree**, click  **New.** The **Create Windows Domain** dialog box opens.
4. Type a name for the Windows Domain, then click **Create**. The new domain is created and its details are displayed in the right-hand side of the window.
5. Under **Default Domain Credentials**, type the following:
   - **Domain**: Type your LDAP/Active Directory domain name prefix.
   - **User**: Type the username needed to access the domain.
   - **Password**: Type the password that matches the username.
   - **Verify Password**: Retype the password.
6. Click **Save**. The settings are saved and the domain credentials have been configured.

Configuring Active Directory Integration for Data Enrichment

In many organizations Active Directory is the main source for managing Users and Security Groups. Active Directory configuration enables you to interface with an LDAP/Active Directory external system and import its data to enrich file data by identifying the actual users and their security groups that access data, and displaying this data in view, reports, auditing and more. This data is used across File Security and SharePoint Security products. For example, when configuring policies' match criteria or defining URM, the data imported from the Active Directory enables auto-completion of user group name and user department name.

When Active Directory data is fetched into SecureSphere, there is a possibility to add a CSV file that contains additional information about Active Directory users. So for each Active Directory user, you can combine in SecureSphere the following information and use it for data enrichment:

- Active Directory data.
- CSV data that is uploaded to SecureSphere using data sources different from Active Directory.

You can also configure integration with Active Directory for managing the users, see Configuring Integration with Active Directory for User Management on page 403.

To configure an LDAP/Active Directory external system:

1. In the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Objects** dropdown list, select **Windows Domain**. Existing **Windows Domains** are displayed.
3. In the **Globals Tree**, click  **New.** The **Create Windows Domain** dialog box opens.
4. Type a name for the **Windows Domain**, then click **Create**. The new domain is created and its details are displayed in the right-hand side of the window.
5. In the **Active Directory Configuration** section, either select an Active Directory external system previously configured in the Admin workspace, or click **Edit** to create a new one. Active Directory details are displayed.



Note: This procedure describes how to configure Active Directory. For information on creating and configuring other External Systems including Active Directory Collection (Forest), LDAP and Radius Authentication, and SQL, see the Admin guide.

6. Type the following:
   - **Name**: Name you want to assign the external system.



Note: When configuring the domain name for LDAP integration, the extension .local should be removed.

- **Primary Server:** IP Address of the Primary Active Directory server.
- **Secondary Server (optional):** IP Address of the Secondary Active Directory server.
- **Use SSL**: Determines whether or not to use SSL to connect to LDAP.
- **Server Port**: Port number used by the Active Directory server.
- **Domain\Account Name**: Account Name\DN
- **Password**: Password used to connect to Active Directory server.
- **Verify Password**: Password used to connect to Active Directory server
- **Base DN**: Base DN of your organization's Active Directory.
- **Follow Referrals**: Select this option to tell/direct the LDAP system to follow referrals.

7. In the **LDAP Custom Fields** section, in the **Custom Field** text box, type the field name as it appears in Active Directory, to receive data from a specific Active Directory field.
8. Click **Save**. The settings are saved and the domain credentials have been configured.
9. Click **Test Connection** to verify the connection is properly configured.
10. Close the **Edit Active Directory** window.
11. To receive data from Active Directory, click **Synchronize Now**.
12. To schedule automatic synch, select **Once** or **Recurring** and perform settings as explained in Configuring Scheduling on page 413.
13. To receive information about the Centrify users in Active Directory, check **Centrify in use**.
14. In the **User Data Enrichment** section, use the following options:
    - **Upload**: A CSV file with the additional information about Active Directory users is sent to SecureSphere and SecureSphere creates a table with this information. The information in this table is combined with the data fetched from Active Directory and then used for data enrichment.
    - **Clear**: Removes from SecureSphere the values that were uploaded from the CSV file, so that the next time you the Active Directory data is fetched, these values do not appear.
    - **Export**: Restores back the latest CSV file information in case you need the data that was deleted using the **Clear** option and you are not interested in receiving a new CSV file.
15. Click **Save**.

Generic Dictionary Groups

Generic Dictionary groups in SecureSphere are groups of data patterns that can be matched to items such as URLs, headers or other strings and enable you to identify and control suspicious behavior that may be posing a threat to your data or network. Then block this traffic, generate alerts and violations, and mask sensitive data in violations. Generic Dictionaries are similar to signatures in SecureSphere, only more flexible.

Generic dictionaries provide two primary benefits that include:

- **Reducing processing overhead**: Generic dictionaries search for suspicious strings only under specific conditions and in specific locations (such as an html page). This is different than using signatures which search all traffic. Generic dictionaries only search after all other mechanisms (for example, signatures), have already been run. This makes operation effective and accurate, and saves time and conserves system resources.
- **Enabling the use of AND operations**: While most criteria in SecureSphere policies can only use a single instance in a policy and enable the use of OR operations with criteria, generic dictionaries can use multiple instances. This enables you to conduct AND operations on two or more criteria defined in a custom policy, providing more advanced detection options.



Note: Using generic dictionaries on all responses could significantly impact SecureSphere performance. It is therefore recommended that you configure a generic dictionary to match under certain conditions and define a new custom policy rule with additional conditions. For example, use a custom rule to set the sensitive data generic dictionary to run only when the profile violation "Parameter Value Length Violation" or "Parameter Type Violation" occur.

The following table lists the main tasks that can be conducted to configure generic dictionary groups.

Configuring Generic Dictionaries Task Overview

	Action	Description	For more information, see...
1	**Define a new generic dictionary group (optional).**	Define a new Generic Dictionary group that includes malicious patterns you want to search for. Alternatively, you can use an existing dictionary group.	Creating New Dictionary Groups on page 351.
2	**Add data patterns to dictionary group.**	Add data patterns to the Generic Dictionary group used to identify patterns.	Adding Data Pattern Definitions and Validation Method to Dictionary Groups on page 352.
3	**Attach dictionary group to security policy.**	Attach Generic Dictionary group to a security policy and configure match criteria that determine when the policy is invoked.	Attaching a Data Masking List to a Service on page 357



Note: You can view or change only the dictionaries for which you have Edit permissions.

EXAMPLE: PREVENTING DIRECT ACCESS TO A WEBPAGE

In this example you have a webpage or area in your site which you only want users to be able to reach from within your domain. Users can currently access this webpage either though a link found in search engines or by typing in the exact host and URL. You are going to create a generic dictionary group to identify the URL from which users are permitted to access the target page, then create a policy that defines the desired behavior. This is done by creating a single generic dictionary and using it in a policy. By using a generic dictionary instead of a signature you significantly reduce SecureSphere processing overhead in two ways. First, by only checking for the signature if users are trying to access a specific page, and second, only having SecureSphere search for this signature after traffic has been filtered by all other protective mechanisms.

To configure a generic dictionary group to prevent direct access:

1. Create a new generic dictionary global object named **wrongReferer**, with the following parameters.
   - **Display Name**: Referred.
   - **Type**: Advanced.
   - **Pattern**: `part="referer:", rgxp="referer:\s*(?!www.mysite.com)"`
2. Create a new Web Application Custom Policy named **wrongReferer**.
3. Add the criteria **Generic Dictionary Search** and configure it with the dictionary **wrongReferer** created in step 1 above.
4. Add the criteria **URL Prefix** and type the value **/forums/**. This represents the area that will be affected. Meaning, you cannot enter a URL inside this area unless coming from the specified domain.
5. Click **Save**. The policy has been configured. It will catch every packet that has a referer that is not www.mysite.com along with the URL prefix. This test is only conducted on URLs starting with "/forums/."



Note: Additionally, you can configure a redirect as the error page to direct users to the company homepage. This might be useful if for example, a user arrived at the page in question through a link from a search engine.

EXAMPLE: VALIDATING INPUT USING MULTIPLE CRITERIA AND THE AND FUNCTION

In this example you have a form on a web site that has a FROM address, a TO address, and a comment. The form is being abused to send spam. While the form contains browser based javascript used to count from 1000 to 0 (the number of characters entered into the comment), someone has been submitting information by bypassing the form and client side validation via a script or other means. This was detected when forms were received with comments that were not empty, yet their counter still read 1000.

To prevent this from happening, you want to write a policy that checks two separate fields and only alerts when both fields match the policy. While this cannot be accomplished using signatures which are limited to the OR function, it can be accomplished by creating two separate generic dictionaries and using them in a custom policy.

To configure a generic dictionary group to validate multiple criteria:

1. Create a new generic dictionary global object named **Text**, with the following parameters.
   - **Display Name**: Text.
   - **Type**: Advanced.
   - **Pattern**: `Part="Text", rgxp="Text=[^\n&]"`
2. Create a second generic dictionary global object named **characterCount**, with the following parameters.
   - **Display Name**: **characterCount**.
   - **Type**: Advanced.
   - **Pattern**: `Part="`**characterCount**`=1000"`
3. Create a new Web Application Custom Policy named **commentValidation**.
4. Add the criteria **Generic Dictionary Search** and configure it with the dictionary **Text** created in step 1 above.
5. Add a second instance of the criteria **Generic Dictionary Search** and configure it with the dictionary **counter** created in step 2 above.



Note: While most criteria in SecureSphere can only be used once in a policy, both Generic and Sensitive data dictionaries can be used in a policy multiple times.

6. Add the criteria **URL Prefix** and type the URL of the specific page that receives this input. For example, **/actionpages/sendForm.asp**.

Click **Save**. The policy has been configured. It will alert whenever an e-mail message is sent in which not empty but its counter still reads 1000. This test is only conducted on URLs starting with **/actionpages/sendForm.asp**.

Protecting and Auditing Custom Information with Lookup Data

Lookup data are customizable attributes that can be added to events taking place in your network to enrich data and make it more relevant to your working environment. First, you configure a lookup data set either manually or by importing information from external systems such as LDAP, Active Directory, or databases, add an attribute you would like to match with an existing data characteristic, then associate lookup data sets with security, auditing or enrichment policies which define data to protect or enrich event information. Lookup Data sets are used for the following activities:

- Adding business/organizational context to SecureSphere policies
- Enriching event data with business/organizational data
- Viewing, aggregating, and filtering data using business/organizational information

- Change management control
- Integrating with other security products and building transactional comprehension

For a list of predefined Lookup Data Sets, see Predefined Lookup Data Sets on page 366. The following table lists the main tasks that can be conducted to configure Lookup Data.

Configuring Lookup Data Task Overview

	Action	Description	For more information, see...
1	**Create a Lookup Data Set**	Create a new Lookup Data set.	Creating a Lookup Data Set on page 366
2	**Configure a Lookup Data Set**	Configure the Lookup Data set by adding attributes and keywords and defining relevance.	Configuring a Lookup Data Set on page 369
3	**Configure a Security or Audit Policy with Lookup Data Set match criteria (optional)**	Use Lookup Data match criteria to configure a custom security or audit policy to determine what traffic to monitor and protect.	Custom Policies Match Criteria on page 600 Security Policies on page 166 Configuring an Audit Policy
3	**Configure a Enrichment Policy (optional)**	Configure an enrichment policy that uses Lookup Data to enrich events by adding custom attributes.	Enriching Data on page 372

EXAMPLE: MONITORING SENSITIVE INFORMATION ACCESS BY DATABASE ADMINISTRATORS WITH MSSQL

Database Administrators play a crucial role in the smooth management of your data. Their credentials are the keys to the kingdom and they have access to almost all data. However while technically your DBAs should be able to work unhindered, their credentials do pose certain risks. Therefore you may want to monitor their access to sensitive information. You can do this by creating a Lookup Data group including system users and their roles, then configure a security policy to alert if they access this sensitive data.



Note: Implementing this example would require that a MSSQL external system would have been configured under Admin > System Definitions > External System.

To monitor sensitive data access by a DBA using a Lookup Data set:

1. From the **Main** workspace, select **Setup > Global Objects**.
2. In the **Scope Selection** dropdown list, select **Lookup Data**.
3. Select **From ADC Template**, type **DBAUsers** as the name, then select the **MSSQL Database Roles** option. A new Lookup Data Set is created.
4. Click the **Query** tab.
5. Select the **External System Name** of the External System you are using to create the Lookup Data Set.
6. Test the query to make sure it works, then test the **Mapping Query**, and finally click **Run Now**. The Dataset is populated with the **Login** and **DB Role**.
7. Create a new DB Service Custom policy.
8. Leave the Action and Followed Action blank.
9. Set Severity to Medium.
10. Add the **Data Set: Attribute Lookup** match criteria, and configure it as follows:

- **Field**: Choose **Database User Names**
- **Lookup Data Set**: Select **DBAUsers**, (the Lookup Data set you configured in step 4).
- **Attribute**: Select **Database Role**.
- **Operations**: Leave it set to **At least one**.
- **User Defined value**: Type DBA.

11. Add the **Sensitive Data Access** match criteria, and set it to **Sensitive**.
12. On the **Apply** tab, Apply it to the required servers.
13. Click **Save**. Settings have been saved. Now when a user who has been identified as a DBA accesses sensitive data, an alert will be generated.

EXAMPLE: MONITORING FINANCE DEPARTMENT INFORMATION WITH LDAP

Database tables managed by the Finance Department typically contain sensitive data regarding various aspects of your company's operation. This might include budgets, salaries and more. To meet regulatory compliance you may need to audit this information. Additionally, you may also want to block access to users that do not belong to the Finance Department. This example provides instructions on how to configure Lookup Data Set and use it to monitor access to Finance Department tables.

This example uses a pre-configured lookup data set that is located in the Lookup Data sets window.



Note: Implementing this example would require that a LDAP external system would have been configured under Admin > System Definitions > External System.

To configure data enrichment with LDAP data:

1. Create a **Table Group** named **FinanceDepartementTables** and include all database tables that belong to the **Finance Department**.



Note: You can also use a table group that was created by a discovery policy.

2. From the **Main** workspace, select **Setup > Global Objects**.
3. In the **Scope Selection** dropdown list, select **Lookup Data**.
4. Select the pre-defined **User Roles From Active Directory (no domain)** option.
5. Click the **Query** tab.
6. Select the **External System Name** of the LDAP External System you are using to create the Lookup Data Set.
7. Enter the **Search Base** you want to query. You may need to obtain this from your network administrator.
8. Test the query to make sure it works, then test the **Mapping Query**, and finally click **Run Now**. The Dataset is populated with the User, Group and Department information from LDAP.
9. Configuring scheduling for the Lookup Data set for once a week on Monday morning.
10. Create a new DB Service Custom policy.
11. Under **Action** choose block, under **Followed Action** choose **Terminate Session**.
12. Set **Severity** to **High**.
13. Add the **Data Set: Attribute Lookup** match criteria, and configure it as follows:
    - **Field**: Choose **Database User Names**
    - **Lookup Data Set**: Select **User Roles From Active Directory (no domain)**, (from step 4).
    - **Attribute**: Select **Department**.
    - **Operations**: Set to **Exclude All**.
    - **User Defined value**: Type **Finance**.
14. Add the **Table Groups** match criteria, and add the **FinanceDepartementTables** table group you created for the Finance department that includes all Finance Department tables.

15. On the **Apply** tab, Apply it to the required services.

Click **Save**. Settings have been saved. Now when a user who has been identified as a DBA accesses sensitive data, an alert will be generated.

Predefined Lookup Data Sets

SecureSphere comes with a number of predefined Lookup Data sets. These are either used by various SecureSphere features, or can be used to query data. The following table lists the predefined lookup data sets in SecureSphere.

Predefined Lookup Data Sets

Name	Description
Default Database Account Mapping	Maps the account name as it is recognized in the database to the organizational/domain user name such as that defined in Active Directory. This Lookup Data Set is used as part of User Rights Management. For more information see Working with User Rights Scans.
IP Addresses of Known Anonymous Proxies	Lists known anonymous proxies which can often be used to hide malicious activity. This data set is automatically configured as part of the Threat Radar and automatically updated by Imperva ADC.
Malicious IP Addresses	Lists IP address that are known to be launching points for malicious activity. This data set is automatically configured as part of the Threat Radar and automatically updated by Imperva ADC.
User Roles - From Active Directory (no domain)	Retrieves Users, Departments and Groups from non-domain LDAP and uses them to populate a Lookup Data set. Extracts user name from CN string to make it user friendly in alerts and reports. Needs to be configured with an External System on the Query Tab before being run.
User Roles - From Active Directory (with domain)	Requires configuration. Retrieves Users, Departments and Groups from domain based LDAP and uses them to populate a Lookup Data set. Extracts user name from CN string to make it user friendly in alerts and reports. Needs to be configured with an External System on the Query Tab before being run.

Creating a Lookup Data Set

The first step in using lookup data is the creation of a lookup data set. A lookup data set is a set of data that contains a key, to which additional attributes are associated. These attributes can either be imported from external systems, or manually entered. SecureSphere provides you a number of means to create lookup data set.



Note: Instead of creating a new Lookup Data Set, a number of pre-defined Lookup Data Sets are provided with SecureSphere that you can use. For more information see Protecting and Auditing Custom Information with Lookup Data on page 363.

- **Manually**: Enables you to manually create a Lookup Data set. For more information, see Manually Creating a Lookup Data Set on page 367.
- **From a CSV File**: Creates a Lookup Data set based on the contents of a comma separated values file. For more information see Creating a Lookup Data Set from a CSV File on page 367.

- **From ThreatRadar**: Creates Lookup Data sets that are used in ThreatRadar services. For more information see ThreatRadar Fraud Prevention Services on page 206
- **From an External System**: Creates a Lookup Data set by connecting to a database or external system then importing the relevant data. For more information see Creating a Lookup Data Set from an External System on page 368.
- **From ADC Template**: Creates a Lookup Data set using templates based on standard database layouts and LDAP information, created and maintained by SecureSphere ADC. For more information, see Creating a Lookup Data Set from a Template on page 368.

Manually Creating a Lookup Data Set

Manually creating a new lookup data set can be useful for tracking events based on non-system information. For example, perhaps you want to track users on vacation to verify no activity is taking place under their name. You'd manually create a lookup data set (or import from file), then configure a security policy to alert when a user on this list generates an event.

To manually create a lookup data set:

1. From the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Scope Selection** dropdown list, select **Lookup Data**. Existing **Lookup Data sets** are displayed.
3. In the **Lookup Data Sets** title bar, click  **New**, then click **Create Data Set**. The **Create Data Set** dialog box appears.
4. Type a **Name** for the Data Set.
5. Select the type of Lookup Data set to create by choosing one of the following:
   - **From Scratch**: Enables you to create a lookup data set from scratch.
   - **Use Existing**: Enables you to create a new lookup data set based on an existing one.
6. Type a **Key Column** name. The key column is the column used to identify the data items. For example, you may want to use user names as your key, then create additional columns to add the Role and Region for each user in the list.
7. Add additional columns by clicking  **New** for every column you want to add. Additional columns are added. These represent the additional categories in addition to the source (key) column you may want to apply. For example, in addition to a User name, you may want to add a Role and Manager column, to populate with user roles and managers.
8. Type a **Name** for each of the additional columns.
9. Select the **Case Sensitive** option if you'd like Key Column data to be case sensitive when used as a criteria in a policy or for filtering.
10. Click **Create**. The new **Lookup Data Set** is created and added to **Lookup Data** category in the **Global Objects** window.
11. Configure the Lookup Data set as described in Configuring a Lookup Data Set on page 369.

Creating a Lookup Data Set from a CSV File

Creating a lookup data set from a CSV file can assist you in preparing data before creating a data set, then importing the values from the CSV file to automatically populate the data set. You need to prepare a CSV file before import. It should be organized in rows.

To create a lookup data set by importing a CSV:

1. From the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Scope Selection** dropdown list, select **Lookup Data**. Existing **Lookup Data** sets are displayed.
3. In the **Lookup Data Sets** title bar, click  **New**, then click **Upload CSV**. The **Import from CSV** dialog box appears.

4. Type a **Name** for the Lookup Data Set.
5. Browse to the location of the CSV file and select it.
6. If you'd like to use the first row as the category (column) names, then enable the **Use First Line as Header** option.
7. Click **Upload**. The file is imported and a new Lookup Data set is created. It now appears in the **Lookup Data Sets** category in the **Global Objects** window under the name you assigned it in step 4.
8. Further configure and customize the Lookup Data set as described in Configuring a Lookup Data Set on page 369.

Creating a Lookup Data Set from an External System

SecureSphere provides you the ability to import data from an external system such as SQL or LDAP. This provides an easy and accurate mechanism for creating Lookup Data sets from existing data, which can be queried and used to populate the Lookup Data set.



Note: In order to import data from an external system. That external system must first be configured in SecureSphere administration. For more information see the SecureSphere administration guide or speak to your SecureSphere administrator.

To create a lookup data set from an external system:

1. From the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Scope Selection** dropdown list, select **Lookup Data**. Existing **Lookup Data** sets are displayed.
3. In the **Lookup Data Sets** title bar, click  **New**, then click **From External System**. The **Create From External System** dialog box appears.
4. Type a **Name** for the Lookup Data set.
5. Select the **Case Sensitive** option if you'd like the key column data to be case sensitive when used as a criteria in a policy or for filtering.
6. Select an **External System Type** from the dropdown list.
7. Select an **External Stem Name** from the dropdown list. Only items that have been previously configured by a SecureSphere administrator are available. See your SecureSphere admin for details.
8. Configure a **Query** and other options as described in Using a Query to Populate a Lookup Data Set on page 371.
9. Once the query has been created and Query Attributes table has been populated as desired, click **Create**. SecureSphere creates a new Lookup Data set which appears in the list of Lookup Data sets in the **Global Objects** window.
10. Further configure the new Lookup Data set as described in Configuring a Lookup Data Set on page 369.

Creating a Lookup Data Set from a Template

In order to help you easily build Lookup Data sets based on particular use cases, SecureSphere comes with a set of Lookup Data set templates that are configured with queries to import a specific subsets of data such as users and roles, then use this information to configure policies. These templates have been configured by SecureSphere ADC to contain customized queries related to specific information commonly used.

To create a lookup data set from a template:

1. From the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Scope Selection** dropdown list, select **Lookup Data**. Existing **Lookup Data sets** are displayed.
3. In the **Lookup Sets** title bar, click  **New**, then click **Create from Template**. The **Create From Template** dialog box appears.
4. Type a **Name** for the **Lookup Data** set.
5. From the **Template** dropdown menu, select the desired template. Optional templates include the items in the table below.

6. Click **Create**. The lookup data set is created with a predefined query to match the required system. You now need to run the query to populate the data set as described in Using a Query to Populate a Lookup Data Set on page 371.

Lookup Dataset Templates

Option	Description
User Roles from Active Directory (no domain)	Retrieves Users, Departments and Groups from non-domain LDAP and uses them to populate a Lookup Data set. Extracts user name from CN string to make it user friendly in alerts and reports.
User Roles from Active Directory (with domain)	Retrieves Users, Departments and Groups from domain based LDAP and uses them to populate a Lookup Data set. Extracts user name from CN string to make it user friendly in alerts and reports.

Configuring a Lookup Data Set

Lookup Data sets are comprised of a number of options that assist in populating their contents from a external systems and defining customizing attributes. This procedure describes how to configure a Lookup Data set.



Note: Both IPv4 and IPv6 addresses can be used in lookup datasets. However, IPv6 addresses are used exactly as they are entered. They are not converted to the long form and SecureSphere will fail to match two identical IPv6 addresses if they are differently expressed: for example, SecureSphere will not recognize that **2001:db8::1:0:0:1** and **2001:db8:0:0:1:0:0:1** are the same.

To configure lookup data sets:

1. From the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Scope Selection** dropdown list, select **Lookup Data**. Existing **Lookup Data** sets are displayed.
3. Select the Lookup Data set you want to configure. That Lookup Data set's details are displayed in the **Details** pane.
4. Configure the Lookup Data set as shown in the table below.
5. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Lookup Dataset Options

Tab	Description
Data (columns)	Add new columns and edit existing columns as required. For more information, see Managing Columns in Lookup Data Sets on page 370.
Keywords and Relevance	Add keywords and assign the Lookup Data set relevance used to assist in later finding them. For more information, see Configuring Lookup Data Set Keywords and Relevance on page 370.
Query	Use the query feature to import data from an external system such as SQL or LDAP and use it to populate the Lookup Data set. For more information, see Using a Query to Populate a Lookup Data Set on page 371.
Scheduling	Configure scheduling to determine when to update the Data Set from its source external system. This guarantees that changes such as new users and other items are reflected in the data set. Scheduling Lookup Data updating is done from the Scheduling tab using standard scheduling functionality as described in Configuring Scheduling on page 413.
Permission	Assign permissions that determine who can access and modify the data set. For more information see Configuring Lookup Data Set Permissions on page 372.

Managing Columns in Lookup Data Sets

Lookup Data sets are made up of columns. The first column represents the Key Value, or that by which all other values are identified. Each additional column represents an attribute for the key. You can configure a Lookup Data set with up to 19 attributes (add up to 19 columns). This enables you to accommodate a wealth of additional information and enrich data generated by events, making it more intuitive and consistent with the specific terminology used by your organization. When creating a lookup data set from an external system, columns will automatically be populated from the external system you use.



Note: Use the search tool to search for a value anywhere in the Lookup Data set. Search is case sensitive.

To manually add and edit columns in a lookup data set:

1. From the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Scope Selection** dropdown list, select **Lookup Data**. Existing **Lookup Data sets** are displayed.
3. Select an existing Lookup Data set. That Lookup Data set's details are displayed in the **Details** pane.
4. Click the **Data** tab. Data options are displayed for the selected Lookup Data set. You can perform the following actions:
   - **Add a new Record**: To add a new record (in the form of a row), click **New** in the header of the Lookup Data table. Then click a cell in the row and type the value.
   - **Edit an existing value**: Click the item that you want to edit.
   - **Delete a row**: Click the row once, then click **Delete**.
   - **Edit columns**: To manage columns in the Lookup Data set, click **Edit Columns**. The **Edit Columns** dialog box appears and enables you to add and remove columns, change the key column name and more.
5. When you are completed with your changes, Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Configuring Lookup Data Set Keywords and Relevance

Lookup Data sets can be assigned SecureSphere keywords. Keywords assist in associating data with relevant categories such as PCI, SOX, HIPPA, and more. Data sets are also assigned a relevance. Relevance categorizes lookup data sets making it easier to find them. Both keywords and relevance assist you when configuring filters, policies and reports using lookup data.

To configure keywords and relevance:

1. From the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Scope Selection** dropdown list, select **Lookup Data**. Existing **Lookup Data sets** are displayed.
3. Select an existing **Lookup Data** set. That **Lookup Data** set's details are displayed in the **Details** pane.
4. Click the **Keywords and Relevance** tab. Keywords and relevance options are displayed for the selected Lookup Data set. Associate the following items, as desired:
   - **ADC Keywords**: Enables you to associate keywords defined by Imperva ADC with the selected Lookup Data set.
   - **User (Defined) Keywords**: Enables you to associate customized keywords created by the SecureSphere administrator with the selected Lookup Data set.
   - **Relevance**: Assigns values in the Lookup Data set a relevance for fields, which is then used to help select the data set and its values in match criteria when configuring filters, policies and reports.
5. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Using a Query to Populate a Lookup Data Set

SecureSphere enables you to interface with external systems such as SQL and LDAP servers and use this information in order to help you build a Lookup Data set that matches fields found in these databases.



Note: In order to import data from an external system, it must first be configured in SecureSphere administration. For more information see the SecureSphere administration guide or speak to your SecureSphere administrator.

To configure a lookup data set from an external system:

1. From the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Scope Selection** dropdown list, select **Lookup Data**. Existing **Lookup Data sets** are displayed.
3. Select an existing **Lookup Data** set. That **Lookup Data** set's details are displayed in the **Details** pane.
4. Click the **Query** tab. Query options are displayed.
5. Select an **External System Type** from the dropdown list.
6. Select an **External System Name** from the dropdown list. Only items that have been previously configured by a SecureSphere administrator are available.
7. Configure a **Query** for extracting the desired data, or if creating from an ADC template, leave this field as is and go to step 8. This determines the cross section of data from which you can choose items to include in the data set. The query window is context sensitive. Options available for configuring a query depend on which External System type is selected, as shown in the table futher below
8. Once a query has been configured, click **Test Query** to run the query and display its results in the **Sample Results** pane. This enables you to see the available columns and select those you want to add to the data set.
9. Repeat steps 8 and 9 to drill down on the cross section of data relevant for your purposes. You may need to do this a number of times until you find the information that best suits your needs.
10. In the **Query Attribute** pane, click  **New** for every category you want to use in the Data Set from the external system. For example, if you want to import the Role, Region and Department columns from the external system, you would add three rows.
11. Copy the column name from the query to the Query Attribute column of the Query Attribute pane.
12. Configure an optional **Pattern** and **Substitution** for each **Data Set Column** to extract specific information from a value and use it as the attribute, as follows:
    - **Pattern**: Defines a pattern used to search for specific values that may be in a column.
    - **Substitution**: Replaces the information from the pattern with a desired format of information.

    Note: When a pattern and substitution is configured, SecureSphere only substitutes those items that match the pattern. If a value does not exactly match the value, it is used as-is.
13. Click **Test Mapping Query** to generate a sample of the data set you configured. The results are displayed in the **Sample Mapped Results** pane.
14. Once you drill down on the results you want to use for the Lookup Data set, click **Run Now**. SecureSphere populates the Lookup Data set with the selected information.
15. Further configure the new Lookup Data set as described in Configuring a Lookup Data Set on page 369

Query Options

Option	Description
LDAP	Enables you to configure options for querying your local LDAP system. Due to the advanced nature of configuration, it is recommended that you consult with your network manager to determine the required parameters for your query. Configuration options include the following: **Search Base**: Defines the location in the directory from which the LDAP search begins. The search base is added to the DN (if it was configured as part of the external system). **Search Scope**: Defines how far to search within the search base. • **One level**: Searches objects immediately within the base object, not including the base object itself. • **Subtree**: Search base object and entire subtree, of which the base object distinguished name is the topmost object. **Filter**: Filters down to include only those items in the subtree you want to utilize.

Configuring Lookup Data Set Permissions

Lookup Data set permissions determine which users can view the content of user data sets and which data sets are available for use with policies. This prevents unauthorized users from seeing the content of a user data set and subsequently exposing sensitive data to unauthorized personnel.

To configure permission to a lookup data set:

1. From the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Scope Selection** dropdown list, select **Lookup Data**. Existing **Lookup Data sets** are displayed.
3. Select an existing **Lookup Data** set. That **Lookup Data** set's details are displayed in the **Details** pane.
4. Click the **Permissions** tab. Permission options are displayed for the selected Lookup Data set.
5. Select the category which contains the users you want to assign permissions. Options include Users and User Groups. The selected category becomes active.
6. Add the desired users to the **Users with View Permissions** pane by selecting them and clicking the green arrow.
7. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Enriching Data

Enriching data in SecureSphere is the process of assigning customized attributes to events taking place in the network. This assists in adding information that matches the context of your company environment, and is then used to configure policies, audit data, and generate reports based on this added information. For example, you can add a user's department, then set controls which determine that users from the R&D department accessing sensitive data should be tracked.

You can enrich events by adding information from two different sources:

- **External sources**: Such as an LDAP server, external ticketing systems or even manually entered data. For an example of configuring an enrichment policy using an external source, see the example Enrich Audit Data with Department Information from LDAP located in Configuring Data Enrichment from a Lookup Data Set on page 377.
- **Another event**: This involves extracting information from or using an attribute of another event. Options include:
- **Web event**: Enables you to extract information from a field in HTTP traffic such as a source IP address, URL or cookie name, then enrich events with this information. For an example of configuring enrichment for web traffic, see the example **Configuring New Web User Tracking** below.

Note: DB enrichment is used to enrich queries that always remain the same. For example, queries generated by applications such as SAP, Oracle EBS, PeopleSoft, or built-in applications. Dynamic queries, such as those that use a real username that constantly changes and not an application user (and regularly change the syntax of the query), or where parameters dynamically change order are not suitable for enrichment.

A number of steps are involved in configuring SecureSphere to work with Data Enrichment. The below table lists the main tasks that can be conducted to configure Enrichment.

Note: When a connection is blocked by SecureSphere, enrichment data for the event that caused the action is not available.

Configuring Enrichment Policy Task Overview

	Action	Description	For more information, see...
1	**Create and Configure a Lookup Data Set (optional).**	Create and configure a lookup data set with customized attributes you want to apply to event data.	Creating a Lookup Data Set on page 366
2	**Create an Enrichment Policy**	Create an Enrichment policy.	Creating an Enrichment Policy on page 374
3	**Configure an Enrichment Policy**	Configure the Enrichment Policy to determine what data to enrich.	Configuring Data Enrichment Policies on page 375 Working with Audit Policies

EXAMPLE: CONFIGURING NEW WEB USER TRACKING

In this example we want to prevent users from purchasing an item from an e-commerce website if an account's shipping address has changed in the past twelve hours (43,200 seconds). This will discourage users trying to use a stolen password from purchasing merchandise with someone else's account. In order to do this, we define a rule that enriches data with an application user name. Then correlate this with the change address and checkout pages of the e-commerce website.

Note: In order to configure this example, you need to previously have configured Web Application User Tracking to track the application user from the login page of the e-commerce website. For more information on Web AUT, see Web Application User Tracking on page 287.

Other use cases for Web enrichment might include:

- Same user performing operations from different IPs in a short time span to issue an alert
- Same IP performing two application operations (identified by URL) within a short time-span (that should not be performed by the same person).
- Track user sequence of operations to understand a process state (for example: login->update cart->logout without payment). You can count these occurrences, and generate alerts.
- Use data enrichment in order to derive application user without text replacement.

To configure data enrichment with an application user name:

1. In the **Main** workspace, select **Policies > Data Enrichment**. The **Data Enrichment Policies** window appears.
2. In the **Enrichment Policies** pane, click . The **Create New Policy** dialog box appears.
3. In the **Create New Policy** dialog box, type **ApplicationUser**, and choose **Web Service Enrichment**, then click **Create**.

4. On the **Rules** tab, next to **Target Field**:
5. Click . A new row is added.
   - Under **Target Field**, select **User Defined Field**, then type the name **ChangeDetails**. This defines that the enrichment data will be kept in new user defined fields called ChangeDetails. This can later be used to reference this information.
   - Click **Save**. Details are saved and the field becomes expandable.
6. Expand the field on the **Rules** tab and complete it as follows:
   - Select the **Enabled** checkbox.
   - Under **Source Event Field**, select **User**.
   - Under **Lookup Data Set**: select **Take Field Value as-is**
   - Expand the **Retention Definitions**.
   - Under **Scope**, select **User**.
   - Under **Valid** For, type 43,200 (the number of seconds in 12 hours).
   - Click **Save**. Details are saved and the field becomes expandable.
7. On the **Match Criteria** tab, click the green arrow of the **URL Prefix** criteria. The criteria is moved up to the **Match Criteria** pane.
8. Expand the criteria and type the URL prefix of the **change address** page of your e-commerce website. For example, changedetails.asp.
9. Click the **Apply to** tab and apply the Enrichment Policy to the relevant web service, then click **Save**. Relevant events are enriched with the application username which can now be used in security policies.
10. Create a **Web Service – Custom** Security Policy and name it **CheckoutDelay**, then configure it as follows:
    - **Action**: Set to block.
    - **Followed Action**: Select **Short IP Block**. This blocks the source IP address for 180 seconds.
    - **Severity**: High
    - **Enabled**: Selected
11. Under Match Criteria, add the **Enrichment Data** criteria by clicking the green arrow, in the **User Defined Field Name** type **ChangeDetails**, and under Operation, select **At least one**.
12. Add the **URL Prefix** criteria by clicking the green arrow, click **New**, then type the URL of the **sales** page on your e-commerce website, for example, checkout.asp.
13. Click the **Apply** tab and apply it to the relevant web service.
14. Click **Save**. Users that change their shipping address will be blocked from buying a product for 180 seconds. This step meant to discourage people with stolen passwords from purchasing a product on a victim's account.

Creating an Enrichment Policy

Enrichment policies determine what event data to enrich.



Note: To use a lookup data set to create an enrichment policy (as opposed to taking information from SQL queries), you need to first configure a Lookup Data set that contains the customized attributes you want to apply to events as described in Creating a Lookup Data Set on page 366.

To create an enrichment policy:

1. In the **Main** workspace, select **Policies > Data Enrichment**. The **Data Enrichment Policies** window appears.
2. In the **Enrichment Policies** pane, click . The **Create New Policy** dialog box appears.
3. In the **Create New Policy** dialog box, type a **Name** for the enrichment policy.
4. From the **Type** dropdown menu, select a type, as follows:

- **Web Service Enrichment**: Creates a policy based on web objects such as HTTP versions, methods and HTTP Request Cookies using the desired match criteria.

5. Click **Create**. The new enrichment policy is created and its details are displayed in the **Details** pane.
6. Configure the enrichment policy as described in Configuring Data Enrichment Policies on page 375.

Configuring Data Enrichment Policies

Data Enrichment policies are comprised of a number of options that assist in customizing their behavior. This procedure provides an overview of how to configure a data enrichment policy.

To configure a data enrichment policy:

1. In the **Main** workspace, select **Policies > Data Enrichment**. The **Data Enrichment Policies** window appears displaying a list of available enrichment policies.
2. In the **Data Enrichment Policies** window, select the policy you want to configure. That policy's configuration details are displayed in the **Details** pane.
3. Configure the policy based on the parameters in the table **Data Enrichment Policy Parameters** below.
4. Click **Save**. The data enrichment policy is saved. If you are in delayed activation mode, you need to : If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Data Enrichment Policy Parameters

Tab	Description
Rules	Determine the items to be enriched, and the values attached to those items. For more information see Configuring Data Enrichment Policy Rules on page 375.
Match Criteria	Select one or more match criteria by which you want to define the policy and configure its details. For more information, see Enrichment Policy Match Criteria on page 607.
Apply To	Apply the data enrichment policy to the desired components. For more information see Applying Policies on page 318.

Configuring Data Enrichment Policy Rules

Data enrichment policy rules determine the type of field you want to add to event data and enable you to select the specific custom attribute you want to use to enrich items. When wanting to enrich an event with values from Lookup Data sets, you need to first create and configure a lookup data set. For more information see Creating a Lookup Data Set on page 366.

To configure enrichment policy rules:

1. In the **Main** workspace, select **Policies > Data Enrichment**. The **Data Enrichment Policies** window appears displaying a list of available data enrichment policies.
2. In the **Data Enrichment Policies** window, select the policy you want to configure. That policy's configuration details are displayed in the **Details** pane.
3. In the **Rules** tab, click **New** , a new row is added to the rules table.
4. Select a **Target Field**. You have the following options:



Note: The options available for the **Target** field depend on what kind of data enrichment policy is being configured.

- **User Defined**: Enables you to manually configure a name to be used as the field. This name (**Target Field Name**) will be displayed when the field is used, for example, in an audit policy.
- **User**: Takes the user name directly from a web event to be used as the field. Only available with Web Service Enrichment policies.

5. If configuring a User Defined policy, type a **Target Field Name** for the field you want to create.
6. Select the desired **Extraction Method**. Available extraction methods include the following:
   - **Lookup from Data Set**: Uses information from a pre-configured lookup data set and assigns it to an event. For information on configuring parameters based on this rule see Configuring Data Enrichment from a Lookup Data Set on page 377.
7. Click **Save** from the upper right-hand corner of the window. Basic enrichment policy parameters are saved and the policy becomes expandable. Expand the rules and configure them as described in step number 6 above.

Configuring Data Enrichment Web Policy Rules

Data enrichment policy rules determine the type of field you want to add to event data and enable you to select the specific custom attribute you want to use to enrich items. When wanting to enrich an event with values from Lookup Data sets, you need to first create and configure a lookup data set. For more information see Creating a Lookup Data Set on page 366.

To configure enrichment policy rules:

1. In the **Main** workspace, select **Policies > Data Enrichment**. The **Data Enrichment Policies** window appears displaying a list of available data enrichment policies.
2. In the **Data Enrichment Policies** window, select the policy you want to configure. That policy's configuration details are displayed in the **Details** pane.
3. Select **Hashing Algorithm** for all the enrichment rules that you define for this policy. Hashing Algorithm is used only in rules where it is enabled, by selecting the **Hash Value** checkbox for each rule.
4. In the **Rules** tab, click **New** , a new row is added to the rules table.
5. Select a **Target Field**. You have the following options:



Note: The options available for the **Target** field depend on what kind of data enrichment policy is being configured.

   - **User Defined**: Enables you to manually configure a name to be used as the field. This name (**Target Field Name**) will be displayed when the field is used, for example, in an audit policy.
   - **User**: Takes the user name directly from a web event to be used as the field. Only available with Web Service Enrichment policies.
   - **Application User**: Takes the user name directly from the database connection to be used as the field. Only available with DB Service Enrichment policies.
   - **Ticket**: Configures the field name as "ticket." Used for configuration with a change management system to enrich any privileged user activity with the ticket ID. Only available with DB Service Enrichment policies.
6. If configuring a User Defined policy, type a **Target Field Name** for the field you want to create.
7. Select the desired **Extraction Method**. Available extraction methods include the following:
   - **Lookup from Data Set**: Uses information from a pre-configured lookup data set and assigns it to an event. For information on configuring parameters based on this rule see Configuring Data Enrichment from a Lookup Data Set on page 377.
   - **From Event SQL**: Extracts literals from an SQL query and assigns it to an event. For information on configuring parameters based on this rule see Configuring Data Enrichment from Event SQL on page 379.
8. Click **Save** from the upper right-hand corner of the window. Basic enrichment policy parameters are saved and the policy becomes expandable. Expand the rules and configure them as described in step number 6 above.

Configuring Data Enrichment from a Lookup Data Set

SecureSphere enables you to enrich events with information in a lookup data set. For example, perhaps you have created a lookup data set that imported users and roles from an Oracle database. You can then configure an enrichment policy to enrich events with this role information. This helps to provide an additional level of information in SecureSphere.

To configure a user defined rule using a lookup data set:

1. Configure an enrichment policy as defined in Configuring Data Enrichment Policy Rules on page 375.
2. In the **Main** workspace, select **Policies > Data Enrichment**. The **Data Enrichment Policies** window appears displaying a list of available data enrichment policies.
3. In the **Data Enrichment Policies** window, select the policy you want to configure. That policy's configuration details are displayed in the **Details** pane.
4. Expand the rule configured for **Lookup from Data Set**. you previously configured. For more information see Creating a Lookup Data Set on page 366. Configuration options are displayed.



Note: if you create a new rule, you have to save the policy before configuring that rule.

5. Configure as shown in the table **Lookup Data Set Configuration** below.
6. Click **Save**. The data enrichment policy is saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Lookup Data Set Configuration

Option	Description
Source Event Field	Select the event field you want to associate with the rule. In order to find the relevant information, the Lookup Data set should have this field configured as one its relevancies. For more information on relevance, see Configuring Lookup Data Set Keywords and Relevance on page 370.
Lookup Data Set	Select the Lookup Data set in which the attribute you want to enrich the event with is located.
Attribute	Select the specific attribute from the lookup data set you want to enrich the event field with.
Source Field Name	The name of the field whose value will be retrieved from Source Event Field. This field is enabled for the following Source Event Field values only: • Header Value • Parameter Value • Cookie Value For example, in order to get the User-Agent header value, configure as follows: **Source Event Field** = **Header Value** and **Source Field Name** = **User-Agent**
Field Name Match	Select **Prefix** or **Exact**. This field is enabled for the following Source Event Field values only: • Header Value • Parameter Value • Cookie Value **Note:** For a web data enrichment policy, if you select **Prefix** and the generated HTTP request contains multiple values which meets the **Prefix** condition, the enrichment data in the alert will show only the first value in the sequence. For example, in order to get the User-Agent header value, configure as follows: • **Source Field Name** = **User-Agent** and **Field Name Match** = **Exact** OR • **Source Field Name** = **User** and **Field Name Match** = **Prefix**
Hash Value	If enabled, the enrichment data are hashed on the MX. It is recommended to enable **Hash Value** for sensitive data.
Retention Definitions	Determines what level to conduct enrichment and other parameters regarding enrichment. Options include the items below: **Note:** The options available depend on the type of data enrichment policy being configured. **Scope** • **Event**: Enriches individual events as they are identified by SecureSphere. • **Stream**: Enriches all events that appear in a TCP Stream. Additionally enables the setting of a timeout. • **User**: Enriches all events that belong to a specific user. • **Source IP**: Enriches all events belonging to a specific IP address. **TTL**: The amount of time a connection can remain idle before timeout. **Order**: The order in which this rule is applied when there is more than one rule in the policy. **Override Old Values**: When enabled, overrides any identical category attributes previously assigned to the event.

EXAMPLE: ENRICH AUDIT DATA WITH DEPARTMENT INFORMATION FROM LDAP

In this example we want to enrich audit data with department information that is being queried from LDAP. We then want to generate a report that shows access to sensitive data by department. SecureSphere provides a built-in Lookup Data set for users and groups in LDAP. This simplifies the process of configuring queries.

Once basic items have been configured, the department field is added to events and can be viewed in audit views and reports.

To configure data enrichment with LDAP:

1. Speak with the SecureSphere Administrator to configure an LDAP endpoint under **Admin > System Definitions > External Systems**.
2. Under **Main > Setup > Global Objects,** select the **Users Roles from Active Directory** Lookup Data set. Configure it as follows:
   - Go to the **Query** tab of the Lookup Data set, select the **External System Type** LDAP, and select the **External System Name** as configured by your administrator.
   - Configure a **Search Base** if desired.

     Note: If you configure a search base here and you receive an error, it may be due to a search base already being configured as part of the external system under Admin.
   - From the **Query Attribute Pane**, click **Test Mapping Query** to verify the query is correct.
   - Click **Run Now** to run the query and populate the Lookup Data Set.
3. Click **Save**. The Lookup Data set is saved.

   Because the default Enrichment policy is pre-configured, there is no need to modify it. To view this default policy, under **Main > Policies > Data Enrichment**, select the **Default enrichment policy for domain users information**. This policy is also applied by default. Once the lookup data set has been configured and populated from LDAP, SecureSphere automatically begins to audit login events with the configured attributes as can be seen in the Rules tab.
4. In **Main > Policies > Action Sets**, create an e-mail **Action Set** and configure it with the required parameters.
5. In **Reports > Manage Reports**, create an audit report and configure it with the following details:
   - In the **General Details** tab, attach the action set created in step 4 as a **Followed Action**.
   - In the **Data Scope** tab, configure the time frame for the **Last 1 Week**, then add the **Sensitive Data Access** match criteria to the enabled fields by clicking its arrow, and set it to **True**.
   - In the **Tabular Tab**, add three columns then configure them as **OS User Name**, **Department**, **Enrichment Data**.
   - In the **Scheduling** tab, configure the report to be generated Weekly, Every Monday at 08:00 AM.
6. Click **Save**. The report configuration is saved. A report will now be generated every Monday and e-mailed out to as configured listing users that have accessed sensitive data.

Configuring Data Enrichment from Event SQL

SecureSphere enables you to enrich events with information (literals) taken from an SQL query. This can help you, for example, extract an application username and use it to enrich event information in SecureSphere. Later enabling you to search for events based on this user name, and generate reports.

To configure a user defined rule to query from an SQL Event:

1. Configure an Enrichment policy as defined in Configuring Data Enrichment Policy Rules on page 375.
2. In the **Main** workspace, select **Policies > Data Enrichment**. The **Data Enrichment Policies** window appears displaying a list of available data enrichment policies.
3. In the **Data Enrichment Policies** window, select the policy you want to configure. That policy's configuration details are displayed in the **Details** pane.
4. Expand the rule configured for **From Event SQL**. Configuration options are displayed.

5. Configure as described in the table below.
6. Click **Save**. The data enrichment policy is saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.



Note: In order the following options to be available, the Data Enrichment Policy configured in step one must have a rule configured with a From Event SQL extraction method, and the policy must have been saved.

Option	Description
Set/Remove field value	Select whether the operation will set enrichment properties based on the SQL value or remove them.
Extract Literal	Extracts a specific literal from a query and uses it to enrich data. **Extract literal #**: Extracts the specific literal from the query based on its chronological presence in the query. For working with Stored Procedures see Working with Stored Procedures on page 381. **From Query**: Defines the query from which you want to extract the literal. The query must appear exactly as it appears in the Parsed Query column in audit data. **Note:** An example of this can be seen in the pre-defined enrichment policy named **Enrichment Policy for Change Management Tickets**.
Additional Conditions	Further conditions the extraction of the literal based on the presence of another. **When Literal #**: Defines the specific literal for conditioning. **Is**: Exact value of the literal.
Transform Value	Enables you to transform a value extracted by substituting it with a more user friendly value. • **Pattern**: Defines a pattern used to search for specific values that may be in a column. • **Substitution**: Replaces the information from the pattern with a desired format of information.
Retention Definitions	Determines what level to conduct enrichment and other parameters regarding enrichment. Options include: **Scope** • **Event**: Enriches individual events as they are identified by SecureSphere. • **Stream**: Enriches all events that appear in a TCP Stream. Additionally enables the setting of a timeout. • **User**: Enriches all events that belong to a specific user. • **Source IP**: Enriches all events belonging to a specific IP address. **TTL**: The amount of time a connection can remain idle before timeout. **Order**: The order in which this rule is applied when there is more than one rule in the policy. **Override Old Values**: When enabled, overrides any identical category attributes previously assigned to the event.

EXAMPLE: CONFIGURING DATA ENRICHMENT TO TRACK ACCESS TO SENSITIVE DATA FROM EVENT SQL

In this example we want to enrich audit data with a customer's name so you can see all users that accessed specific data regarding that customer. This can help track leaked data about people, for examples celebrities or politicians. Please note that this example is only an illustration and should be modified to match your working environment.



Note: This feature needs to be pre-configured. Only once it has been configured will it audit data which can be reviewed.

1. Examine queries generated by your application when relevant data is viewed (for example when customer details form is opened). You should look for a query which contains the customer identifier for example, name, ID or account number.
2. In the **Main** workspace, select **Policies > Data Enrichment**. The **Data Enrichment Policies** window appears.
3. In the **Enrichment Policies** pane, click . The **Create New Policy** dialog box appears.
4. In the **Create New Policy** dialog box, type a **Customer Details**, and choose **DB Service Enrichment**, then click **Create**.
5. On the **Rules** tab, next to **Target Field**:
   - Click . A new row is added.
   - Under **Target Field**, select **User Defined Field**.
   - Under **Target Field Name**, type **CustomerName**.
   - Under **Extraction Method**, select **From Event SQL**.
   - Click **Save**. Details are saved and the field becomes expandable.
   - Select the Enabled checkbox.
6. Expand the Field on the services tab and complete it as follows:
   - Select the **Set Field Value** option.
   - In **Extract Literal #**, type the number of the literal from the query which contains the customer name.

     Note: You can copy the query to the query field and save it. SecureSphere replaces all literals in the query with a question mark (?), so you can count the ? until you get to the relevant literal.
   - In **From Query**, type the query. In our example, we'll use the following:
     ```
     select * from customer_details where customer_name=?
     ```
   - In **Retention Definitions** under **Scope**, select **Event**.

   Every time SecureSphere sees this query, it will take literal #1, and put it in user defined field called customer name.
7. Apply the Enrichment Policy to the relevant service, then click **Save**. Relevant events are enriched with this data which can now be used in security and audit policies.
8. Configure a Security Policy with the enriched event data to generate an alert when a specific customer's data has been accessed.
9. Configure an Audit Policy to index the enrichment data you configured so it can later be used for auditing. This is done by selecting the audit policy, then on the **Settings** tab, under **Data Index > Indexed User Define Fields**, select **CustomerName** (the item you configured in step 5 above) as User Defined 1 or User Defined 2. This collects all data that has been enriched with the customer name.

   You can then see audit information regarding the enriched data in audit views:
   - It appears in the event itself you can see in SecureSphere auditing by clicking retrieve event data, then selecting **User Defined 1** (user defined field associated with the enrichment data configured in step 9).
   - If you defined the field in the audit policy as User Defined 1, it appears in this field in Data view. You need to select this column or look it up in a filter.

Working with Stored Procedures

Stored Procedures and their parameters can be reflected in SecureSphere as follows:

- **Format 1**: The stored procedure appears as normal query with the parameters represented as literals,

  OR

- **Format 2**: A query that contains only procedure name and the parameters appear as bind variables in a dedicated column in the audit data.

The definitions of data enrichment are differ according to the stored procedure format, as presented in the following example. In the following example the stored procedure called sp-login gets two parameters called Name and Type. For data enrichment we use the value of the parameter called Name.

Data Enrichment with Stored Procedure Examples

	Raw Query (as it appears in Audit)	Bind Variables	Parsed Query	Enrichment Settings
example for format 1	sp-login :Name='John', :type='user'		sp-login ?=?,?=?	Extract literal#2 from query sp-login ?=?,?=?
example for format 2	sp-login	Name='John', type='user'	sp-login	Extract literal#1 from query sp-login



Note: The parameter used for enrichment must be textual. For example, parameters of VARCHAR type are supported and parameters of Number type are not supported.

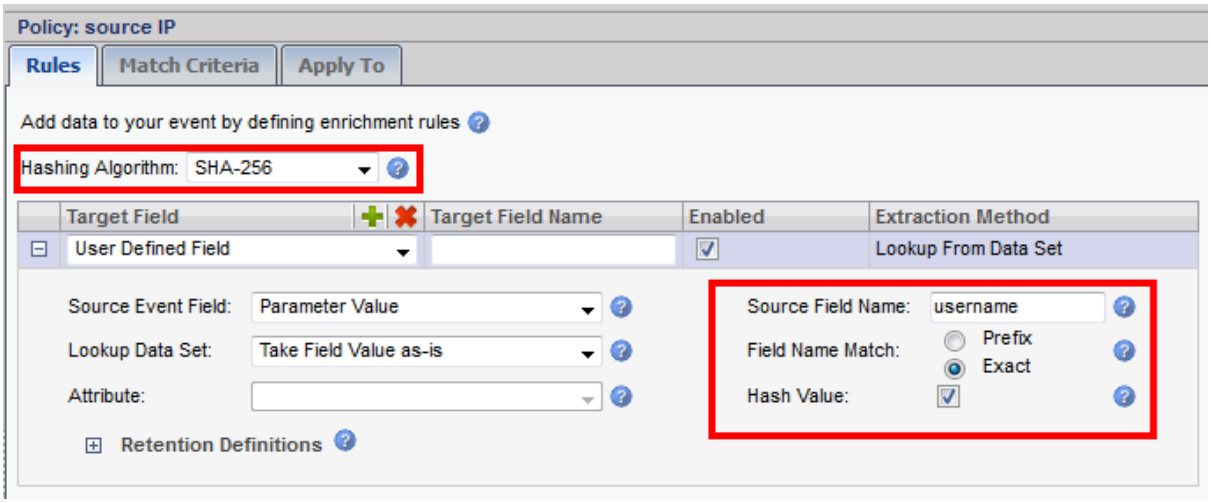
Content Hashing in Data Enrichment Policies

SecureSphere enables you to hash sensitive values that may be used as part of Data Enrichment Policies.

This is particularly useful when working with Imperva's Fraud Prevention solution to communicate with 3rd party fraud providers. Data Enrichment Policies can be used to extract specific information which can then be sent to these providers by allowing differentiation between values, without revealing actual data such as user names, headers, cookies, etc.

To configure content hashing:

1. Create a new Enrichment Policy or edit an existing one as described in Creating an Enrichment Policy on page 374.
2. On the **Rules** tab of the **Dataset**, configure the information as shown below.



- **Hashing Algorithm**: Select the algorithm used to hash the data. Options include SHA1, SHA-256 and SHA-512.

- **Source Field Name**: When selecting a **Source Event Field** of types Cookie Value, Header Value, or Parameter Value, enables determining which Source field will be taken from the event and hashed as part of the Lookup Data Set, and use it as the basis of enrichment. Note that if not selecting one of these Source Event Fields, the Source Field name is disabled.

  For example, if you want to obtain the value of the parameter **username**, you'd set **Source Event Field** to **Parameter Value**, then under **Source Field Name** to the right, you'd type **username**.
  So in the following URL, the **bolded** item is the Parameter Value, and SecureSphere would extract, in this case Jane.

  www.myapp.com/login.jsp?**username=jane**
- **Field Name Match**: Determines if the source field name needs to be the full parameter name, or only a prefix, which is useful for cookies. Options include exact (full parameter name), or prefix.
- **Hash Value**: Enable this option to turn on hashing.

3. Complete configuration of the Lookup Dataset as required.
4. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

   Once configured, SecureSphere hashes fields that match the configuration using the hashing algorithm you selected.

Introducing Active Modules

The active module engine is the platform on which Imperva bases the value-added capabilities generated by the ADC. These features include:

- **Change Tracking**: Enables the periodical scanning of specific databases and web configuration files external to SecureSphere. For more information, see Change Tracking - External Objects on page 383
- **Track Value Changes**: Enables the periodic generation of a report that list changes to specific database columns in specific database tables. For more information, see Track Value Changes
- **Legacy Active Module Support**: Provides explanation about the Auto Server discovery module, the Sensitive Data Discovery Module and the Web Vulnerability module. For more information see Legacy Active Module Support

Change Tracking - External Objects

This section reviews the various aspects of working with SecureSphere Change Tracking - External Objects. It reviews the following subjects:



Note: For information regarding reports on change tracking - external objects, see:

- Understanding Change Tracking Reports
- Pre-Defined Reports on page 636

Overview of Change Tracking - External Objects 384
Before Running a Change Tracking - External Objects Policy 384
Creating a New Change Tracking - External Objects Policy 385
Viewing a Change Tracking - External Objects Policy 386
Editing a Change Tracking - External Objects Policy 386

Overview of Change Tracking - External Objects

SecureSphere's Change Tracking - External Objects feature enables administrator to periodically scan specific database and web configuration files external to SecureSphere, for example, to Oracle configuration files, and identify changes to those files. Change Tracking enables tracking of changes of the operating system files on any server protected by SecureSphere, monitoring not only changes to database content and structure, but also to the underlying database server configuration.

For example, the Change Tracking Active Module detects changes to the database server environment and underlying OS configuration such as start parameters configuration, database executable, network configuration files and OS parameter variables.

SecureSphere Change Tracking can be used to track changes on files belonging to the following Operating Systems:

- Unix OS: AIX, HP-UX and Solaris
- Linux
- Windows

Before Running a Change Tracking - External Objects Policy

Before you can run a Change Tracking - External Objects policy, you must:

- Have appropriate permissions on the tracked host as described in Change Tracking - External Objects Policy Permissions on page 384
- Create a configuration file on the tracked host as described in Change Tracking - External Objects - Policy Configuration on page 384

Change Tracking - External Objects Policy Permissions

In the **Direct Access Information** tab of the server group to which the tracked host belongs, define **SSH User Name** and **OS Type** for each IP address.

Change Tracking - External Objects - Policy Configuration

In order to run Change Tracking, you need to create a temporary local directory to store information and a support file.

1. Create a temporary local directory to store the information on each target host (you may use an existing directory). The user defined in the previous step must have the following permissions:
   - write permissions for the temporary directory
   - read permission for all the files on which the track changes is applied
2. Create a file named `.secsph.rc` on each designated host target in the home directory of the user defined in the **Direct Access Information** tab. The file must define the following two variables on each host:
   - The home directory of each one of the databases:
   - ORACLE_HOME for Oracle
   - DB2PATH for DB2
   - SYBASE for Sybase
   - The `CTDATA` variable that points to the temporary directory defined in the previous step.

   For example, if you run this active module on Oracle 9.2 called `proddb` on Linux, the `.secsph.rc` file contains the following:

   ```
   export ORACLE_HOME=/oracle/proddb/9.2.0
   ```

```
export CTDATA=/tmp
```

Creating a New Change Tracking - External Objects Policy

To create a Change Tracking - External Objects policy:

1. In the **Main** workspace, click **Setup** > **Active Modules**.
2. In the Filter pane, under **By Module**, click **Change Tracking - External Objects**.
3. In the middle pane, click .
4. In the **Create New Policy** window, enter the following data:
   - **Name**: Enter a name for the policy.
   - **Description**: Optionally, enter a description of the policy.
   - **Module**: Select **Change Tracking - External Objects** from the list.
   - **Followed Action**: Optionally, select an Action Set to be performed when a report is successfully created.
5. Click **Create**.
6. In the **Details** tab, under **Policy Parameters**, enter the following data:
   - Select the database files for which you want to track changes. You can select any number of items from the list.

     Note: Predefined Registry Keys are not currently available.

   - To define a pattern that restricts the tracking to specific objects, click  under **User Defined Patterns**, and enter the following data as described in the table **User Defined Pattern Options** below.
7. In the **Apply To** tab, select the server groups,web, or database services to track. For more information, see Security Policy Exceptions on page 173.

   Note: Changing tracking supports tracking of changes to configuration files of web servers.

8. In the **Scheduling** tab, select one of the following:
   - **None**: Change Tracking - External Objects is not scheduled, and can be run only manually.
   - **Once**: Change Tracking - External Objects is run on the date and time you specify in **At** and **At Time**.
   - **Recurring**: Change Tracking - External Objects is run periodically according to the schedule you define.

     Note: The **Scheduling** tab defines when Change Tracking - External Objects is run. The **Schedule** option in the **Archive** tab defines when the data which Change Tracking - External Objects produces is archived.

9. In the **Archive** tab, define how the data produced by the scan will be archived. For more information, see Configuring and Applying Archive Settings on page 325.
10. Click **Save**.

    The policy is saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

User Defined Pattern Options

Parameter	Description
Object Type	The type of the object to be tracked. Enter a **File.**
Pattern	A pattern qualifying the objects to be tracked. Patterns can use the `*`, `?` and `$` wildcard characters. For example, if **Object Type** is **File**, then: `*.config` means all files whose names end with the suffix `config` `/etc/*.conf` means all files in the `/etc/` directory whose names end with the suffix `conf`, such as `resolve.conf`, `hostmane.conf` and so on. `a?b.info` means files named, for example, `aab.info`, `axb.info` and so on. `/etc/?bc.conf` means files named, for example, `/etc/abc.conf`, `/etc/bbc.conf` and so on. **Note:** Specify the full path, for example: `/home/oracle/product/10.2.0/db_1/network/admin/a?b.gif` You can also use environmental variables defined in the `.secsph.rc` file. For example, if `$MyEnvVar` is defined in `secsph.rc` as `Export MyEnvVar=/etc`, then `$MyEnvVar/path/*` means `/etc/path/*`
Report Missing	Specifies whether to track the absence of the objects defined by the above parameters. Enter **TRUE**, **FALSE**, **YES** or **NO**. If **TRUE** or YES is **specified**, then a change record is generated each time that the specified item cannot be found, including the first time.

Viewing a Change Tracking - External Objects Policy

To view a Change Tracking - External Objects policy:

1. In the **Main** workspace, click **Setup** > **Active Modules**.
2. In the **Filter** pane, under **By Module**, click **Change Tracking - External Objects**.
3. In the middle pane, select a Change Tracking - External Objects policy.
   The policy is displayed in a number of tabs in the rightmost window.

Editing a Change Tracking - External Objects Policy

To edit a Change Tracking - External Objects policy:

1. In the **Main** workspace, click **Setup** > **Active Modules**.
2. In the **Filter** pane, under **By Module**, click **Change Tracking - External Objects**.
3. In the middle pane, select a Change Tracking - External Objects policy.
   The policy is displayed in a number of tabs in the rightmost window.
4. Edit the information in the tabs as required.

5. Click **Save**.

   The policy is saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Application Structure Modeling

Application structure modeling is a multi-phase process in which first SecureSphere learns usage in your network and builds a dynamic profile of its day to day operation (structure), then enables you to fine-tune SecureSphere operation by observing various aspects of its behavior and using plugins to handle dynamic behavior that may be part of your work environment, and learn how to parse traffic effectively. Application modeling is built around a number of primary features, these include plugins, and text replacement. For more information about how plugins assist in handling dynamic behavior, see Plugins Overview on page 478.

Working with Tasks

This section reviews the various aspects of working with SecureSphere tasks. It reviews the following subjects:

- Tasks Overview on page 387
- Task Types on page 388
- Understanding the Tasks List on page 388
- Task Followed Action Configuration Guidelines on page 389
- Creating Review Tasks on page 390
- Creating Assignment Tasks on page 390
- Working with Open Tasks on page 391

Tasks Overview

Tasks in SecureSphere are actions assigned to users to facilitate follow up. They enable you to provide an additional layer of supervision by assigning and tracking tasks that need to be conducted. Additionally, tasks can help to meet compliance requirements by creating a review cycle and notifying users when urgent tasks have not been completed by a certain time. You can configure some tasks to be automatically generated, while manually configuring others.

EXAMPLE: TASKS USE CASE

SecureSphere has generated an assessment report identifying missing permissions. Together with this report a review task is automatically generated and assigned to the IT officer. In addition, an e-mail notification is sent to the company's compliance officer. The IT officer decides that the missing permissions are important and need to be addressed immediately. He adds a comment to the task and approves it, then re-assigns it to the Webmaster for completion. Once again an e-mail notification is sent to the compliance officer keeping him notified. The webmaster constantly puts off completing the task. As a result the task passes its overdue date and e-mails are sent out to both the IT and compliance officers. As a result the IT officer pays a visit to the webmaster, and together they resolve the issue, ensuring that the company remains compliant.

To view the Tasks workspace:

- In the upper-right hand corner of SecureSphere, click **Tasks**.

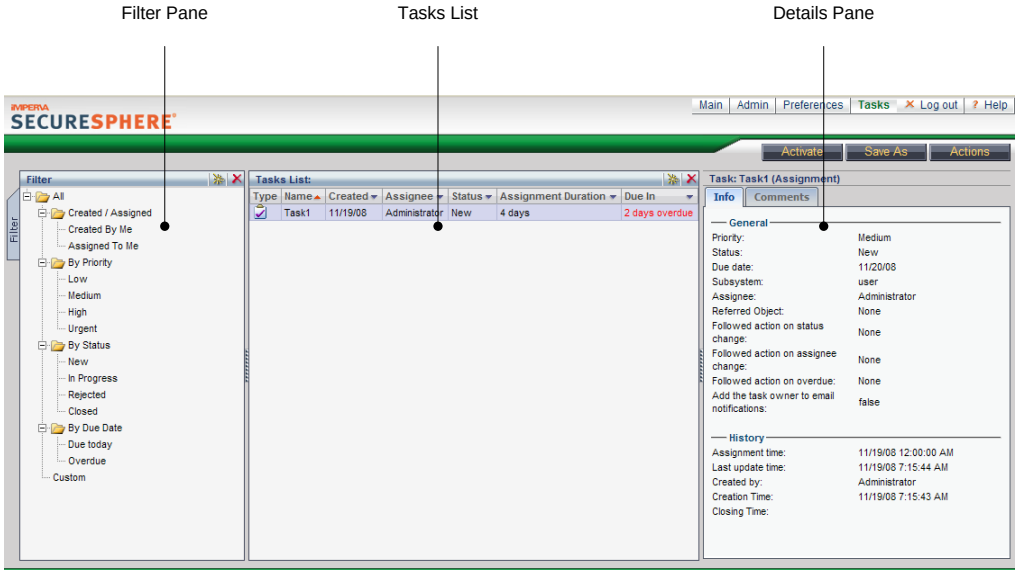


Figure 65: Tasks Workspace

The Tasks workspace has three primary sections:

- **Filter Pane**: Enables you to filter items by category including Create/Assigned, By Priority, By Status, and By Due Date.
- **Tasks List**: Lists tasks and their details summarizing tasks information in a single row. For more information on the Tasks List, see Understanding the Tasks List on page 388
- **Details Pane**: Provides detailed information regarding the selected tasks and enables users to add comments to tasks.

Task Types

SecureSphere provides two types of tasks:

- **Review Tasks**: Tasks that are automatically generated and linked to an item in SecureSphere. For example, a policy configured to generate report can also be configured with a review task. This review task is generated together with the report, and linked to the report to provide trackable follow-up. For more information on Review tasks, see Creating Review Tasks on page 390.
- **Assignment Tasks**: Tasks that are not linked to a specific item. For example, an assignment task can be created if you see an alert generated by problematic code in the web server that you want the web admin to review. You can manually assign a task for the web admin to review. For more information on Assignment Tasks, see Creating Assignment Tasks on page 390.

Understanding the Tasks List

The Tasks List provides a summary of information regarding existing tasks. The following table reviews columns available in the Tasks List.

Tasks Table Contents

Column Heading	Description
Type	Represents the type of task. Two task types are available in SecureSphere:  **Review Task**: Used to generate a task that is assigned to users that is linked to an object. For more information, see Creating Review Tasks on page 390  **Assignment Task**: Used to assign tasks not linked to an object. For more information, see Creating Assignment Tasks on page 390.
Name	Name of the task. This name is determined by the Name parameter in the source action interface.
Created	Date the task was created.
Assignee	Person to whom the task is currently assigned. This field is updated if a task is reassigned.
Status	Status of the task. Available tasks include New, In Progress, Closed and Rejected.
Assignment Duration	Number of days configured for the task to be completed.
Due In	Actual number of days until the task is due to be completed.

Task Followed Action Configuration Guidelines

Both Review and Assignment tasks can be configured as followed actions. Review tasks can only be configured as a followed action, while Assignment tasks can also be manually configured in the Tasks pane.

You may want to configure an Assignment task as a followed action to automatically follow up on any number of activities such as when a task's status or assignee has been changed, is the task is overdue, or to notify of other problematic activity. For example, you may want to configure a system event that notifies of low disk space, or an audit policy that failed to generate an Assignment task to mitigate the failure. Configuring Followed Actions involves the tasks listed in The following table.

Configuring Tasks – Task Overview

	Action	Description	For more information, see...
1	**Create Action Sets.**	Create an action set to be assigned as a followed action, which generates a task.	Creating Custom Action Sets on page 330.
2	**Configure Action Interfaces.**	Add Review or Assignment action interfaces to the action set and configure its parameters.	Configuring Action Interface Parameters on page 332.
3	**Configure Policy with Followed Actions.**	Attach the action set to a policy as a followed action.	Assigning Followed Actions on page 341.

Creating Review Tasks

Review tasks are tasks linked to a specific object in SecureSphere. For example, you can configure SecureSphere to generate a review task every time a report is generated, and have this report linked in the task. This task can be assigned to any number of users, which enables creating a review workflow. Then, when a report is generated, a task is automatically created for the designated users, who are then notified about the task. They can open the task, see the linked object, and mark it as completed, or alternatively reject the task.

Example: Your organization generates an assessment report once a week. This report is distributed to a number of executives within the company. You want the report reviewed by both the IT and compliance officers before it is distributed to these people. You can configure a review task and attach it to the policy that generates the assessment report.

Review tasks are created as Followed Actions, and then attached to a policy. For an overview of this process see Task Followed Action Configuration Guidelines on page 389.

For information on how to work with review tasks, see Completing Review Tasks on page 391.

Creating Assignment Tasks

Assignment tasks are tasks assigned to users that are not linked to a SecureSphere object used to follow up on general tasks. Once a task is created, it is open until a user accepts or rejects it. Users can see open tasks by logging in and viewing the Tasks workspace. They can then accept the task, or reject the task then reassign it to a different user. Assignment tasks are created by completing a form which defines the task and assigns followed actions to enable tracking of the task. There are two ways of creating assignment tasks, each of which has a different purpose.

- **Automatically generated as the response to an event**: Used to create tasks as a response to an event. For example, perhaps a system event was generated indicating disk space is low, and the IT administrator should check the problem, or perhaps a scheduled audit failed and the Network Administrator needs to resolve the problem. For more information on creating tasks as followed actions, see Task Followed Action Configuration Guidelines on page 389.
- **Manually created in the Tasks workspace**: Used to create tasks for general reasons. For example, perhaps you received an alert regarding problematic code on the web server, and you want the web master to review the problematic code. You can manually create an assignment task for his purpose.

To manually create an assignment task:

1. In the **Tasks** workspace, from the **Tasks List** title bar, click  **New**. The **Assignment Task** dialog box appears.
2. In the **Assignment Tasks** dialog box, complete the items as shown in the table **Assignment Task Options** below.
3. Click **Create**. The assignment task is saved down and added to the **Tasks** list. The assignee can see the new task by logging in and opening the Tasks workspace.

   For information on how to complete review tasks, see Completing Assignment Tasks on page 392.

Assignment Task Options

Option	Description
Name	Type a name for the task.
Priority	Select the desired priority from the dropdown list. Options include Low, Medium, High and Urgent.
Due in	Type the number of day in which the task should be completed. Tasks are due at 12:00 am on the due date.
Assign to	Select a system user to whom to assign the task.
Followed action on status change (optional)	Select a pre-defined followed action to conduct when someone changes the status of the task.
Followed action on assignee change (optional)	Select a pre-defined followed action to conduct when someone reassigns the task to another assignee.
Followed action on overdue (optional)	Select a pre-defined followed action to conduct once the amount of time that has passed since the task was assigned surpasses that defined in the **Due in** field.
Add the task owner to email notifications	When enabled, e-mail's task owner with any changes regarding the task.

Working with Open Tasks

Once a task has been created, it appears in the **Tasks** List, and its details are displayed in the Details Pane. Assignees can then see their open tasks when they log in and access the Tasks workspace. The workflow for working with tasks is different depending upon the type of task that has been assigned. The two workflows include:

- Completing Review Tasks on page 391
- Completing Assignment Tasks on page 392



Note: SecureSphere checks every day at 12:00 am to see if there are any tasks due. If a task's status has not been changed by this time on its due date, it is marked by SecureSphere as Overdue.

Completing Review Tasks

Review tasks can be assigned to a number of parties. Review tasks cannot be reassigned, only accepted or rejected.

To complete a review task:

1. In the **Tasks** workspace, select an active task. The task's details are displayed in the **Details** pane.
2. Review the task to understand its requirements.
3. In the **Details** pane, click the **Referred Object** hyper link. The referred object is opened.
4. Examine the referred object.
5. Click the **Comments** tab and add a comment if required. This step is optional.

6. Click the **Actions** button and mark the task as either **Signed** or **Rejected**. If the task is signed, it signifies that the task has been completed and its status is changed to closed. If the task is rejected, it signifies that the assignee has decided not to take on the task, and the task's status is changed to rejected.

Completing Assignment Tasks

Assignment tasks are reviewed by the assigned party only. If the assigned party rejects the task, it can be reassigned.

To complete an assignment task:

1. In the **Tasks** workspace, select an active task. The task's details are displayed in the **Details** pane.
2. Review the task to understand its requirements.
3. Click the **Actions** button and mark the task as accepted or rejected:
   - **Accepted**: Signifies that the user has agreed to take upon the task.
   - **Rejected**: Signifies that the assignee has decided not to take on the task. The task's status is changed to rejected. It can then be reassigned or closed.
4. Click the **Actions** button and mark the task as closed, rejected or reassigned.
   - **Reassigned**: Reassigns the task to another system user.
   - **Closed**: Signifies that the assigned task has been completed.
   - **Rejected**: Signifies that the assignee has determined they will not complete the task. This option is not available if the task was marked as rejected in step 3.

Configuring Preferences

SecureSphere enables you to configure a number of details about your account that assist in system management and customizing how certain items are displayed. Items that can be configured include:

- **User Details**: Including your name as used for tasks and e-mail address. For more information see Configuring User Details on page 392.
- **Password**: Enables you to change your password. For more information, see Changing Passwords on page 393.
- **Preferences**: Including your default home page and other display options. For more information see Configuring User Preferences on page 393.

Configuring User Details

User details are used to identify you in SecureSphere for use in tasks and the e-mail addressed for notification. User details are user specific, and relate to the currently logged in user.

To configure user details:

1. In the **Preferences** workspace, click **User Details**. User details are displayed.
2. Update the following information as required:
   - **Username**: Username assigned by the SecureSphere administrator. This field is disabled and is for viewing purposes only.
   - **Full name**: Name as you want it displayed. This field is used with tasks.
   - **Email**: Your e-mail address. Used in connection with action sets that generate e-mail to users.
   - **Phone**: Your phone number.
   - **Title**: Your title.
   - **GEO location**: Your geographic location.
   - **Department**: The department you belong to.

- **Last Login**: The time and date of your last login to SecureSphere.
- **Password Updated**: The last time and date of the last time your password was updated.



Note: You can change your password by clicking Change Password, as described in Changing Passwords on page 393.

3. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Changing Passwords

SecureSphere provides an easy mechanism that enables you to change your password. It is recommended that passwords are changed on a regular basis. Passwords must conform to the rules set under **Admin > System Definitions > Password Settings.**

To reset your password:

1. In the **Preferences** workspace, click **User Details**. User details are displayed.
2. From the bottom of the **User Details** pane, click **Change Password**. The **Change Password** dialog box appears.
3. Type your old password.
4. Type a new password.
5. Re-enter the new password for confirmation.
6. Click **Change**. Your password is changed.

Configuring User Preferences

User preferences are items that enable you to configure how SecureSphere displays information when you are logged in. User preferences are user specific, and relate to the currently logged in user. Items that can be configured include the home page that is displayed when you log into SecureSphere, the number of items listed on each page, and the number of days for which events are displayed.

To configure user preferences:

1. In the **Preferences** workspace, click **Preferences**. User preferences are displayed.
2. Configure the following information as required:
   - **Home Page** : Select the home page you want displayed immediately following logon. Available options are based on your access rights.
   - **Paging**: Enter the maximum number of items (rows) you want to be displayed on a page. In order for this parameter to go into effect you must logout and log back in.

     This setting will take effect after the SecureSphere GUI is restarted.
   - **Default Time Frame for Event Filtering**: Type the number of days for which you want to display events and alerts.
3. To restore the default "factory" settings for the currently logged-in user, click **Restore settings**.
4. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

Configuring Integration with Third Party Management Systems

SecureSphere enables you to configure integration with a number of third party management systems for tracking alerts and events that take place in your network. SecureSphere supports integration with the following 3rd party systems:

- **ArcSight Event Collection System**: Enables the sending of SecureSphere events to ArcSight for management and mitigation. For more information see, Configuring Integration with ArcSight on page 394.
- **BMC Remedy Event Management System**: Enables the sending of SecureSphere events to BMC Remedy for management and mitigation. For more information see, Configuring Integration with BMC Remedy Service Desk on page 395.
- **RSA enVision platform**: Provides an effective security and information event management (SIEM) and log management solution. For more information see, Integrating with RSA enVision on page 398.
- **Active Directory System**: Manages roles and users from a central location out of SecureSphere - Active Directory. For more information see, Configuring Integration with Active Directory for User Management on page 403

Configuring Integration with ArcSight

This procedure describes how to configure SecureSphere integration with ArcSight, enabling the reporting of SecureSphere events to ArcSight where events can be correlated and processed.

Configuring integration with Arcsight involves the tasks listed in the following table.

Configuring Integration with Arcsight – Task Overview

	Action	Description	For more information, see...
1	**Create action set.**	Create an action set to be assigned as a followed action, which generates a task.	Creating Custom Action Sets on page 330.
2	**Configure Action Interfaces.**	Add Review or Assignment action interfaces to the action set and configure its parameters.	Configuring Action Interface Parameters on page 332.
3	**Configure Policy with Followed Actions.**	Attach the ArcSight action set to a policy as a followed action.	Assigning Followed Actions on page 341.

To configure integration with ArcSight:

1. Create a new System Event Type Action Set with an intuitive name. For example, ArcSightSyslogEvent as described in Creating Custom Action Sets on page 330.
2. In the **Main** workspace, select **Policies > Action Sets**. The **Action Sets** window appears.
3. Select the new Action Set created in step 1 above. Available action interfaces are listed in the Action Interface pane.
4. Add the desired action interface to the Selected Actions pane by clicking on its green arrow. Note there are four event types for CEF Standard:
   - **Regular Security Event** (System Log > Log security event to System Log (syslog) using the CEF standard): Logs a regular security event to System Log (syslog) using the CEF standard.
   - **Custom Event** (System Log > Custom Policy Security Event): Logs a custom event to System Log (syslog) using the CEF standard.

- **Firewall Security Event** (System Log > Firewall Security Event): Logs firewall event to System Log (syslog) using the CEF standard.
- **System Event** (System Log > Log system event to System Log (syslog) using the CEF standard): Logs a system event to System Log (syslog) using the CEF standard.

5. Expand the selected CEF System Log action interface by clicking on the plus sign (+) to its right.
6. Configure it as follows:
   - **Name**: Type a name for the Syslog event.
   - **Syslog Host**: Type the IP or host name of the ArcSight server.
   - **Syslog Log level**: Select the desired syslog log level from the dropdown list (info, warn, debug or error).
   - **Message**: Type a message with placeholder information to be used by syslog to create a message readable by ArcSight. This message must follow CEF guidelines. For a description of the CEF syslog message, including syntax and available placeholders, see ArcSight Placeholders and Syntax on page 535.

     Note: The message field may be locked by your SecureSphere administrator. In this case the message is predefined and can only be modified by the administrator.

   - **Facility**: Select the required facility (type of authorization required by your installation of Arcsight).
   - **Run on every event**: Select this checkbox if you want to send a syslog event to Arcsight for every event.
7. In the **Policies > Security** window, select the action set created in step 1 from the followed action dropdown list of the policy you want to configure.
8. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

   When a violation occurs, an alert is generated and a syslog message is sent to ArcSight.

Configuring Integration with BMC Remedy Service Desk

BMC Remedy Service Desk is an incident and problem automated workflow solution. BMC Remedy Service desk integration allow SecureSphere to open service desk tickets for any event type.

This procedure describes how to integrate SecureSphere with BMC Remedy Service Desk. The integration also facilitates working with other BMC Remedy ARS applications by using a different web service and a soap message as described below. Integration is accomplished by defining the Remedy Create incident action set with the necessary configuration items, then, then configuring SecureSphere to send events. This enables the opening of a Remedy ITSM ticket for relevant alerts, specific system events, and reports generated in SecureSphere. Configuring integration with BMC Remedy involves the tasks listed in the table **Configuring Integration with BMC Remedy – Task Overview** below.

Note: Out of the box SecureSphere provides full integration with ticket creation for the BMC Remedy Service Desk Application. If a different application is defined to handle tickets, then you should modify the web service and the SOAP request used for this integration.

Configuring Integration with BMC Remedy – Task Overview

	Action	Description	For more information, see...

	Action	Description	For more information, see...
1	**Complete Prerequisites.**	Complete prerequisites in order to prepare Remedy for integration.	Remedy Server Configuration Prerequisites on page 396
2	**Configure SecureSphere action interface.**	Configure integration of SecureSphere with the Remedy server.	Configuring SecureSphere for Integration with BMC Remedy Service Desk on page 396
3	**Configure Policy or Report with Followed Actions.**	Attach the Remedy action set to a policy or report as a followed action.	Configuring Policy or Report with Followed Action for Remedy on page 397.

Remedy Server Configuration Prerequisites

In order to configure integration between Remedy and SecureSphere, you need to conduct the following tasks on the Remedy server.

1. Create a user for authentication to the Remedy server - if required (not the user who is the assignee of the incident).
2. Create a customer user ("First Name" and "Last Name") that can create incidents - if required. This is not the same as the authentication user.

Configuring SecureSphere for Integration with BMC Remedy Service Desk

This procedure describes how to configure the necessary SecureSphere components for integration with BMC Remedy Service Desk application.

To configure SecureSphere for integration with BMC Remedy Service Desk:

1. Create a new Action Set with an intuitive name. For example, Remedy Create Incident as described in Creating Custom Action Sets on page 330.
2. In the **Main** workspace, select **Policies > Action Sets**. The **Action Sets** window appears.
3. Select the new Action Set created in step 1 above. Available action interfaces are listed in the **Action Interface** pane.
4. Add the interface named **Remedy Create Incident > Remedy Create Incident** to the **Selected Actions** pane by clicking on its green arrow.
5. Configure the following:
   - **Name**: Name for the interface.
   - **XML SOAP Request**: A pre-configured soap request used to pass information to the Remedy BMC Service Desk application.

     Note: To send custom fields to the ticket you can edit the XML field in the action interface. To integrate with another BMC Remedy application you should replace the existing XML SOAP request with a different request that supports the different web service (see web service below).
   - **Host**: Host name of the web-server. Can be an IP address or web host name.
   - **Protocol**: Protocol used to communicate with Remedy. Options include HTTP and HTTPS.
   - **Port**: Port number used to communicate with Remedy.
   - **Remedy Server**: Name of the Remedy server that runs on the web server. Type a valid HTTP name.
   - **Web Service**: The web service HPD_IncidentInterface_Create_WS is used to open an incident (ticket) in BMC Remedy Help Desk. You can use a different web service to create a tickets for a different application. If you choose to use a different web service, you must replace the XML SOAP Request that invoke the service.

Note: This field is defined by the SecureSphere administrator and locked for editing.

- **Username**: User name used to access the Remedy server.
- **Password**: Password required for authentication to Remedy server (if required).
- **First Name**: First name of the Remedy ITSM user which will be the Customer (initiator) of the incident.
- **Last Name**: Last name of the Remedy ITSM user which will be the Customer (initiator) of the incident.

Note: The first and last name fields represent the incident reporter (owner) in Remedy, and must match an existing Remedy user.

- **Impact**: Impact level for the incident. Options include 1-Extensive/Widespread, 2-Significant/Large, 3-Moderate/Limited, 4-Minor/Localized.
- **Urgency**: Urgency level of the incident.
- **Incident Summary Template**: Summary of the incident (title). Can use place holders.
- **Incident Notes Template**: Notes of the incident (detailed description). Can use place holders.
- **CI Name** (optional): Together with Lookup Keyword (below), this represents a configuration item that is pre-configured on the Remedy server. The CI is associated to all incidents opened by this action interface.
- **Lookup Keyword** (optional): This is the class type of the CI specified in the setting above.

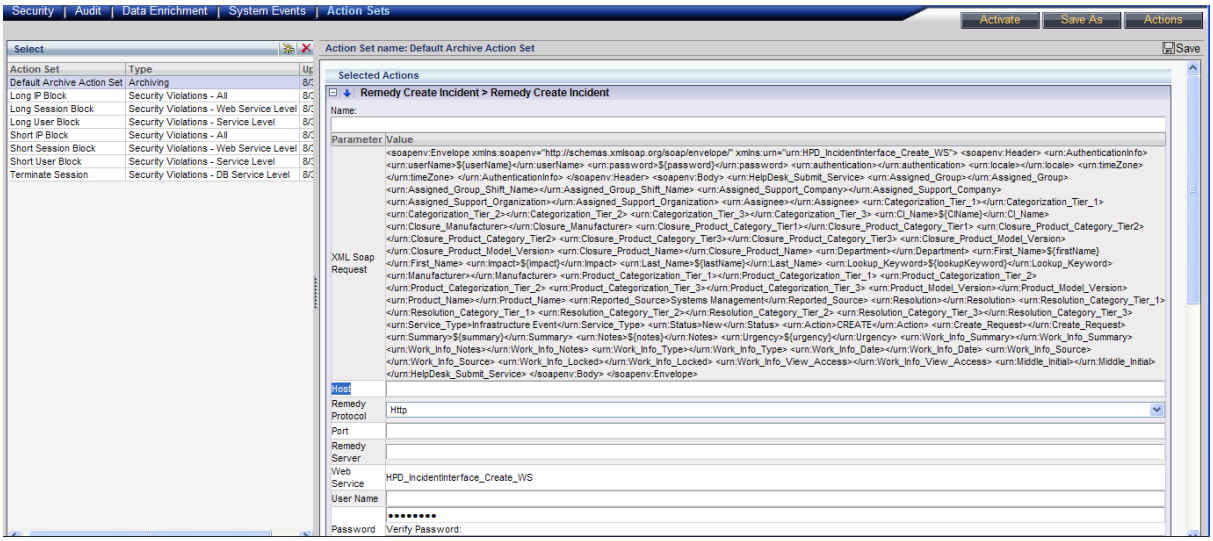


Figure 66: Example: Action Set Configured for Integration with Remedy BMC

6. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For more information, see Activating Settings on page 75.

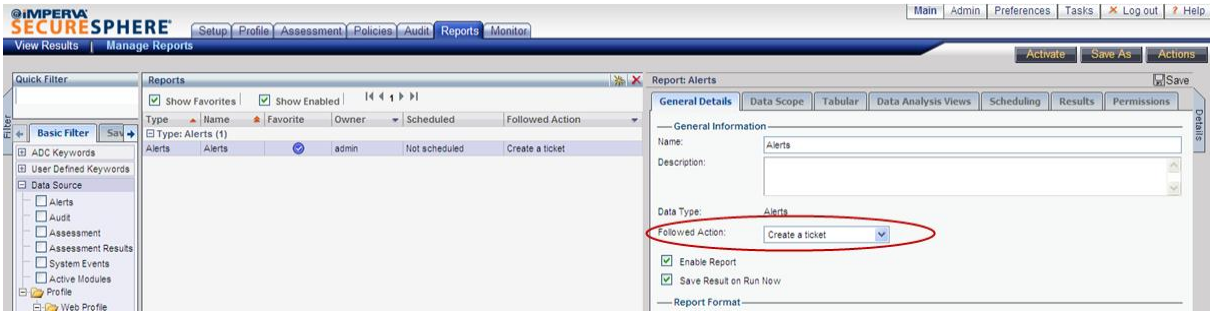
Configuring Policy or Report with Followed Action for Remedy

This procedure describes how to attach an action set as a followed action to open a Remedy ticket from SecureSphere. Followed actions can be attached to a security event such as a violation taking place, then can also be attached to a report to assign the report to a user through Remedy who is responsible for reviewing the report.

This example attaches a followed action to an existing report.

To configure SecureSphere for integration with Remedy:

1. Create a new Action Set with an intuitive name. For example, Remedy Create Incident as described in Configuring SecureSphere for Integration with BMC Remedy Service Desk on page 396.
2. In the **Main** workspace, click **Reports**. The reports window is displayed.
3. Configure a new report, or navigate to an existing report to which to attach the Remedy followed action. For example, you can attach the action set as a followed action to an Alert report, so that every time the report is generated, a ticket is open in remedy to review the report.
4. In the **General Details** tab of the details pane, select the **Create a Ticket** action set from the Followed Action dropdown menu configured in step 1, as shown in the following illustration.



5. Click Save. Settings are saved and the action interface has been attached to the report. Every time this report is generated, it will open a new ticket in Remedy, as illustrated in the following diagram.

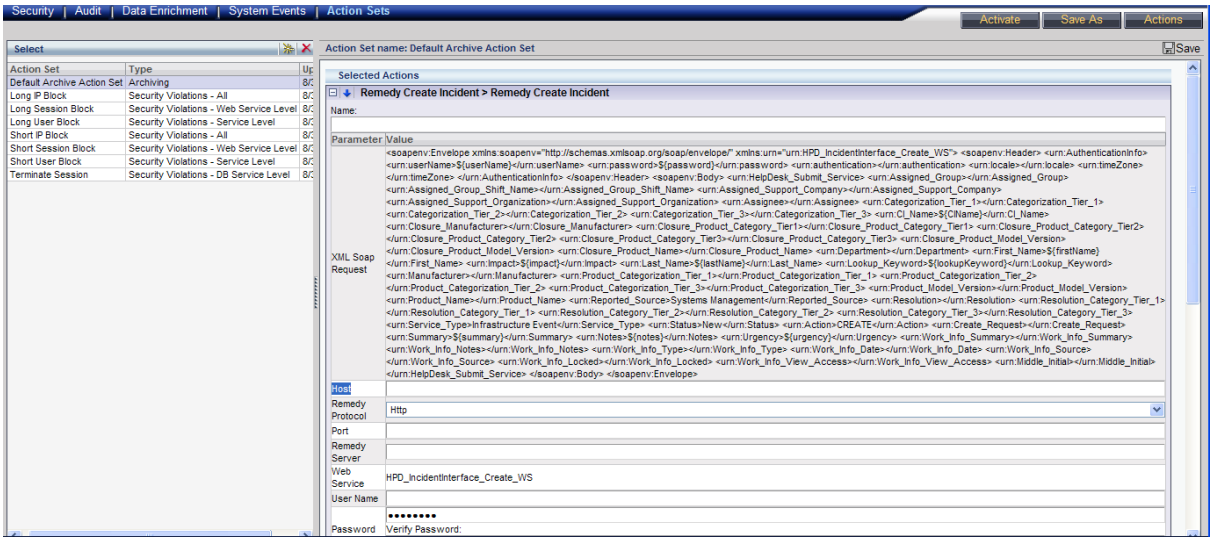


Figure 67: Example: Action Set Configured for Integration with Remedy BMC

Integrating with RSA enVision

The RSA enVision platform provides an effective security and information event management (SIEM) and log management solution, capable of collecting and analyzing large amounts of data in real-time, from any event source and in computing environments of any size.

Supported RSA details include the following:

- **Version**: 3.7 and later
- **Collection Method**: Syslog
- **Service**: NIC Collector Service

Configuring Integration with RSA enVision

The RSA enVision platform supports the Imperva SecureSphere event source. There are three main use cases for which SecureSphere can be configured to integrate with enVision. They include the following:

Action	Description	For more information, see...
Configure Security Alert integration	Configure SecureSphere to send **security alerts** to enVision using a syslog interface.	Configuring SecureSphere to Send Security Alerts to enVision on page 399
Configure System Event integration	Configure SecureSphere to send **system events** to enVision using a syslog interface.	Configuring SecureSphere to Send System Events to enVision on page 401

Configuring SecureSphere to Send Security Alerts to enVision

The following procedure describes how to integrate SecureSphere with enVision for the purpose of sending security alerts generated in SecureSphere to enVision.

To configure SecureSphere with RSA enVision:

1. In the **Main** workspace, select **Policies** > **Action Sets**. The **Action Sets** window appears.
2. In the **Action Sets** window, click **Create New** . The **Action Sets** dialog box appears.
3. In the **Name** field, type **enVision Alerts**.
4. From the **Apply to event type dropdown list,** select **Any Event Type**.
5. Click **Create**. The **Create New** dialog box closes and a new item appears in the **Select** table.
6. Select the **enVision Alerts** action set and move the **System Log > Log security event to RSA enVision** action from the **Available Action Interfaces** to the **Selected Actions** by clicking the green arrow next to the action.
7. Expand **System Log > Log security event to RSA enVision** and set shown in the table **RSA enVision Alert Options** below.
8. Click **Save**.
9. Click **Policies** > **Security**. The **Security Policies** window appears.
10. Create a new policy or select an existing policy to which you want to apply the new action set and select this action set from the Followed Action drop-down list. For more information on using action sets as following actions in policies, see Assigning Followed Actions on page 341.
11. Click **Save**. Integration has been configured and all security alerts that match the configured policy will be sent via syslog to enVision.

RSA enVision Alert Options

Parameter	Description
Name	Type **enVision Alerts**
Syslog Host	Type the IP address of your enVision system.
Syslog Host Level	If not configured, select **Info**
Message	**Note:** This field is for reference, it should be available by default when selecting the indicated action interface. If not, copy and paste the following text, removing all line break characters. `Imperva,alert#=$!{Alert.dn},event#=$!{Event.dn},createTime=$!{Alert.createTime},updateTime=$!{Alert.lastUpdateTime},alertSev=$!{Alert.severity},group=$!{Alert.serverGroupName},ruleName="$!{Alert.alertMetadata.alertName}",evntDesc="$!{Alert.description}",category=$!{Alert.alertType},disposition=$!{Alert.immediateAction},eventType=$!{Event.eventType},proto=$!{Event.sourceInfo.ipProtocol},srcPort=$!{Event.sourceInfo.sourcePort},srcIP=$!{Event.sourceInfo.sourceIp},dstPort=$!{Event.destInfo.serverPort},dstIP=$!{Event.destInfo.serverIp},policyName="$!{Rule.parent.displayName}",occurrences=$!{Alert.aggregationInfo.occurrences},httpHost=$!{Event.struct.httpRequest.url.host},webMethod=$!{Event.struct.httpRequest.url.method},url="$!{Event.struct.httpRequest.url.path}",webQuery="$!{Event.struct.httpRequest.url.queryString}",soapAction=$!{Event.struct.httpRequest.soapAction.soapAction},resultCode=$!{Event.struct.httpResponse.responseCode},sessionID=$!{Event.struct.webCorrelationInfo.sessionId},username=$!{Alert.username},addUsername=$!{Event.struct.user.additionalUser},responseTime=$!{Event.struct.responseTime},responseSize=$!{Event.struct.responseSize},direction=$!{Event.struct.networkDirection},dbUsername=$!{Event.struct.rawUser.rawUser},queryGroup=$!{Event.struct.queryGroup.queryGroup},application="$!{Event.struct.application.application}",srcHost=$!{Event.struct.host.host},osUsername=$!{Event.struct.osUser.osUser},schemaName=$!{Event.struct.databases.schemaName},dbName=$!{Event.struct.databases.databaseName},hdrName=$!{Event.struct.httpRequest.headers.name},action="$!{Event.struct.query.parsedQuery}"`
Facility	Select **Syslog**
Run on Every Event	In not enabled, select Run on Every Event

Configuring SecureSphere to Send System Events to enVision

The following procedure describes how to integrate SecureSphere with enVision for the purpose of sending system events generated in SecureSphere to enVision.

To configure SecureSphere audit records to be sent to RSA enVision:

1. In the **Main** workspace, select **Policies** > **Action Sets**. The **Action Sets** window appears.
2. In the **Action Sets** window, click **Create New** . The **Action Sets** dialog box appears.
3. In the **Name** field, type **enVision Database Activity Monitoring**.
4. In the **Apply to event dropdown list,** select **Audit**.
5. Click **Create**. The **Create New** dialog box closes and a new item appears in the **Select** table.
6. Select the **enVision Events** action set and move the **System Log > Log system event to RSA enVision** action from the **Available Action Interfaces** to the **Selected Actions** by clicking the green arrow next to the action.
7. Expand **System Log > Log system event to RSA enVision** and configure as shown in the table **RSA enVision System Event Options** below.
8. Click **Save**.
9. Click **Policies** > **Security**. The **Security Policies** window appears.
10. Create a new policy or select an existing policy to which you want to apply the new action set and select this action set from the Followed Action drop-down list. For more information on configuring System Event policies using action sets as following actions in policies, see Assigning Followed Actions on page 341
11. Click **Save**. Integration has been configured and all system events that match the configured policy will be sent via syslog to enVision.

RSA enVision System Event Options

Parameter	Value
Name	Type **enVision Events**
Primary Host	Type the IP address of your enVision system.
Syslog Host Level	If not configured, select **Info**
Message	**Note:** This field is for reference, it should be available by default when selecting the indicated action interface. If not, copy and paste the following text, removing all line break characters. `Imperva,event#=$!{Event.dn},createTime=$!{Event.createTime},eventType=$!{Event.eventType},eventSev=$!{Event.severity},username=$!{Event.username},subsystem=$!{Event.subsystem},message="$!{Event.message}"`
Facility	Syslog
Run on Every Event	In not enabled, select Run on Every Event

Configuring SecureSphere to Send Audit Records to enVision

The following procedure describes how to integrate SecureSphere with enVision for the purpose of sending audit records from SecureSphere to enVision.

1. In the **Main** workspace, select **Policies** > **Action Sets**. The **Action Sets** window appears.
2. In the **Select** pane, click . The **Action Set** dialog box appears.
3. In the **Name** field, type **enVision Database Activity Monitoring**.

4. In the **Apply to event dropdown list,** select **Audit**, then click **Create**. The new Action Set appears in the **Action Sets** pane and is selected.
5. Select the **enVision Alerts** action set and move the **System Log > Log Audit events to System Log (Gateway syslog)** action from the **Available Action Interfaces** to the **Selected Actions** by clicking the green arrow next to the action.
6. Expand **System Log > Log Audit events to System Log (Gateway syslog)** and configure in the table **RSA enVision Log Audit Events Options** below.
7. Click **Save**.
8. Click **Policies > Audit**. The Audit window appears.
9. Create a new policy or select an existing policy to which you want to apply the new action set and select this action set from the Followed Action drop-down list. For more information on configuring Audit policies, see Configuring an Audit Policy.
10. In the **Followed Action** tab, select the name of the newly created action set, enVision Events.
11. Click **Save**. Integration has been configured and all audit records that match the configured policy will be sent via syslog to enVision.

RSA enVision Log Audit Events Options

Parameter	Description
Name	enVision Database Activity Monitoring
Primary Host	Type the IP address of your enVision system.
Primary Port	Type the envision port number (typically 514)
Syslog Host Level	Info
Message	**Note:** Copy and paste the following text, removing all line break characters. `IMPERVAImperva,dstIP=$!{Event.destInfo.serverIp},dstPort=$!{Event.destInfo.serverPort},dbUsername=$!{Event.struct.user.user},srcIP=$!{Event.sourceInfo.sourceIp},srcPort=$!{Event.sourceInfo.sourcePort},creatTime=$!{Event.createTime},srvGroup=$!{Event.serverGroup},service=$!{Event.serviceName},appName=$!{Event.applicationName},event#=$!{Event.dn},eventType=$!{Event.struct.eventType},usrGroup=$!{Event.struct.userGroup},usrAuth=$!{Event.struct.user.authenticated},application="$!{Event.struct.application.application}",osUsername=$!{Event.struct.osUser.osUser},srcHost=$!{Event.struct.host.host},dbName=$!{Event.struct.databases.databaseName},schemaName=$!{Event.struct.databases.schemaName},bindVar=$!{Event.struct.rawData.bindVariables},sqlError=$!{Event.struct.complete.errorValue},respSize=$!{Event.struct.complete.responseSize},respTime=$!{Event.struct.complete.responseTime},affRows=$!{Event.struct.query.affectedRows},action="$!{Event.struct.query.parsedQuery}",rawQuery="$!{Event.struct.rawData.rawData}"`
Facility	Syslog

Configuring Integration with Active Directory for User Management

You can configure SecureSphere to manage users and roles from an external central location - Active Directory. Integration with Active Directory provides central management and requires low maintenance of user permissions. All the changes in user definitions are performed in the external system, Active Directory, without the need to log into SecureSphere.

Active Directory users and groups are represented in SecureSphere as follows:

- **External Roles**: Represents a group of users defined in Active Directory. In SecureSphere these roles are mapped to the roles in external systems and can receive SecureSphere roles and permissions, similarly to SecureSphere users. For more information on external roles see Creating an External Role on page 408
- **External Users**: A user that is defined in Active Directory and in SecureSphere this user is created automatically during user's first login. Such user cannot be deleted or assigned roles in SecureSphere, only in the external system. External users can receive SecureSphere permissions for specific objects in the system, granted explicitly by Administrator, or automatically when the user creates objects. For more information on external users see Viewing External Users on page 410.

In addition, you can configure integration with Active Directory to enrich your data, see Configuring Active Directory Integration for Data Enrichment on page 360.

The process of working with external users includes the following steps:

#	Action	Description	For more information, see...
1	**Define the external system**	The only external system that is currently supported for the authorization procedure is LDAP Authorization & Authentication	External System - LDAP Authentication and Authorization on page 405
2	**Configure authentication and authorization for the external system you have defined**		Authentication and Authorization Configuration for Active Directory on page 404
3	**Perform Active Directory settings**	Create groups in LDAP for which you want to define external roles in SecureSphere. Assign the users to the LDAP groups for which you want to access SecureSphere	
4	**Create External Roles in SecureSphere**	The External Role name must be identical to the name of the Active Directory group it represents.	Creating an External Role on page 408
5	**Configure External roles**	Assign SecureSphere roles and grant the permissions to external roles.	Assigning Roles to External Roles on page 409
6	**Log in to SecureSphere**	Log in to SecureSphere using Active Directory credentials	Logging Into SecureSphere in the Authorization Mode on page 410
7	**View External users**	External users appear with a list of external roles, as defined in Active Directory. The External Roles list is read-only.	

Authentication and Authorization Configuration for Active Directory

The **Authentication and Authorization** configuration includes the following two parts:

- In the **Authentication** section, add username and password to identify the user.
- In the **Authorization** section, define the process of granting roles and permissions for SecureSphere/external system users.

To configure user authentication:

1. In the **Admin** workspace, select **System Definitions** > **Authentication & Authorization Configuration**. The **Authentication & Authorization Configuration** pane appears.
2. For **User Authentication**, select one of the following options:
   - **SecureSphere**: Allows authenticating using the internal SecureSphere database only.
   - **External**: Allows authenticating using external authentication systems. If you select this option, you must select the external system from the **External Authentication System** dropdown list. The external systems that you can choose from are defined in the **External Systems** pane, see Defining External Systems on page 404.
   - **User Specific**: Allows authenticating using SecureSphere database or external authentication systems. If you select this option, you must select the external system from the **External Authentication System** dropdown list. The external systems that you can choose from are defined in the **External Systems** pane, Defining External Systems on page 404.
3. To enable managing SecureSphere users in LDAP, select **External Authorization**. The LDAP group permissions are reflected in the SecureSphere external role permissions. For **External Authorization System**, select an external system from the dropdown list.

   For information about external systems, see Defining External Systems on page 404.



Notes:

- When de-selecting the External Authorization option, all the external users are moved to the regular users section in the **Users & Permissions** window. They maintain their external system authentication, but lose their external role assignment.
- When selecting the **External Authorization** option, all Users with external system authentication become externally authorized: they are moved to the External Users section in the Users & Permissions window, and their permissions are based on their External Roles.

4. Click **Save**.

Defining External Systems

You can create a new External System, or edit and delete an existing one.

To create a new External System:

1. In the **Admin** workspace, click **System Definitions**.
2. In the **System Definitions** pane, select **External Systems**.
3. In the **External System** pane, click . The **External Systems** window appears.
4. In the **Name** text box, enter the name for the external system.
5. From the **Type** drop-down list, select **LDAP Authentication & Authorization**.
6. If you change any of these parameters, click **Save**.

   If you are in delayed activation mode, you must activate these settings before they take effect. For more information, see the **SecureSphere User Guide**.

To edit an existing External System:

1. In the **Admin** workspace, click **System Definitions**.
2. In the **System Definitions** pane, select **External Systems**.
3. In the **External System** pane, select an External System from the list.

4. Expand the External System by clicking on the plus-sign to the left of its name.
5. Edit the External System's parameters.
   See External Systems - LDAP on page 405 for an explanation of External System parameters.
6. If you change any of these parameters, click **Save**.
   If you are in delayed activation mode, you must activate these settings before they take effect. For more information, see the **SecureSphere User Guide.**

To delete an existing External System:

1. In the **Admin** workspace, click **System Definitions**.
2. In the **System Definitions** pane, select **External Systems**.
3. In the **External System** pane, select an External System from the list.
4. Click .
5. Click **Save**.
   If you are in delayed activation mode, you must activate these settings before they take effect. For more information, see the **SecureSphere User Guide**.

External Systems - LDAP

You can define LDAP external systems. The connection can be established once this external system is enabled.

LDAP configuration Parameters

Parameter	Description
Name	The external system's name.
Primary Server	The IPv4 or IPv6 address or the host name of the primary external system.
Secondary Server	The IPv4 or IPv6 address or the host name of the secondary external system.
Use SSL	Specifies whether to connect to the external system using SSL.
Port	The port number on the external system.
Account Name DN	The account name or DN (Distinguished Name) under which to connect to the external system.
Password	The password to use when connecting to the external system.
Verify Password	Re-enter the password.
Base DN	The Base Distinguished Name is the domain account name in the LDAP server's navigation tree.
Follow Referrals	Select this option to tell/direct the LDAP system to follow referrals.

Click **Test Connection** to verify that the connection to the external system is functioning properly.

External System - LDAP Authentication and Authorization

You can set the parameters related to LDAP authentication & authorization of SecureSphere users.

The connection can be established once this external system is enabled.

If you change any of these parameters, click **Save**. If you are in delayed activation mode, you must activate these settings before they take effect. For more information, see the **SecureSphere User Guide**.



Note:

- The admin user cannot be authenticated using LDAP, but is always authenticated by SecureSphere.
- When configuring the external system using a hostname, and not an IP address: In order to use LDAP to authenticate SecureSphere users, a DNS client must be enabled on the management server using the `impcfg` command. For more information, see Name Resolution (DNS client).

LDAP Authentication Window Parameters

Parameter	Description
Connection Settings	
Name	The external system's name.
Primary LDAP Server Host	The IP address or the host name of the primary LDAP server.
Port	The port number of the LDAP service on the primary LDAP server Default: 389
Secondary LDAP Server Host	The IP address or the host name of the secondary LDAP server.
Port	The port number of the LDAP service on the secondary LDAP server.
Use SSL	Select to specify that communication with LDAP servers is encrypted. If this option is selected, ensure that the port number is changed accordingly. **Upload Certificate**: Enables uploading ssl certificates and subsequently monitoring traffic that is encrypted. Supports .cer certificates. **Validate** Certificate: Verifies that the certificate is valid.
Access Mode These parameters specify whether access to the LDAP server is anonymous or conducted under a specific user. Select one of the following:	
Anonymous	Use the "guest" account on the LDAP server, if it is configured.
User Account	If **User Account** is selected, then the following parameters must be defined: • **Account DN**: The Distinguished Name of the domain account under which the LDAP server is to be accessed. • **Password**: The domain account's password. • **Verify Password**: Re-enter the password.
Search Settings The following parameters define how to locate a user in LDAP.	
Identifier Field	Enter the unique identifier that SecureSphere uses to identify the external LDAP users. If **Identifier Field** is empty, the user name in SecureSphere is the same as on the LDAP server. **Note:** Case-Sensitivity: Depending on the LDAP server, the login process (authentication by sam-name/email, etc.) may or may not be case-sensitive. However regarding **Identifier Field** vs. name, the comparison in SecureSphere is case-sensitive (this is a cross-system behavior). For example if the user CN in LDAP is "John Smith", you must set the SecureSphere user name to "John Smith" not "john smith". Default: CN.
Base Path for Users	The domain path in the authentication server's navigation tree where the LDAP users are stored. **Note:** You can set this parameter only when the External Authentication parameter is enabled, see step 3.

Parameter	Description
Search Filter	The search criteria for the unique identifier. **Note:** If you change the search filter in your authentication server's account (for example, from username to email) users will then have to use a different login name.
Users Search Scope	Select one of the following: • **One Level**: Limits the search to the base path only. • **Subtree:** Allow searching subtree branches.
Authorization Settings	
Enable	Enables reflecting LDAP permission group structure in SecureSphere, where LDAP groups are mapped to SecureSphere external roles.
Base Path for LDAP Groups	The domain path in the authentication server's navigation tree where the LDAP groups are stored. **Note:** You can set this parameter only when the External Authentication parameter is enabled, see step 3.
Groups Search Scope	Select one of the following: • **One Level**: Limits the search to the base path only. • **Subtree:** Allow searching subtree branches.

Click **Test Connection** to verify that the connection to the LDAP servers is functioning properly.



Note: To point SecureSphere to the Active Directory Global Catalog, set **Base Path** to the domain root (for example, DC=corpmain,DC=com) and **Port** to 3268.

To view or modify the LDAP authentication configuration for individual users:

1. In the Admin workspace, click **Users & Permissions**.
2. In the **List** pane, select a user.
3. Click the **Details** tab.
4. To specify that this user is to be authenticated externally, select **External** under **Authenticator**.
5. If you want to check if this user exists on the authentication server, click **User Lookup**.
6. The **LDAP User Lookup** dialog box appears.
7. Type the username of the LDAP user to search for and click **Search**. The **Test User** progress bar appears.

Creating an External Role

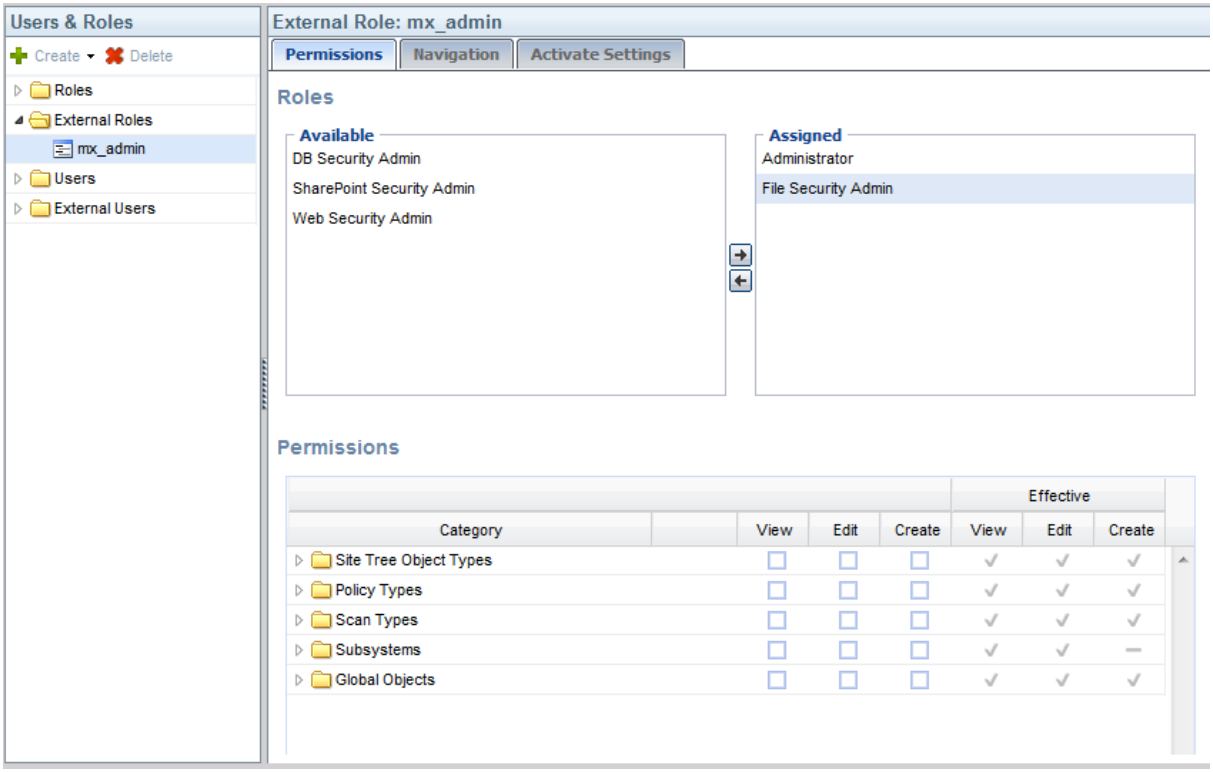
Before creating External roles you need to make sure you have defined in Active Directory the groups for which you want to create external roles in SecureSphere and assigned users to your Active Directory groups. The users assigned to the groups in Active Directory are the users that will access SecureSphere and appear in SecureSphere as External Users.

To create an External Role:

1. In the **Admin** workspace, select Users and Permissions.
2. In the **Users & Roles** pane, click , then **Create New External Role**. The **Create New External Role** window appears.
3. Type a **Name** for the external role that is the name of the group in LDAP.

Note: The name is not case sensitive.

4. To check if there is a role in LDAP, click **External Role Lookup**.
5. Optional: In the **Assigned Roles** section, move the role you want to assign the external role from the **Available** pane to the **Assigned** pane.
6. Click **Create**. The new external role is created and appears under **External Roles** in the **Users and Roles** pane to the left-hand side of the window.



You now need to configure this role with the desired SecureSphere permissions by selecting the SecureSphere roles that will be assigned to it, see Assigning Roles to External Roles on page 409.

Assigning Roles to External Roles

SecureSphere enables you to assign roles to External Roles. By assigning roles to External Roles you grant them the permissions that have been configured for the assigned roles.

To assign a role to external roles:

1. In the **Admin** workspace, select Users and Permissions.
2. In the **Users & Roles** pane, expand the **External Roles** category, then select the external role you want to assign a new role. That external role details are displayed in the details pane.
3. On the permissions tab, move the role you want to assign the external role from the **Available** pane to the **Assigned** pane.

Note: For information on creating new roles, see Creating an External Role on page 408.

4. Click **Save**. Settings are saved and the user has now been granted the permissions available in the assigned role.

Logging Into SecureSphere in the Authorization Mode

Once external roles have been created, external system users that were not defined as SecureSphere users can login to SecureSphere. In that case all the Active Directory users that are members of Active Directory group for which you define external roles can login into SecureSphere.

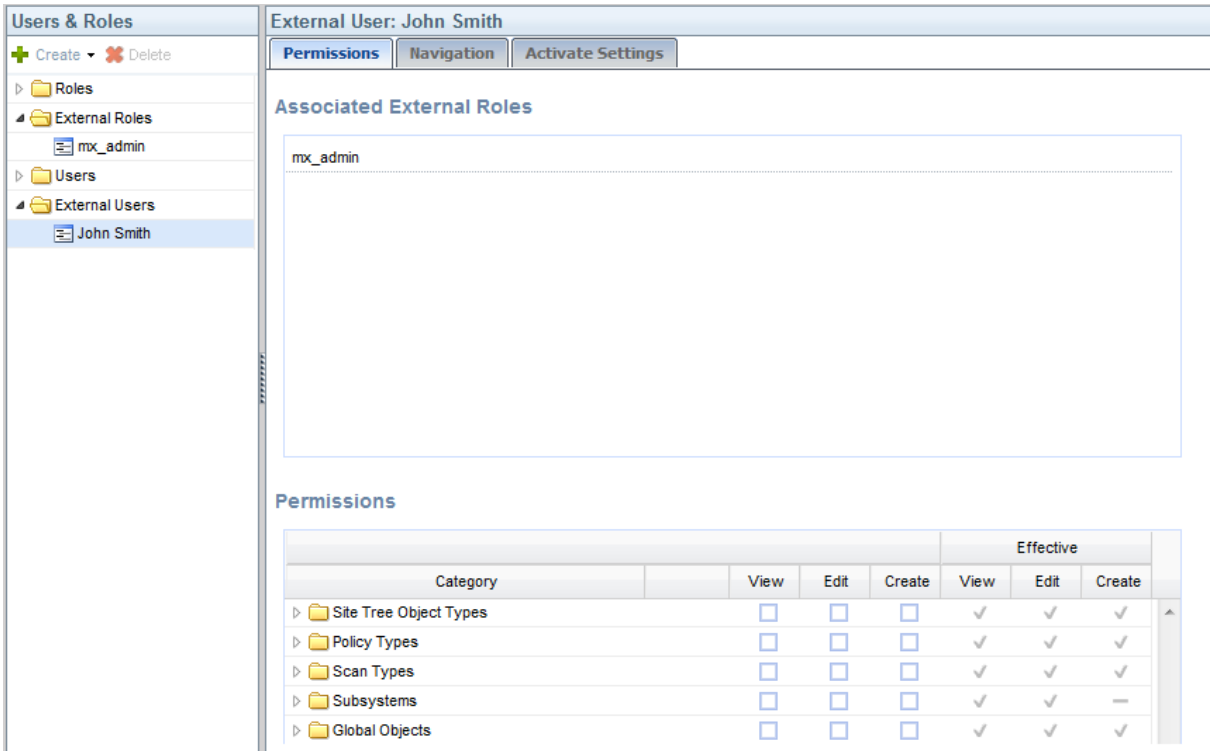
When such an Active Directory user attempts to login, SecureSphere connects to Active Directory and searches for that user. If this user is authenticated in Active Directory and there is an external role in SecureSphere for that Active Directory user group, then this user is successfully logged into SecureSphere with the Active Directory user name and is added as an External User. This user receives all the permissions granted to all matching external roles. This configuration is consistent throughout the Active Directory - SecureSphere session.

Note: External users are assigned to external roles in SecureSphere that reflect Active Directory groups as follows: 1- When the Active Directory user is the member of the Active Directory group; 2- When an Active Directory group is not assigned to the user, but has a nested group which is assigned to the user within Active Directory.

SecureSphere user name is defined according to the Identifier Field that was defined during the Authentication process. For more information, see External Systems - LDAP on page 405.

Viewing External Users

- Once you have created an External Role, you can view External Users that logged into SecureSphere, see Logging Into SecureSphere in the Authorization Mode on page 410. The users that you view here are Active Directory group users for which you have created the External Roles.
- The Roles pane within the External Users window is read only.



External Roles and Users Daily Updates

SecureSphere provides automatic daily updates from the Active Directory server for configuration validation and automatic error fixes, as follows:

- To delete external roles that no longer exist in Active Directory.
- To delete external users that no longer exist in Active Directory.
- To update external roles for each external user.
- To update name changes in external users and roles.



Note: The user that was deleted from Active Directory may still appear in the External Users list in SecureSphere after deletion until the daily job runs. Likewise, changes in a user's roles will not be updated until the user login or daily job.

System Event Policies

System event policies define what SecureSphere is to do when certain system events occur. You can define for example, that an email be sent to the system administrator whenever a certain gateway comes up, but not when other gateways come up. This section reviews the following subjects about System Event Policies:

- Working with System Event Policies on page 411
- Creating a System Events Policy on page 411



Notes:

- For information on archiving system events, see the Administration Guide or speak to your SecureSphere administrator.
- For more information on system events, see Managing System Events on page 107.

Working with System Event Policies

To work with system events policies, perform the following tasks:

1. Define an action set of **Type** "system event" or "Any event type".

   These are the actions to be performed when the system event takes place. You an create a new action set or use an existing one. For information on action sets, see Working with Action Sets and Followed Actions on page 326.
2. Create a system events policy and define the policy parameters.

Creating a System Events Policy

To create a System Events policy:

1. In the **Main** workspace, click **Policies** > **System Events**.
2. In the middle pane, click .
3. In the **Create New Policy** window, enter **Name** and select the **Type** of system event.

   It is recommended that you give the policy a meaningful name.
4. Click **Create**.
5. In the **Policy Details** tab, the information you entered in the **Create New Policy** window is displayed, and you can modify this data.

- **Display Name** is the name you entered in the **Create New Policy** window.
- **Matching Text Segments** restricts the policy to apply only to system events of the given type which contain the specified text. A complete list of system events and their matching text segments can be found in About Event Types on page 547.

  If you leave **Matching Text Segments** empty, the policy will be invoked for all events.

6. In the **Followed Action** tab, select a **Followed Action**.

   Followed Action is an action set.

   No matter what you set **Matching Text Segments** and **Followed Action** to, the system event is always written to the system log and always appears in the SecureSphere Dashboard. The system events policy only relates to what happens in addition.
7. Click **Save**.

EXAMPLE: - CREATING A SYSTEM EVENTS POLICY

In this example, a systems event policy will be created which sends an e-mail to the system administrator whenever the user Joe user logs in to SecureSphere.

1. In the **Main** workspace, select **Policies** > **Action Sets**.
2. In the left pane, click .
3. In the **Action Set** window, enter a **Name**, for example, "e-mail sysadmin".
4. Under **Apply to event type**, select **System events**.

   The only other option for action sets that will be used in system events policies is **Any event type**.
5. Click **Create**.
6. Configure the action set to send an e-mail to the sysadmin by selecting and configuring the appropriate action interface in the middle pane.

   For information on how to create and configure action sets, see Working with Action Sets and Followed Actions on page 326.
7. Click **Save**.
8. In the **Main** workspace, click **Policies** > **System Events**.
9. In the middle pane, click .
10. In the **Create New Policy** window, enter **Name**, for example, "when Joe logs in".
11. Under **Type**, select **User logged in**.
12. Click **Create**.
13. In the middle pane, select the policy you just created.
14. In the **Policy Details** tab, under **Matching Text Segments**, enter "Joe".

    This restricts the policy only to apply only to system events of the given **Type** (in this case **User logged in**) which contain the text "Joe".
15. In the **Followed Action** tab, select the action set created at the beginning of this procedure, "email sysadmin".
16. Click **Save**.

The figure below shows the login by Joe, as it appears in the SecureSphere Dashboard.

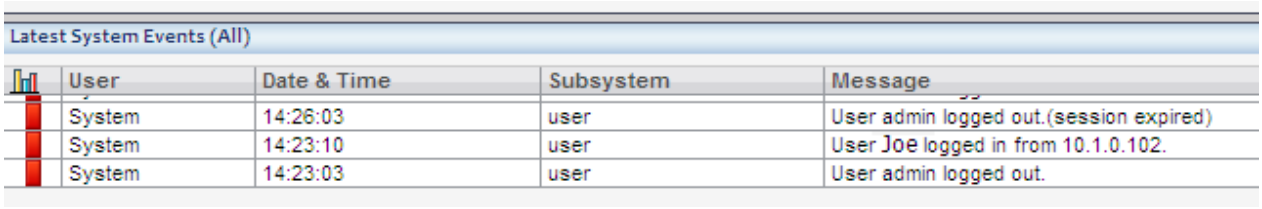


Figure 68: Login from 10.1.0.102

The figure below shows the content of the email that was sent as a result of the system event policy triggering its Followed Action.

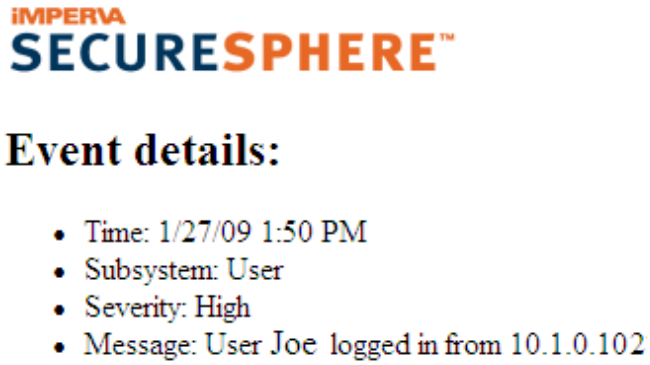


Figure 69: Email Sent as a Result of the System Event Policy

Configuring Scheduling

SecureSphere enables you to configure scheduling for a number of different aspects of its operation. For example, you can configure reports or assessment policies to run on a regular schedule. The locations which enable you to configure scheduling include:

- When scheduling a server discovery policy
- When configuring an assessment job
- When scheduling a web scanner integration policy
- When scheduling the update of ThreatRadar services
- When scheduling a report
- When scheduling the update of ADC content

To configure Scheduling:

1. Access scheduling from the relevant location.You have the options described in the table **Scheduling Options** below.
2. In **Starting from**: Click the Schedule icon, then select a date.
3. From the **At Time** dropdown menu: Select a time from the dropdown menu. Alternatively you can type in a time.
4. Click **Save**. Settings are saved. If you are in delayed activation mode, you may need to activate these settings. For more information, see Activating Settings on page 75.

Scheduling Options

Option	Description
None	No job is scheduled.
Once	Schedules a job to be executed a single time in the future.
Recurring	Schedules a job to be executed on a regular basis. You have three options: **Daily**: Schedules the item to be run on a regular basis. Enter the number of days you want between runs. **Weekly**: Schedules the item to be run on a weekly basis. Enter the number of weeks you want between runs, and select one or more days of the week on which to run the report. **Monthly**: Schedules the item to be run each X month. Configure one of the following: • **Day**: Schedules the item to be run on a specific day of the configured months. Enter a day and the number of months you want between execution. • **Day of the Week**: Schedules the item to be run on a day of a week every X month. Select a chronological day (first, second, etc), a day of the week, and type the monthly interval. For example, the second Tuesday of every second month. What month this job is run depends on the item configured in step 3 below.

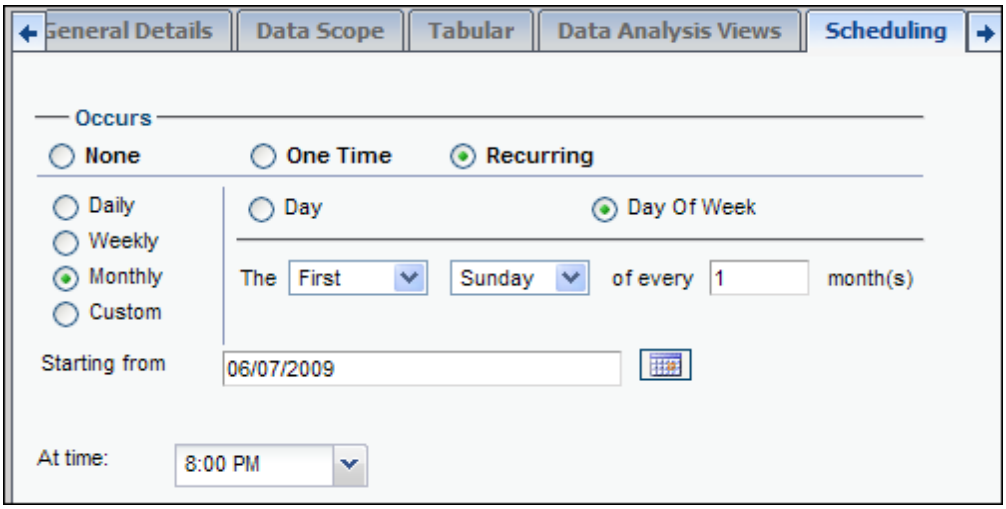
EXAMPLE: SCHEDULING A REPORT

Every month your department has a meeting which takes place the first Monday morning of the month. Your IT manager wants a report prepared for this meeting containing last month's activity.

You want the report to be generated before the morning meeting, so you schedule run it on Sunday evening 8:00 PM so it is ready when you come into the office first thing Monday morning.

1. In the **Main** workspace, choose **Reports > Manage Reports**. The **Manage Reports** window appears.
2. Select the report you want to configure with scheduling. That report's details appear in the **Details** pane.
3. Click the **Scheduling** tab.
4. In **Occurs**, select the **Recurring** option.
5. Select the **Monthly** option.
6. Select the **Day of the Week** option.
7. Select the **First Monday** of every **1** Month.
8. In **Starting From**, choose the current date. SecureSphere automatically configures scheduling to start the next Monday.
9. In the **At Time** dropdown menu, select **8:00 AM**.

10. Click **Save**. Settings are saved and the report has been scheduled.



Understanding the Execution Plan

The information displayed in the execution plan section of the Scheduling tab details which phases of discovery have been configured to run for the scan. It also identifies possible configurations errors such as missing credentials.

Customizing How SecureSphere Handles Web Traffic

SecureSphere provides a number of ways to include or exclude certain types of web traffic to monitor and protect including:

- **Web Service Restrictions**: Excludes static web items such as static parameters, soap elements and specific file types from being monitored. For more information, see Configuring Web Service Restrictions on page 416.
- **Source Restrictions**: Configures IP Groups and URL Prefixes that assist in focusing monitoring and protection or enables excluding these items from monitoring or protection. For more information, see Configuring Restrictions on page 418.
- **Web Error Pages and Policies:** Configures how SecureSphere handles errors with web traffic and enables you to configure custom error pages with useful information. For more information, see Configuring Web Error Page Policies on page 421.
- **Advanced Web Service Settings**: Configures advanced handling of web traffic items such as timeout and supported encoding options. For more information, see Configuring Advanced Web Service Settings on page 424.
- **OCSP Support**: Configures SecureSphere to support the parsing of the OCSP protocol. For more information, see Configuring OCSP Support on page 424.
- **Client Certificate Validation**: Configures SecureSphere to support the validation of client certificates using Certificate Authorities. For more information, see Validating SSL Client Certificates on page 436.

Configuring Web Service Restrictions

When working with web services, a number of data objects used in your data infrastructure may pose no actual threat to the network or data. SecureSphere enables you to exclude them from monitoring which eliminates alerts regarding these items and reduces operational SecureSphere overhead. The following table lists the main tasks that can be conducted to configure to restrict web traffic processing.

Configuring Web Restrictions Task Overview

	Action	Description	For more information, see...
1	**Configure Parameter Groups for exclusion (optional).**	Configure groups of parameters that you want to exclude from monitoring.	Configuring Parameter Groups on page 416.
2	**Configure Parameter Type Objects (optional).**	Configure parameter type objects for defining a required format for web traffic parameters.	Configuring Parameter Type Configuration Objects on page 417
3	**Configure SOAP elements for exclusion (optional).**	Configure SOAP parameters that you want to exclude from monitoring.	Configuring SOAP Elements on page 417.
4	**Configure file types for exclusion (optional).**	Configure file types by extension that you want to exclude from monitoring.	Configuring Static File Lists on page 417.
5	**Attach configured groups to a web service.**	Associate the configured groups as required to a SecureSphere site.	Associating Web Service Data Objects on page 418.

Configuring Parameter Groups

Parameter Groups define a list of parameters that are static and do not pose a threat to networks or data. By creating a list of these parameters, then configuring a web service to ignore them, you can eliminate any alerts that may have been generated, reducing SecureSphere administrator workload and gateway processing overhead.

To configure parameter groups:

1. In the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Objects** dropdown list, select **Parameters Groups**. Existing **Parameters Lists** are displayed.
3. In the **Globals Tree**, select an existing parameters list or create a new list as described in Creating New Global Objects on page 346. The selected parameters group list appears in the Configuration pane on the right-hand side of the screen.
4. In the **Configuration** pane, click  **New.** A new line is added to the bottom of the parameters list table.
5. Type the parameter to be added to the list.
6. Repeat steps 4 and 5 as required.
7. Click **Save**. The settings are saved.
8. Associate the parameter group with the web service as described in Associating Web Service Data Objects on page 418.

Configuring Parameter Type Configuration Objects

Parameter Type Configuration objects define lists of parameters and enables the creation of rules to enforce a complex format when parameters are received as part of web traffic.

To configure parameter type configuration objects:

1. In the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Objects** dropdown list, select **Parameters Groups**. Existing **Parameters Lists** are displayed.
3. In the **Globals Tree**, select an existing parameters list or create a new list as described in Creating New Global Objects on page 346. The selected parameters group list appears in the Configuration pane on the right-hand side of the screen.
4. In the **Configuration** pane, click  **New.** A new line is added to the bottom of the parameters list table.
5. Type the name for a parameter, then type a regular expression to define how the parameter should be formatted. For more information on regular expressions, see Regular Expressions on page 475.
6. Repeat steps 4 and 5 to add additional parameters if desired.
7. Click **Save**. The settings are saved.
8. Associate the parameter group with the web service as described in Editing a URL Parameter Value Type on page 260.

Configuring SOAP Elements

The SOAP Elements global object defines a list of SOAP elements that are static and do not pose a threat to networks or data. By creating a list of Soap elements, then configuring a web service to ignore them, you can eliminate any alerts that may have been generated reducing SecureSphere administrator workload and gateway processing overhead.

To configure SOAP elements:

1. In the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Objects** dropdown list, select **SOAP Elements Groups**. Existing **SOAP Elements Groups** are displayed.
3. In the **Globals Tree**, select an existing SOAP elements list or create a new list as described in Creating New Global Objects on page 346. The selected SOAP elements list appears in the **Configuration** pane on the right-hand side of the screen.
4. In the **Configuration** pane, click  **New.** A new line is added to the bottom of the SOAP Elements table.
5. Add the name of the **SOAP element** to be added to the list.



Note: Only type the specific element itself and not all of the XML. For example, if you have a SOAP that includes <xml> <action>4</action> </xml>, you would only add <action>4</action>.

6. Repeat steps 4 and 5 as required.
7. Click **Save**. The settings are saved.
8. Associate the parameter group to the web service as described in Associating Web Service Data Objects on page 418.

Configuring Static File Lists

A static file list defines a list of static file types that you can choose not to profile in a web application by selecting **Learn all URLs except for static URLs without parameters** under **Profiling URLs without Parameters** in the web application's **Definitions** tab. Though the file types are not learned in the profile, signatures and protection engines (XSS, SQLi *etc.*) are still applied to them.

To configure a Static File List:

1. In the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.

2. In the **Objects** dropdown list, select **File Type Groups**.
3. Select Default Static File List, or click **New** to create a new static file list.
4. In the **Details** pane, click **New.** A new line is added to the bottom of the table.
5. Type the **file extension** to be added to the list.
6. Repeat steps 4 and 5 as required.
7. Click **Save**. The settings are saved.
8. Associate the static file list with the web application as described in Web Application Definitions Parameters on page 71.

Associating Web Service Data Objects

Once data objects have been configured for restriction, they need to be associated with the web service from which you want to exclude them.

To associate a data object to a web service for exclusion:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. Click the Web Service to which you want to associate objects for exclusion. Web service details appear in the **Site** window.
3. Click the **Operation** tab. Operation options are displayed.
4. Under **Restrictions**:
   - To associate previously configured parameters, select the file from the **Ignore parameters** drop-down list.
   - To associate previously configured SOAP elements, select the file from the **Ignore SOAP elements** drop-down list.



Note: To ignore static files, select **Learn all URLs except for static URLs without parameters** under **Profiling URLs without Parameters** in the web application's **Definitions** tab. The file types defined in the static file list specified in **Static Files** in the web service's **Definitions** tab (by default **Global Objects** > **File Type Groups** > **Default Static File List**) are not learned in the profile.

5. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Configuring Restrictions

Restrictions enable you to exclude certain types of data from SecureSphere monitoring and protection based on their source, or alternatively monitor and protect only that type of data, while ignoring all other traffic. Two types of source restrictions can be configured. They include:

- **Based on IP Addresses**: Enables you to restrict monitoring and protection at the Web **Service** level, based on groups of IP addresses. For further information see Configuring IP Based Restrictions on page 419.
- **Based on URL Prefixes/Directories**: Enables you to restrict monitoring and protection at the Web **Application** level, based on a specific list of URL prefixes or directories. For further information see Configuring URL Prefixes / Directory Based Restrictions on page 420.

Configuring IP Based Restrictions

SecureSphere enables you to define a group of IP addresses then use it to either determine IP addresses that SecureSphere should ignore, or restrict its focus to only that specific set of IP addresses. For example, you can create an IP group that lists all IP addresses belonging to power users, then restrict SecureSphere monitoring traffic related to only this IP group.

The following table lists the main tasks that can be conducted to configure how to restrict supervision IP based traffic.

Configuring IP Based Restrictions Task Overview

	Action	Description	For more information, see...
1	**Create an IP Group (optional).**	Define a new IP group that defines traffic to restrict. Alternatively, you can use an existing IP group as is or add IP addresses.	Creating New Global Objects on page 346
2	**Add addresses to the IP Group (optional).**	Add addresses to an IP group to list the IPs you want to focus on or remove from supervision.	Configuring Source Restrictions on page 419.
3	**Modify source restriction configuration.**	Configure a site's source restriction options by choosing the required option and selecting an IP group.	Configuring Source Restrictions on page 419.

Configuring Source Restrictions

This procedure describes how to configure a SecureSphere site to ignore specific IP addresses, or limit supervision to these addresses.

To add or modify source restrictions:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. Click the server group you want to configure with restrictions. The server group is selected and its details appear in the **Sites** window.
3. Click the **Definitions** tab. Definitions options are displayed.
4. Configure a new IP group as described in Configuring IP Groups on page 347, or select an existing IP Group from one of the following drop-down menus:
   - **Ignore IP Group**: Selects a pre-configured IP group that belongs to the selected Server Group and excludes addresses from this IP group from being monitored (except for firewall violations).

     Notes:
     - This feature is not available with firewall rules.
     - This feature is not available in Reverse Proxy mode and appears as disabled. To alternatively configure ignoring of IP addresses when in Reverse Proxy mode, configure a policy with exceptions. For more information on Reverse Proxy mode, see Chapter 15, Reverse Proxies and the SecureSphere Administration Guide.
     - If one of the IP addresses in the IP group is that of a proxy (for example, a load balancer) then traffic forwarded by that proxy from other IP addresses is also ignored.
   - **Limit monitoring to IP group**: Limits SecureSphere monitoring to IP addresses belonging **only** to the selected group. Traffic for all other IPs will not be monitored. This can be useful before going into production by assisting to focus on traffic from a limited group of IP addresses, teaching SecureSphere standard behavior for this particular site.



Notes:

- For further information on source restrictions, see Configuring Web Service Restrictions on page 416.
- This feature is not available in Reverse Proxy mode and appears as disabled. For more information on Reverse Proxy mode, see Reverse Proxies on page 459 and SecureSphere Admin Guide.

5. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Configuring URL Prefixes / Directory Based Restrictions

URL Prefixes/Directory Groups enable you to define a URL prefix or directory and subsequently restrict monitoring to that group or exclude that group from monitoring.



Note: Excluding items using the URL prefixes / directory groups excludes them from being profiled, monitored and reported on. However, they will still generate alerts when found in signatures, protocol or firewall.

The following table lists the main tasks conducted to configure URL Prefixes and directories to restrict web traffic.

Configuring URL Prefixes/Directory Based Restrictions Task Overview

	Action	Description	For more information, see...
1	**Create a URL Prefixes/Directories Group.**	Create a URL Prefixes/Directories group that defines traffic to restrict.	Creating New Global Objects on page 346
2	**Configuring URL Prefixes/ Directory Group**	Configure the URL Prefixes / Directory group to match your requirements.	Configuring URL Prefixes/Directories Group on page 420
3	**Add addresses to the URL Prefixes/Directories Group.**	Add URLs, prefixes or directories you want to restrict or exclude supervision to the group.	Configuring Webworm Protection on page 426.
4	**Modify source restriction configuration.**	Configure a site's URL and directory based restrictions by choosing the required option and selecting an IP group.	Configuring Source Restrictions on page 419

Configuring URL Prefixes/Directories Group

This procedure describes how to add URL prefixes or directories to an existing group.

To add items to a URL Prefixes/Directory groups:

1. In in **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Objects** dropdown list, select **URL Prefixes/Directories Groups**. Existing **URL Prefixes/Directories Groups** are displayed.
3. In the **Globals Tree**, select an existing URL Prefixes/Directories group or create a new group as described in Creating New Global Objects on page 346. The selected URL Prefixes/Directories group appears in the pane on the right-hand side of the screen.
4. To add a new URL Prefix or directory to the group, in the **Details** pane, click  **New.** A new line is added to the **URL Prefix** table.

5. Type the item you want to be part of the group. You can type a full URL, URL prefix, or directory name which will subsequently include all items belonging to that URL prefix or directory name.
6. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.



Note: For information on how to add URL Prefixes / Directories groups to web applications in SecureSphere, see Adding URL Prefixes/Directories Groups to Web Applications on page 421.

EXAMPLE: CONFIGURING URL PREFIXES / DIRECTORIES

You have a website www.news.network.com and want to restrict monitoring and protection to a set of files in this site. The website has the following URLs:

- www.news.network.com/2006/articles/health/wheatgrass.htm
- www.news.network.com/2007/editorials
- www.news.network.com/images

If you add a full URL (www.news.network.com/2006/articles/health/wheatgrass.htm), only this URL is part of the group. If you add www.news.network.com/2006. All items in the 2006 directory are automatically included as part of the group. and if you type www.news.network.com/20, all URLs that contain that string (meaning for all years until 2099) would be included as part of the group.

Adding URL Prefixes/Directories Groups to Web Applications

This procedure describes how to configure a SecureSphere Web application to exclude traffic from specific URLs or directories on the application level, or restrict supervision to these URLs or directories.

To add or modify prefixes/directory groups:

1. Configure URL/Directory Groups as described in Configuring URL Prefixes/Directories Group on page 420.
2. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
3. In the **Sites Tree**, expand the relevant server group and service, then click on the web application you want to configure. Web application options appear in the right-hand side of the window.
4. Associate the URL Prefixes/Directories Group on the application level by selecting as follows:
   - **Restrict Monitoring to URLs**: Restricts SecureSphere monitoring and protection the URLs and directories defined in the attached URL Prefixes/Directories Group.
   - **Ignore URLs/Directories**: Excludes the URLs and directories defined in the attached URL Prefixes/Directories group from monitoring and protection. These items will be ignored by SecureSphere.
5. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Configuring Web Error Page Policies

Web error page policies are policies that enable you to determine what web error page is displayed as the result of a specific set of circumstances. This can be useful when wanting to use different error pages to be displayed to users based on specific events taking place in SecureSphere. The following table lists the main tasks that can be conducted to configure web error page polices and the rules that define their behavior.

Configuring Web Error Page Policies Task Overview

	Action	Description	For more information, see...

	Action	Description	For more information, see...
1	**Create a web error page.**	Define new error pages to used to warn users of web errors.	Creating New Global Objects on page 346
2	**Configure the web error page as required**	Configure a web error page with information and placeholders to provide users with helpful information.	Configuring Web Error Pages on page 422.
3	**Configure web error page policies.**	Configure web error page policies that determine when a particular web page is displayed to users.	Configuring Web Error Page Policies on page 423
4	**Attach web error page policy to a web service.**	Attach the configured web error page policy to a SecureSphere site.	Associating a Web Error Page Policy to a Web Service on page 423.

Configuring Web Error Pages

Web error pages are displayed to users when they try to access a forbidden page. They enable you to supply a meaningful error page and prevent data leakage of internal server information. Instead of receiving an error message such as **The system is not responding**, you can provide users with meaningful explanations and actionable information. Some examples of information you may want to include in web error pages include:

- What has happened, for example, a quarantine page saying: **You have been blocked for 1 hour**.
- What users can do about correcting the problem, for example: Contact Customer Care Error # 3582." This would enable customer care personnel to use the Event ID to identify the alert that caused the violation.
- A general message such as: **This URL is allowed for internal access only**.



Note: You may want to consider not using an error page. A lack of any response may prevent malicious users from being able to determine if a specific service is online.

Web error pages can be customized using html and placeholders, or can alternatively be a redirect which points users to an existing error page. Error pages needs to be small enough to fit into a single packet. Placeholders that can be incorporated into the error pages include $(SESSION_ID), $(HOST) and $(EVENT_ID).

To configure a web error page:

1. In in **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Objects** dropdown list, select **Web Error Page Groups**. Existing **Web Error Page Groups** are displayed.
3. In the **Globals Tree**, expand the **Web Error Pages** folder. Web error pages that have already been created are displayed.
4. Select an existing web error page or create a new web error page as described in Creating New Global Objects on page 346. The selected web error page appears in the pane on the right-hand side of the screen.
5. Select one of the following:
   - **Redirect**: Enables you to redirect users to any URL. Once this option is selected, you can enter a URL to an existing error page.
   - **Page**: Enables you to customize a default message using html and placeholders displayed when users encounter errors. For examples of how to configure redirect and HTML error pages, see Error Page Configuration Examples located in Configuring the Default Error Page on page 61.
6. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Configuring Web Error Page Policies

Web error page policies enable you to configure SecureSphere to display specific web error pages as the result of specific circumstances. Error page policies consist of one or more rules that determine when to display a specific web error page. You may want to display different error pages to provide meaningful information for certain situations.

To configure a web error page policy:

1. In the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. In the **Objects** dropdown list, select **Web Error Page Groups**. Existing **Web Error Page Groups** are displayed.
3. In the **Globals Tree**, expand the **Web Error Page Policies** folder. Web error page policies that have already been created are displayed.
4. Select an existing web error page policy or create a new web error page policy as described in Creating New Global Objects on page 346. The selected web error page policy appears in the Configuration pane on the right-hand side of the screen.
5. In the title bar of the **Configuration** pane, click  **New**. The **Create New Error Page Rule** is displayed.
6. Configure the following:
   - **Name**: Type a name for the rule.
   - **Priority**: Type a number that determines the priority of the rule when more than one rule is configured.
   - **Error Page**: Select an existing error page to be displayed when a rule is met. For information on configuring a web error page, see Configuring Web Error Pages on page 422.
7. Click **Save**. The new rule is added to the **Configuration** pane.
8. Click the **Rule Name** hyperlink in the rule you just created. The **Edit Rule** dialog box appears enabling you to configure rule criteria that determine the circumstances under which the web error page is displayed.
9. Click the green arrow  of the **Available Match Criteria** you want to add as part of the policy. The field is moved up to the **Match Criteria** pane.
10. Configure match criteria as required.
11. Repeat steps 8 and 9 as required.
12. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

EXAMPLE: CONFIGURING WEB ERROR PAGE POLICIES

You want to display a special error page to users when a webworm is encountered, a second error page when a profile violation occurs detecting an error with input, and a third error page for general purposes. You can create a error web page for when invalid input is entered, which states "Invalid input was detected. Your tracking ID is {$EventID}. Please contact support for further information." Then configure a web error page policy with the Violation criteria, while adding Parameter Read Only Violation, Parameter Type Violation and Parameter Value Length Violation in this criteria. This would result in the selected error page being displayed when any of these criteria are encountered and subsequently displaying the custom error page.

Associating a Web Error Page Policy to a Web Service

Once a Web Error Page Policy has been configured, it needs to be attached to the relevant SecureSphere site. For information on configuring a web error page policy, see Configuring Web Error Page Policies on page 423.

To associate a web error page policy to a web service:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. In the **Sites Tree**, expand the relevant site and server group, then click on the web service you want to configure. Web service options appear in the **Sites** window.
3. In the **Sites** window, click the **Operation** tab. Operation tab options are displayed.
4. In the **Web Error Page Policies** pane, click  **New**. A new row is added to the web error page policies table. From the drop down list select an existing web error page policy.

5. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Configuring Advanced Web Service Settings

A number of advanced settings regarding how SecureSphere handles web traffic are available. These settings determine how the selected server group manages traffic that matches the configured parameters.

To configure advanced web service settings:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. In the **Sites Tree**, expand the relevant site and server group, then click on the web service you want to configure. Web service options appear in the **Sites** window.
3. In the **Sites** window, click the **Operation** tab. Operation tab options are displayed.
4. In the **Advanced Settings** pane towards the bottom of the screen, select the desired options as follows:
   - **Connection Timeout**: Determines the amount of time a connection can remain idle (and retains contents of a session), before timeout.
   - **Case Sensitive URLs**: Distinguishes similar URLs that only differ by letter case. Enables the profiling of two similar URLs separately.
   - **Support Microsoft Encoding in URLs**: Identifies Microsoft Encoding when used in URLs and translates encoded URLs, preventing false positives and displaying URLs in unicode in web profiles.
   - **Empty Parameter IIS Style**: Identifies empty parameters that are passed in URLs, subsequently building a cleaner profile and promoting more effective monitoring. When IIS is used, parameters of URLs do not need to be followed by an equal sign.
5. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Configuring OCSP Support

Online Certificate Status Protocol (OCSP) is used to obtain the revocation status of X.509 digital certificates. This assists web infrastructure including servers and Internet browsers in determining if a web site certificate is valid, has been revoked, or if its status is unknown. Validating the OCSP protocol verifies the integrity of the information it is carrying and can prevent attempts to serve certificates that may otherwise not be valid.

The following table lists the main tasks that can be conducted to configure OCSP Support.

Configuring OCSP Support Tasks Overview

	Action	Description	For more information, see...

	Action	Description	For more information, see...
1	**Enable OCSP web application option.**	Configure OCSP protocol support under Advanced web application settings and apply the OCSP security policy.	Web Application Definitions Parameters on page 71
2	**Enable Case Sensitive URLs**	To meet OCSP protocol requirements, you need to enable the Case Sensitive URLs option under the Advanced HTTP Settings in the service operations.	See the item **Case Sensitive URLs** in the table **Advanced HTTP Settings** located in Web Service Operations on page 65
3	**Configure URL Prefixes list.**	Configure a URL prefixes list that contain only those URLs that are OCSP.	Configuring URL Prefixes / Directory Based Restrictions on page 420
4	**Customize the OCSP policy (optional).**	You can optionally customize the OCSP policy to the required SecureSphere application. If no policy is configured, alerts will not be generated.	OCSP Protocol Validation Policies on page 212

Once you have activated OCSP support on a web application, it parses OCSP traffic. However, if you also have HTTP traffic running on the server, it can create inconsistencies when parsing and learning events. Subsequently, you must configure a URL Prefixes list that contains all of the URLs that are OCSP, then add that list under **Restrict Monitoring to URLs** on the **Definitions** tab of the specific web application, or alternatively create a list of the HTTP URLs, and configure the service to ignore them under restrictions. You can also configure Host/URL to application mapping. For more information see Configuring Host/URL to Application Mapping on page 63.

EXAMPLE: CONFIGURING OCSP SUPPORT FOR A WEB APPLICATION

This example provides a short overview of how to configure OCSP support for a web application.

To configure OCSP support for a web application:

1. In the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.
2. From the **Scope Selection** dropdown menu, select **URL Prefixes / Directories Groups**.
3. Create a new **URL Prefixes** group and add the URLs in your network that run on OCSP and that you want this application to monitor, then save the URL Prefixes group.
4. Click the **Sites** tab. The **Sites** window appears.
5. In the **Sites Tree**, select the Server Group, Web Service and Web Application to which you want to monitor OCSP traffic.
6. In the **Definitions** tab, under **Restrict Monitoring to URLs**, select the URL prefixes group you created in step 3 above.
7. Under **Advanced Settings**, enable the **Parse OCSP Requests** option.
8. From the top right-hand corner of the window, click **Save**.
9. In the **Sites Tree**, select the Web Service to which you want to monitor OCSP traffic and click **Operation**.
10. In the **Operation** tab, in the **Advanced HTTP Settings** area, select **Case Sensitive URLs**.
11. From the top right-hand corner of the window, click **Save**.
12. Optionally configure an OCSP policy as described in OCSP Protocol Validation Policies on page 212 then apply it to the web application.

Configuring Gateway HTTP Proxies

You can configure a Gateway Group to use an HTTP Proxy to access external sites.

You must first define a Gateway HTTP Proxy global object. For information on how to create and configure a global object, see Creating New Global Objects on page 346. Then specify as follows:

Specify whether to create the object from scratch, or to base its definition on an existing object.

Next,

- **Name** - The name of the HTTP Proxy global object.
- **IP address** - The IP address of the HTTP Proxy.
- **Port** - The port on which communication between the gateways and the HTTP proxy will take place.

After clicking **Create**, open the newly created object and define the following additional parameters:

- **Authentication Policy** - Choose **Basic**, **Digest** or **NTLM** from the dropdown menu.
- Enter the credentials: **User**, **Password** and Verify Password for authenticating to the HTTP proxy.
- **Domain** - Enter the domain.

Finally, click **Save**.

Configuring Webworm Protection

A web worm utilizes a web server vulnerability to spread to a large amount of web servers in a short period of time. Some web worms utilize well-known vulnerabilities. These worms can be easily stopped using the signatures layer, as a signature for that attack probably already exists. Zero-day worms utilize as of yet unknown signatures and therefore require special configuration to prevent their propagation. The following table lists the main tasks that can be conducted to configure webworm protection.

Configuring Webworm Protection Tasks Overview

	Action	Description	For more information, see...
1	**Configure susceptible directories.**	Configure directories that are susceptible to webworm attack.	Configuring Susceptible Directories on page 426
2	**Configure web worm policy.**	Configure the policy that determines how to protect susceptible directories.	Securing Web Applications on page 206
3	**Associate policy to application.**	Associate the policy to the required SecureSphere application.	Applying Policies on page 318

Configuring Susceptible Directories

The Susceptible Directory Group is a list of folders on your web server which may contain sensitive data. You can then configure profiles and policies to identify these groups as susceptible and subsequently provide tighter control over users who can access and modify the contents in these folders. The Default Susceptible Folder List cannot be modified.



Note: When a susceptible folder is protected, it cannot be deleted from the profile (not even if the related policy is deleted/un-applied to the application).

To add directories to a susceptible directory group:

1. In the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window appears.

2. In the **Objects** dropdown list, select **Susceptible Directory Groups**. Existing **Susceptible Directory Group Lists** are displayed.
3. In the **Globals Tree**, select an existing Susceptible Directories list or create a new list as described in Creating New Global Objects on page 346. The selected Susceptible Directories list appears in the pane on the right-hand side of the screen.



Note: You cannot modify the Default Susceptible Directories list however you can create a new list based on the default and containing all of its objects.

4. Click  **New.** A new line is added to the bottom of the **Susceptible Directory Elements** table.
5. Type the directory you want to add as a susceptible directory

   You can now associate the Susceptible Directory group as required.

Configuring Session Tracking

Session tracking enables SecureSphere to track sessions by using token names. Using session identifiers rather than IP addresses to track sessions provides a more accurate method of tracking sessions.

Configuring Session Tracking involves the following procedures:

- Adding Tokens to a Session Tracking List on page 427
- Associating a Session Tracking List to a Web Service on page 428

Adding Tokens to a Session Tracking List

Session identifiers are pieces of data used to identify a session. SecureSphere uses session identifiers as an alternative to IP addresses to identify users when monitoring, auditing and blocking traffic. Session identifiers are stored either within session cookies or in HTTP parameters. Configuring the system to identify parameters or cookies that hold session identifiers enables SecureSphere to track user activity accurately over time. SecureSphere comes with a list of standard session identifiers for most major applications already installed. However, your network or applications may use custom identifiers. In which case they need to be added.

To add tokens to a session tracking list:

1. Consult with your system administrator and obtain a list of token names and cookie types used in your network. This is used to create the session tracking list.
2. In the **Main** workspace, select **Setup > Global Objects**. The **Global Objects** window is displayed.
3. In the **Objects** dropdown list, select **Sessions Identifiers Groups**. Existing **Sessions Identifiers Groups** are displayed.
4. In the **Globals Tree**, click the **Default Sessions Tracking List**. The **Default Sessions Tracking List** is displayed in the right-hand side of the window.
5. In the **Token Name** title bar, click  **New**. A new row is added at the bottom of the **Session Identifiers** table.
6. Type the custom **Token Name** used in your network.
7. Select a **Token Type**. Token type is the means by which your system passes a token. If users in your network have cookies turned on, then the token is passed in the form of a cookie or cookie prefix. If cookies are not turned on, then they are passed as a parameter in the URL.
8. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Associating a Session Tracking List to a Web Service

Once a Session Tracking List has been configured with tokens, it needs to be attached to the relevant SecureSphere site. For information on adding tokens to a Session Tracking List, see Adding Tokens to a Session Tracking List on page 427.

To associate a session tracking list to a web service:

1. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
2. In the **Sites Tree**, expand the relevant site and server group, then click on the HTTP service you want to configure. HTTP service options appear in the right-hand side of the window.
3. In the **Sites** window, click the **Operation** tab. Operation tab options are displayed.
4. In the **Session Tracking** pane, select the **Session Tracking List with the identifiers** you want to use for tracking sessions.
5. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

URL Rewrite

SecureSphere's URL Rewrite feature can be used in two situations:

- To rewrite host name and/or path in Web traffic URLs and headers.

  In this case the Web client is unaware that the traffic is being diverted.
- To redirect Web traffic to another server.

  In this case, response code 302 is sent to the Web client, which then accesses the new URL in a new request.

The SecureSphere administrator defines URL Rewrite rules that define how to rewrite or redirect responses and/or requests. URL Rewrite rules are organized in groups, and these groups are applied to Web services.

Both URL Rewrite and URL Redirection are configured in the same way, with only a single parameter in the **Create URL Rewrite Rule** window being different. For more information, see Creating a URL Rewrite Rule on page 431.

Because URL Rewrite and URL Redirection intercept and modify Web traffic, they are available only when the SecureSphere gateway is configured in the Reverse Proxy or Transparent Reverse Proxy modes.

URL Rewrite Architecture

With URL rewrite, the Web client remains unaware of the redirection. This feature can be used for several purposes, for example:

- When a Web site's name has been changed, but applications accessing the site have not been updated to use the new name.

- To hide Web servers and Web site structure from clients.

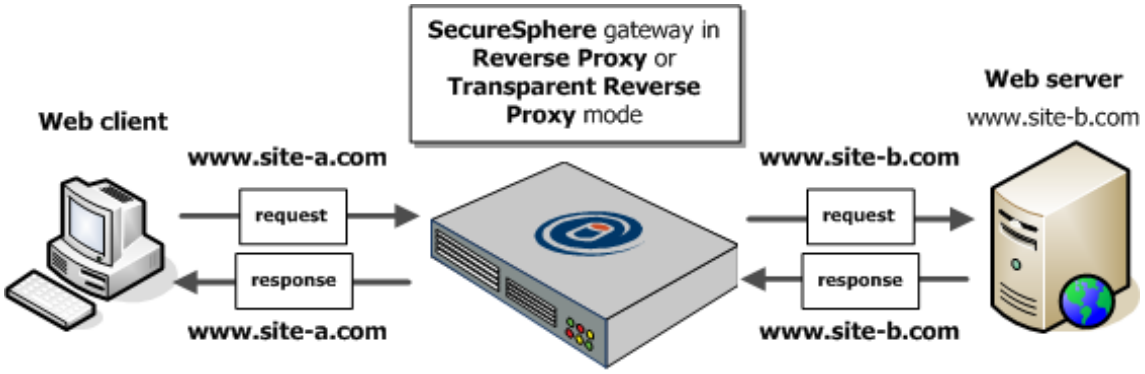


Figure 70: URL Rewrite

Note: Rewrite rules come in pairs, one rule for incoming traffic and an inverse rule for outgoing traffic so that the Web client is unaware of the URL rewrite.

The SecureSphere gateway, configured in Reverse Proxy or Transparent Reverse Proxy mode, intercepts both requests from the client and responses from the server. The SecureSphere administrator must define URL rewrite rules in pairs, one rule for incoming traffic (requests) and an inverse rule for outgoing traffic (responses). In this way the Web server and the Web site structure are hidden from the Web client.

URL Redirection

With URL redirection, the SecureSphere gateway, configured in Reverse Proxy or Transparent Reverse Proxy mode, intercepts Web client requests and returns a 302 response, redirecting the Web client to another URL.

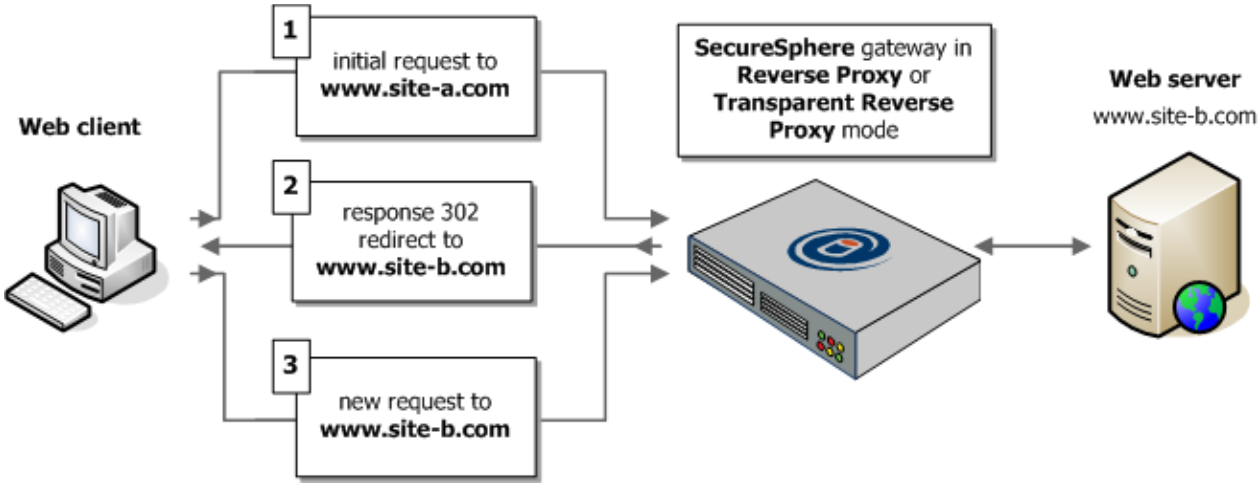


Figure 71: URL Redirection

Note: With URL Redirect, the Web client is aware of the redirection.

URL redirection can be used for several purposes, for example:

- When a Web site has been reorganized, but Web users may still be accessing the now obsolete URLs
- When Web site administrators want to force Web clients to use HTTPS, even if they access a HTTP URL

URL Rewrite Groups

URL rewrite rules are grouped in URL rewrite groups, and the groups are applied to Web services. Rules in a group are evaluated sequentially in order of their priorities. All matching rules are applied, but there is significance to priorities because each rule operates on the output of the previously applied rule.

For example, if a group consists of the following rules:

- Rule 20: rewrite www.X.com/ to www.Y.com/
- Rule 30: rewrite www.Y.com/forms to www.Z.com/

Rule 20 does not hide Rule 30, which is applied to Rule 20's output, if it matches. So www.X.com would become www.Z.com.

If more than one URL Rewrite Group is applied to a Web service, then the highest priority rules in all the groups are applied first, followed by the second highest priority rules in the second group, and so on.



Note: For information on avoiding infinite loops, see Avoiding Infinite Loops on page 430.

Avoiding Infinite Loops

With URL Redirection, administrators should be careful not to write rules that result in infinite loops. For example, consider the following rules.

- Rule 20: redirect www.X.com/ to www.Y.com/
- Rule 30: redirect www.Y.com/ to www.X.com/

The sequence of events is then as follows:

1. The client sends a request to access www.X.com.
2. Because of rule 20, the SecureSphere gateway sends the client a 302 redirect response to www.Y.com.
3. The client sends a request to access www.Y.com.
4. Because of rule 30, the SecureSphere gateway sends the client a 302 redirect response to www.X.com.
5. The client sends a request to access www.X.com.
6. Because of rule 20, the SecureSphere gateway sends the client a 302 redirect response to www.Y.com.

There is no end to this cycle of request-redirect-request-redirect.



Note:

- Infinite redirection loops can occur with URL Redirection, but not with URL Rewrite.
- Infinite redirection loops can occur only if the traffic to the redirected server also passes through the SecureSphere gateway.

-

Configuring URL Rewrite and URL Redirection

To configure URL Rewrite and URL Redirection perform the following tasks:

URL Rewrite and URL Redirection Tasks Overview

Task	For more information, see

	Task	For more information, see
1	**Create a Rewrite Group Global Object**	Creating a URL Rewrite Group on page 431
2	**Create and configure any number of Rewrite Rules**	Creating a URL Rewrite Rule on page 431
3	**Associate the Rewrite Group with a Web service**	Applying a URL Rewrite Rule Group to a Web Service on page 432

Creating a URL Rewrite Group

To create a URL Rewrite Group:

1. In the **Main** workspace, select **Setup > Global Objects**.
2. In the **Objects** dropdown list, select **URL Rewrite Groups**.
3. In the **Globals Tree** pane, click .
4. In the **Create URL Rewrite Group** window, enter a **Name** for the group.
5. Select either **From Scratch**, or **Use Existing** and select a group from the list.

   If you select **Use Existing**, the new group will inherit its parameters from the existing group. You can select **Use Existing** only if other groups already have been defined.
6. Click **Create**.
7. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.
8. Next, populate the URL Rewrite group with URL Rewrite rules.

Creating a URL Rewrite Rule

To create a URL Rewrite Rule:

1. In the **Main** workspace, select **Setup > Global Objects**.
2. In the **Objects** dropdown list, select **URL Rewrite Groups**, then select a URL Rewrite group.
3. In the middle pane, click .
4. In the **Create URL Rewrite Rule** window, enter the following data:
   - **Name**: Enter a name for the rule.
   - **Priority**: Enter a priority.

     Rules in a group are evaluated sequentially in order of their priorities. All matching rules are applied, but there is significance to priorities because each rule operates on the output of the previously applied rule.

     Tip: It is highly recommended to assign different priorities to each rule in the group, in order to avoid ambiguity in the order of their application. Also, it is recommended that you assign priorities in multiples of ten, that is, 10, 20, 30 and so on, so that you can insert additional rules in between them at a later date, if required.
   - **Find what**: Enter **Host** and **Path** strings to search for.

     Note: Both **Host** and **Path** can be regular expressions. For example, in **Path**, to match "default", write "default$", otherwise "default/xxx" will also be matched.
   - **Replace with**: Enter **Host** and **Path** strings with which to replace the **Host** and **Path** under **Find what**.
5. Under **Apply To**, select **Request** or **Response**.

With URL Rewrite rules, as opposed to URL Redirection rules, you must define rules in pairs: one rule for **Request** traffic and an inverse rule for **Response** traffic.

6. To specify that this is a URL Redirection rule rather than a URL Rewrite rule, select **Redirect Client Requests** under **Redirection**.

   If you selected **Response** under **Apply To**, then **Redirect Client Requests**.
7. Select **HTTP**, **HTTPS** or **Original**. Note that the protocol is not specified under **Replace with**, so you must specify it here.
   - **HTTP**: Redirect to HTTP URLs.
   - **HTTPS**: Redirect to HTTPS URLs.

     This option is useful, for example, for financial sites which want to ensure that all access to their sites uses HTTPS.
   - **Original**: Redirect to the same protocol as the original request.
8. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Applying a URL Rewrite Rule Group to a Web Service

To apply a URL Rewrite Rule Group to a Web Service:

Before you apply a URL Rewrite Group to a Web service, you must first define the URL Rewrite Group. For more information, see Creating a URL Rewrite Group on page 431.

1. In the **Main** workspace, select **Setup** > **Sites**.
2. In the **Sites** pane, select a site.
3. Select an HTTP service.
4. In the middle pane, click the **Reverse Proxy** tab.
5. Under **URL Rewrite**, click .
6. Select a URL Rewrite group from the list.

   You can select any number of URL Rewrite groups.
7. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Supporting Web Load Balancers

This section reviews the various aspects of working with web load balances and forwarding connections in SecureSphere. It reviews the following subjects:

- Overview of Load Balancer and Forwarded Connections on page 432
- Providing Full Visibility of Client IP Addresses on page 433
- Providing Accurate IP Blocking on page 434
- Configuring Forwarded Connections on page 434

Overview of Load Balancer and Forwarded Connections

The **Forwarded Connections** capability provides you with full visibility of client IP addresses and prevents unnecessary IP blocking in SecureSphere configurations with web load balancers, which distribute the load between the web servers, provide caching and other services.

SecureSphere can be installed between the load balancer and the server, as in the first figure below, or between the balancer and the client, as in the second figure below.

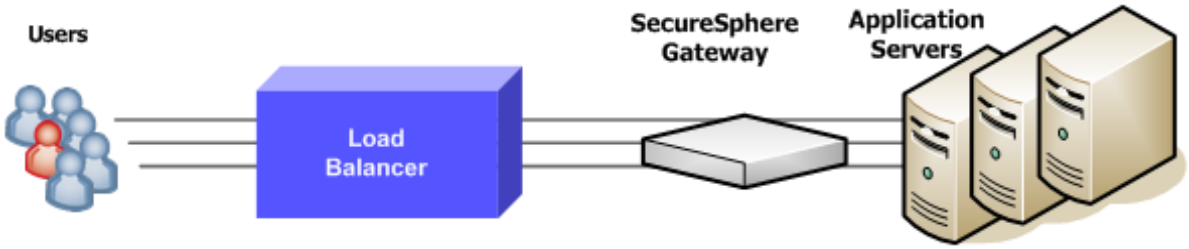


Figure 72: SecureSphere between the Load Balancer and the servers

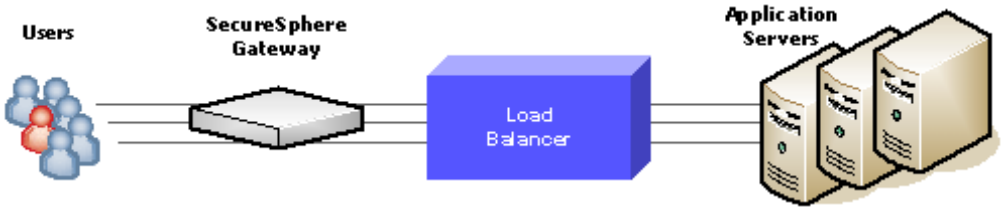


Figure 73: SecureSphere between the Load Balancer and the clients

Load balancers can be installed using the following modes:

- **Transparent mode**: The load balancer communicates with the server using the real Source IP address of the client.
- **Non-transparent**: The load balancer communicates with the server using load balancer's IP address.

SecureSphere's load balancer support is required when the load balancer is between the users and the SecureSphere Gateway and in the Non Transparent mode. The following table shows when the **Forwarded Connections** capability is used.

Using the Forwarded Connection Capability

	SecureSphere is deployed between the client and the Load Balancer	SecureSphere is deployed between the Load Balancer and the server
Load Balancer in the Transparent mode	Not relevant	Not relevant
Load Balancer in the Non-Transparent mode	Not relevant	Required

Providing Full Visibility of Client IP Addresses

When SecureSphere is deployed in the load balancer's environment, client IP addresses are typically represented by load balancer's IP address in alerts and reports, which makes it impossible to identify the attacker or even get information about clients. To provide full visibility of client IP addresses in alerts and reports, you can use the **Forwarded Connections** capability.

Providing Accurate IP Blocking

The **Forwarded Connections** capability enables you to work with clients behind the proxy in the same manner as with the regular clients, blocking only the IP addresses that sent problematic load. This is done using **Followed Action**, see Working with Action Sets and Followed Actions on page 326.

When SecureSphere detects a policy violation, one of the options to react to that violation is to block the source IP address from which the policy violation originated. So when the load balancer is transparent and uses the client's IP address for the server side connection, SecureSphere blocks the client. When the load balancer uses its own IP address for the server side connection and no forwarded connections are configured, SecureSphere blocks the load balancer's address, causing a denial of service to the load balancer's traffic.

So when the client's IP address is used by the server, then only the problematic source is blocked. In case your servers use the load balancer's IP, you need to configure forwarded connections. Otherwise this IP address is blocked together with all the client IP addresses behind the load balancer, which must not happen in an environment with content delivery solutions.



Note: The Forwarded Connections X-Forwarded-For feature only begins to block requests once the actual source IP address of an XFF header is identified by SecureSphere. Additionally note that the destination server may see partial requests until this point, but will not receive all headers. If a header triggers a violation before SecureSphere completes parsing the header, SecureSphere will stop sending packets to the server even before the actual source IP address is identified. In any of these cases, the request is not parsed by the web server.

Configuring Forwarded Connections

The **Forwarded Connections** capability enables SecureSphere to identify the IPs behind the proxy. When SecureSphere is installed between the web servers and the load balancer in the Non-Transparent mode, Non-Transparent proxy applications usually indicate the IP address of the original client connected to the proxy. The indication is for general usage by the web server or any other network device.

Usually this is done using the dedicated HTTP header **X-Forwarded-For**, although some applications may use a different header. When there are several proxies between the web-servers, the **X-Forwarded-For** header lists the IP addresses of all of them, after that of the client. SecureSphere uses the last IP address in the list (which is either that of the real client or of the last proxy, that is, the proxy closest to the SecureSphere Gateway) as the source IP address, both in matching source IP addresses in correlation rules, and also in alerts issued to the MX.

When the header is not sent by the load balancer, or not parsed by SecureSphere, the load balancer's IP address is used by SecureSphere. To avoid blocking of the load balancer's IP address, the followed actions that are used for blocking must be configured never to block the load balancer's IP address. This is done by defining the load balancer IP addresses as **Trusted IPs** in the followed action's configuration.

In addition, you can define a custom web correlation policy that protects against infected load balancers. When the **Forwarded Connections** capability is used, the **Source IP Addresses** match criteria identifies client IPs by matching the HTTP headers. You can define a correlation policy that validates not only the client IPs, but also the load balancer's IP address. This is done by using the **Source IP Addresses** match criteria in combination with the **Proxy IP Address** match criteria, where **Proxy IP Address** is used to identify load balancers.



Note: When used with traffic that has been modified as part of Forwarded connections, the IP Block action interface is applied to the traffic's source address.

The following table summarizes the configuration tasks related to the **Forwarded Connections** capability.

Forwarded Connection Tasks Overview

Tasks Overview	See
To enable working with clients behind the proxy in the same manner as with the regular clients, you need to perform the following: • To identify client IPs, define **Forwarded Connections**. • To make sure that load balancer IPs are never blocked in case the header is unknown, define the load balancer's address as **Trusted IP**.	The procedure: **To configure Forwarded Connections** below The procedure: **To define load balancer's address as trusted address** below
To define a custom web correlation policy using the real client IPs when the clients are behind the load balancers, you can use the following match criteria: • Source IP Addresses • Proxy IP Addresses	Web Service Custom Policies on page 198

To configure Forwarded Connections:

1. Define the load balancer IPs as an IP group, see Configuring IP Groups on page 347.
2. In the **Main** workspace, select **Setup**, then from the **Filtering** pane select the service for which you want to define **Forwarded Connections**. The **Details** pane appears.
3. To enable Forwarded Connections, select **Operation** and then under **Forwarded Connections**, select **Identify real client IP according to HTTP forwarding header**.
4. In the **Forwarded Connections** table, click . The new row appears in the table.
5. In the **Forwarded Connections** table, set the parameters in the table **Forwarded Connections Parameters Overview** below.
6. Click **Save**.

Forwarded Connections Parameters Overview

Parameters	Description
Header	Define the dedicated header that indicates the source IP addresses. Default: X-Forwarded-For. **Note:** In CDN deployments or in non-transparent load balancer deployments, you cannot ignore alerts based on IP addresses in XFF headers (for example, using the Ignore IP Groups global object). It is recommended that you create an exception for each triggered policy.
Proxy IP Group	Select the IP group that includes IP addresses of the load balancers. It is recommended to define real addresses and not to use **Any** in order to provide protection to the specific IP range. Define separate IP groups for different load balancers, as each group of clients might use a different header. For instructions about how to define an IP range, see Configuring Webworm Protection on page 426.

To define load balancer's address as trusted address:

1. In the **Main** workspace, select **Policies > Action Sets**. The **Action Sets** window appears.
2. In the **Action Sets** window, click the green arrow next to **Long IP Block** to move it to the **Selected Items** area.
3. Define the trusted IPs as described in Configuring Action Interface Parameters on page 332.

To move to reverse proxy:

- To indicate client IP addresses when SecureSphere is in the Proxy mode, you need to enable the **Report forwarded client IP in HTTP header** option.

To indicate the client IP addresses behind the SecureSphere system in the proxy mode:

1. In the **Main** workspace, select **Setup**, then from the **Site Tree** pane select the service that you want. The **Definitions** pane appears.

2. In the **Reverse Proxy** area, select **Report forwarded client IP in HTTP header**.

Validating SSL Client Certificates

SecureSphere enables you to validate client certificates in Kernel Reverse Proxy mode, which authenticates users before enabling them to access web applications. This section reviews the various global objects which need to be prepared in order to configure client certificate validation and the process of associating these global objects with a web service to validate client certificates.

To configure client certificate validation a number of objects need to be prepared. Once these objects have been configured, you complete configuration on the SecureSphere web service listed as hosting the traffic you want to validate. The following table lists the main tasks involved with configuring client certificate validation.

Configuring Client Certificate Validation – Task Overview

	Action	Description	For more information, see...
1	**Import Certificate Authority Public Certificates**	Import public certificates obtained from a Certificate Authority you want to use to validate in your environment.	Configuring Certificate Authorities on page 436.
2	**Import a Certificate Revocation List - CRL (optional)**	Import all up-to-date CRLs to identify what certificates have been revoked.	Configuring Certificate Revocation List (CRL) on page 437.
3	**Configure Certificate Authority Groups (optional)**	Create a CA Group and add certificates previously imported from a certificate authority.	Configuring Certificate Authority Groups on page 437.
4	**Configure Client Authentication Rules**	Select the parameter to be taken from the client and what is done with it on validation success and failure.	Configuring Client Authentication Rules on page 438.
5	**Associate certificate and Authentication rule with Web Service**	Configure a web service with a CA Group and Client Authentication Rule to complete configuration.	Associating Client Certificate Validation on page 438

Configuring Certificate Authorities

Certificate Authorities enable users to download certificates containing a public key from valid certificate authorities and then use this public key to validate traffic. The first step in this process is downloading the required public keys into the Certificate Authorities global object.



Note: SecureSphere supports the .pem (privacy enhanced mail) and .crt certificate formats.

To import certificates into the Certificate Authorities global object:

1. Download certificates from the relevant certificate authority.
2. Under **Client Authentication Authorities** click **Edit** then select **Certificate Authorities**, or from the **Setup > Global Objects** window select **Certificate Authorities > Certificate Authorities**. Existing Certificate Authorities are displayed.
3. In the heading of the **Details** pane, click **New**. The **Upload New Certificate** window opens.

4. Navigate to the location of the CA file you downloaded from the Certificate Authority in step 1.
5. If you want to overwrite an existing certificate, select the **Overwrite Existing Certificate From Same Issuer** checkbox.
6. Click **Upload**. The certificate is uploaded.
7. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

   If your application accepts traffic from more than one CA, you can add these certificates to a certificate group. For more information on certificate groups, see Configuring Certificate Authority Groups on page 437.

Configuring Certificate Revocation List (CRL)

Certificate Revocation Lists (CRL) are lists of certificates that have been revoked which are used to check the validity of a certificate. SecureSphere allows you to download CRLs from Certificate Authorities in the form of .crl files, which are then used by SecureSphere during the process of validation.

To import CRLs into the CRL global object:

1. Download one or more Certificate Revocation Lists (CRLs) from the relevant certificate authority.
2. Under **Client Authentication Authorities** click **Edit** then select **Certificate Revocation Lists**, or from the **Setup > Global Objects** window select **Certificate Authorities > Certificate Revocation Lists**. Existing Certificate Revocation Lists are displayed.
3. In the heading of the **Details** pane, click **New**. The **Upload New CRL** window opens.
4. Type a **Name** for the CRL list.
5. Select a **Certificate Authority** with which to associate the CRL list.



Note: CRLs contain a list of revoked certificates associated with a certificate authority. If no certificate authority is selected, users with bad certificates will not be filtered.

6. Click **Browse** and navigate to the location of the CRL file you downloaded from the Certificate Authority in step 1.
7. Click **Upload**. The CRLs are uploaded.
8. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Configuring Certificate Authority Groups

If you configure more than one Certificate Authority, you can group them for use in a Certificate Authority Group. Certificate Authorities Groups enable you to group related certificates to be used in a specific environment. Once group has been configured, you associate it with a Web Service under its Reverse Proxy settings.



Note: Certificate Authority Public Certificates are assigned to the default group if no groups are manually configured.

To configure a new Certificate Authority Group:

1. Under **Client Authentication Authorities** click **Edit** then select **Certificate Authority Groups**, or from the **Setup > Global Objects** window select **Certificate Authorities > Certificate Authority Groups**. Existing Certificate Authority Groups are displayed.
2. From the upper right-hand corner of the **Globals** pane, click **New**. The **Create Certificate Authority Group** window opens.
3. Type a **Name** for the Certificate Authority Group.
4. Click **Create**. The group is created.

5. Move the desired certificates to the Group Members pane by clicking the right arrow.
6. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

   You now need to associate the certificate group with a web server. For more information see Associating Client Certificate Validation on page 438.

Configuring Client Authentication Rules

A Client Authentication Rule determines various factors regarding how SecureSphere will handle traffic when a certificate validation has succeeded, and failed. And when successful, determines what information is passed on to the web server.

To configure Client Authentication Rules:

1. Under **Client Authentication Authorities** click **Edit** then select **Certificate Authentication Rules**, or from the **Setup > Global Objects** window select **Certificate Authorities > Certificate Authentication Rules**. Existing Certificate Authentication Rules are displayed.
2. From the upper right-hand corner of the **Globals** pane, click **New**. The **Create Client Authentication Rule** window opens.
3. Type a **Name** for the client authentication rule, then click **Create**. The new rule is created and its options are displayed.
4. Select the action to be taken **On Failure**. Options include:
   - **Block**: Blocks the connection
   - **Block and Send Error Page**: Blocks the connection and displays the selected error page to users. Select an error page to use by clicking the dropdown list, or create a new one by clicking the edit button
   - **Don't Block and send in header**: Does not block the connection and sends the configured information in the header.
5. Select an item to be sent in the header **On Success**. Options include:
   - **Nothing**: Does not send anything to the web server if authentication was successful.
   - **Subject CN**: Sends the Common Name (CN) in the subject field to the web server.
   - **Subject Email**: Sends the e-mail of the in the subject field to the web server.
   - **Entire Certificate**: Sends the entire certificate to the web server.
6. If required, modify **HTTP Header Name**: by default, the HTTP Header name is: **X-Client-Cert**.
7. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

   You can now use the Client Authentication Rule to configure client certificate validation. For more information see Associating Client Certificate Validation on page 438.

Associating Client Certificate Validation

Once all the required global objects containing information regarding Certificate Authorities, Client Authentication Rules, and if relevant Certificate Revocation lists have been configured, you need to associate them with the desired web service in order to activate client certificate validation.



Notes:

- Authenticating a client certificate requires specifying both a CA and a Server Certificate.
- When configuring a Reverse Proxy Rule to authenticate a client certificate, initiating an encrypted connection for the rule always requires the client to present a certificate.

To configure Client Certificate Validation:

1. Configure the Certificate Authority, Certificate Authority Group, Certificate Revocation List (if desired), and Client Authentication Rules global objects as described in table **Configuring Client Certificate Validation – Task Overview** located in Validating SSL Client Certificates on page 436.

2. Configure the desired web service for Reverse Proxy support. For more information see Reverse Proxies on page 459.
3. In the **Main** workspace, select **Setup > Sites**. The **Sites** window appears.
4. In the **Sites window**, select **Server Group**, and **Web Service** you want to configure to support client certificate validation. That web service's details are displayed.
5. Click the **Reverse Proxy** tab. **Reverse Proxy** options are displayed.

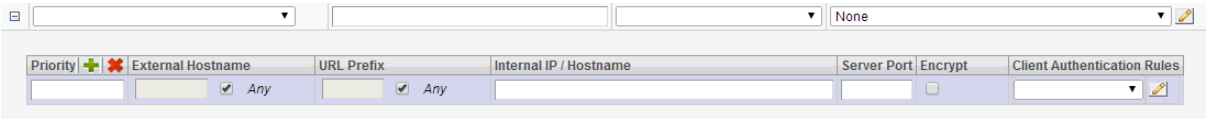


Figure 74: Reverse Proxy Decision Rule Configuration

6. If you are adding Client Authentication to an existing Reverse Proxy configuration, under **Client Authenticating Authorities**, select the **Client Authority Group** you previously configured. For more information, see Configuring Certificate Authority Groups on page 437.
7. If you are assigning priority, under Client Authentication Rules, select the Rules you previously created. For more information, see Configuring Client Authentication Rules on page 438.



Note: For information on configuring Reverse Proxy and Priorities before configuring Client Authentication support, see Kernel (Non-Transparent) Reverse Proxy on page 459

8. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Malware Detection Support

Malware detection products such as the FireEye Malware Protection System (MPS) identify malware-infected hosts on internal IPs.

SecureSphere supports integration with FireEye MPS. SecureSphere enables you to define defensive actions against the malware-infected hosts that FireEye MPS identifies.

FireEye MPS passes information about malware-infected hosts to SecureSphere. The SecureSphere MX server manages a list of internal IPs infected by malware. Based on the threat level of the malware infection, you apply SecureSphere policies that trigger a defensive action such as audit, alert, or block.

Malware detection is supported by SecureSphere version 10.0, patch 2 and later.



Note: SecureSphere malware detection is supported only when your system uses FireEye MPS.

The following table lists the main malware detection support tasks.

Malware Detection Support Task Overview

	Action	Description	For more information, see...

	Action	Description	For more information, see...
1	**Configure malware detection support in SecureSphere and FireEye.**	Configure the SecureSphere malware detection feed user and the malware detection IP group. Configure FireEye to enable SecureSphere malware detection support. Verify that FireEye passes malware information to SecureSphere.	Configuring Malware Detection Support on page 440
2	**Display and understand local IP addresses identified as infected by malware.**	Display and understand the **Lookup Data Set: Malware Infected Local IP Addresses** table, which lists local IP addresses infected by malware and characteristics of the malware.	Understanding the List of Local IP Addresses Identified as Infected by Malware on page 448
3	**Modify advanced settings for malware detection support.**	You may: • Specify names and targets to exclude from malware identification. • Modify the threat level mapped to malware types.	Modifying Advanced Settings for Malware Detection Support on page 450
4	**Apply policies that relate to malware-infected local IP addresses.**	Understand and apply recommended policies, or define and apply custom policies, which trigger a defensive action when malware-infected local IP addresses are identified.	Applying Malware Detection Policies on page 454

Configuring Malware Detection Support

You must configure both SecureSphere and FireEye to enable SecureSphere support of FireEye malware detection.

In SecureSphere you:

1. Create and configure a user to handle the malware detection feed from FireEye.
2. Configure the malware detection IP group global object with the IP address of the FireEye appliance so that SecureSphere recognizes the FireEye malware feed address.

In FireEye, you configure HTTP Server settings so that FireEye passes information about malware-infected hosts to SecureSphere.

After configuring malware detection support in SecureSphere and FireEye, you verify that FireEye passes malware information to SecureSphere.



Notes:

- All names must be entered in SecureSphere exactly as specified.
- SecureSphere assumes that only one FireEye server scans a network. If more than one FireEye server scans the same network, a SecureSphere MX only accepts updates about that network from the first FireEye server. A SecureSphere MX can accept input from multiple FireEye servers only when each server scans a different network.

The following table lists the malware detection support configuration tasks.

Malware Detection Support Configuration Task Overview

	Action	Description	For more information, see...
1	**Configure the malware detection feed user.**	In SecureSphere: • Create the malware detection feed user. • Configure permissions for the malware detection feed user. • (For SecureSphere versions 10.0 and later) Change the malware detection feed user password.	Configuring the Malware Detection Feed User on page 441
2	**Configure the malware detection IP group.**	In SecureSphere, configure the malware detection IP group global object with the IP address of the FireEye appliance.	Configuring the Malware Detection IP Group on page 444
3	**Configure FireEye to enable SecureSphere malware detection support.**	In FireEye, configure FireEye to enable SecureSphere malware detection support. In FireEye and SecureSphere, verify that FireEye passes malware information to SecureSphere.	Configuring FireEye to Enable SecureSphere Malware Detection Support on page 445

Configuring the Malware Detection Feed User

You create a user to handle the malware detection feed from FireEye. Then you assign the user permissions to the Lookup Data Sets related to the malware detection feed.

If you are using SecureSphere versions 10.0 and later, you also need to change the password for the malware detection feed user.

The following table lists the malware detection feed user configuration tasks.

Malware Detection Feed User Configuration Task Overview

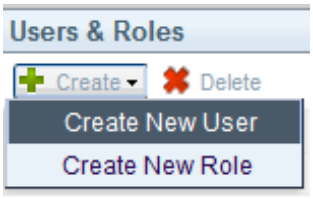
	Action	Description	For more information, see...

	Action	Description	For more information, see...
1	**Create the malware detection feed user.**	Create a SecureSphere user that handles the malware detection feed from FireEye.	Creating the Malware Detection Feed User on page 442
2	**Configure permissions for the malware detection feed user.**	Assign the malware detection feed user permissions to the Lookup Data Sets related to the malware detection feed.	Configuring Permissions for the Malware Detection Feed User on page 443
3	**Change the malware detection feed user password.**	Change the password for the malware detection feed user.	Changing the Malware Detection Feed User Password on page 444

Creating the Malware Detection Feed User

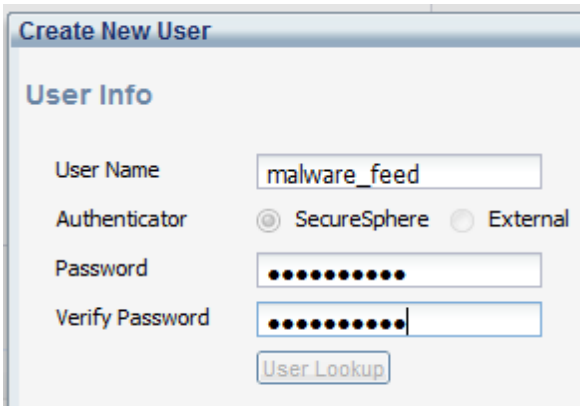
To create the malware detection feed user:

1. In the **Admin** workspace of the SecureSphere MX, select **Users & Permissions**. The **Users & Permissions** window appears.
2. In the **Users & Roles** pane, click  **Create** and select **Create New User**.



The **Create New User** window opens.

3. Under **User Info**:
   - In the **User Name** field, enter **malware_feed**.
   - Enter and verify a password.





Notes:

- For SecureSphere versions 10.0 and later, you must change the password later. See Changing the Malware Detection Feed User Password on page 444.
- Do not assign roles to the new user.

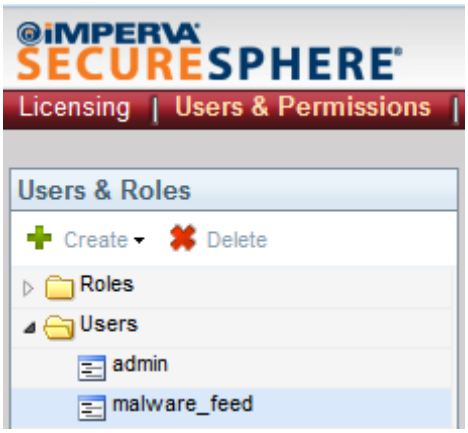
4. To create the malware detection feed user, click **Create**.

Configuring Permissions for the Malware Detection Feed User

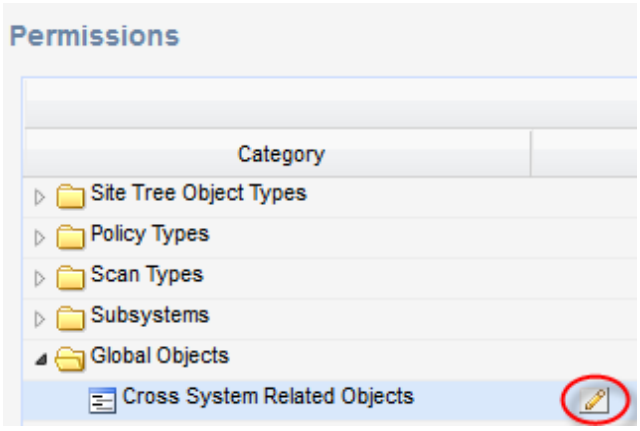
SecureSphere uses Lookup Data Sets to display the malware detection feed from FireEye and to enable you to modify how the feed is displayed and handled. See Understanding the List of Local IP Addresses Identified as Infected by Malware on page 448 and Modifying Advanced Settings for Malware Detection Support on page 450. You configure user permissions to enable the malware detection feed user appropriate access to the Lookup Data Sets.

To configure permissions for the malware detection feed user:

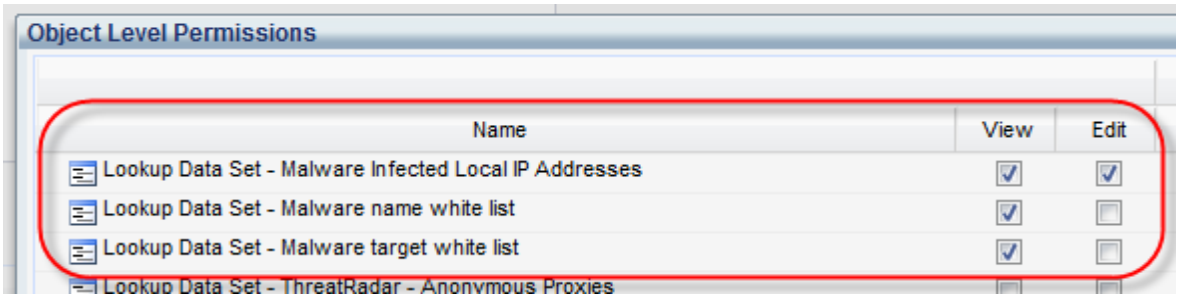
1. In the **Admin** workspace of the SecureSphere MX, select **Users & Permissions > Users & Roles > Users > malware_feed**.



2. In the **Permissions** tab under **Permissions**, expand **Global Objects**.
3. Select **Cross System Related Objects** and click to display the **Object Level Permissions** window.



4. In the **Object Level Permissions** window:



- For **Lookup Data Set - Malware Infected Local IP Addresses**, enable **View** and **Edit**.
- For **Lookup Data Set - Malware name white list**, enable **View**.

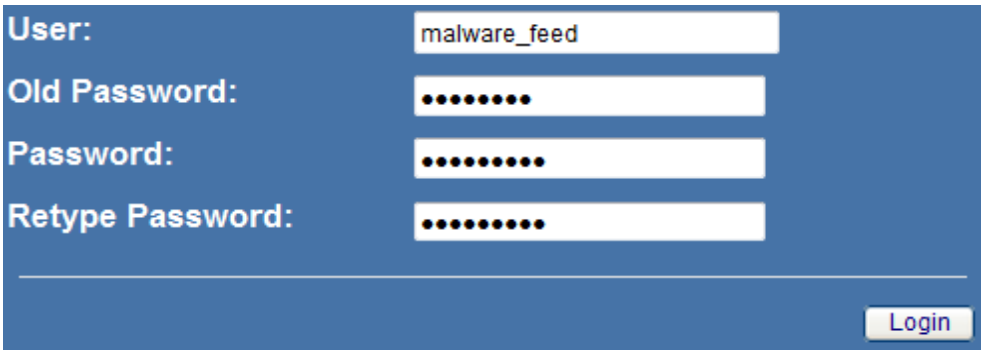
- For **Lookup Data Set - Malware target white list**, enable **View**.

5. In the **Object Level Permissions** window, click **Save**.
6. In the **Permissions** tab of the **Users & Permissions** window, click **Save**.

Changing the Malware Detection Feed User Password

To change the malware detection feed user password:

1. On the SecureSphere menu bar, click **Log out**.
2. In the log out confirmation window, click **Yes**. The SecureSphere login screen appears.
3. In the SecureSphere login screen, enter the following:
   - **User**: Enter the malware detection feed user name: **malware_feed**.
   - **Password**: Enter the password that you entered when you created the malware detection feed user.
4. In the SecureSphere login screen, click **Login**. The change password screen appears.
5. In the SecureSphere change password screen, enter the following:



   - **User**: Enter the malware detection feed user name: **malware_feed**.
   - **Old Password**: Enter the password that you entered when you created the malware detection feed user.
   - **Password**: Enter a new password for the malware detection feed user.
   - **Retype Password**: Re-enter the new password.
6. In the SecureSphere login screen, click **Login**. The Main SecureSphere workspace appears.
7. On the SecureSphere menu bar, click **Log out**.
8. In the log out confirmation window, click **Yes**. The SecureSphere login screen appears.
9. In the SecureSphere login screen, log in with your regular user name and password.

Configuring the Malware Detection IP Group

To enable SecureSphere to recognize the FireEye malware feed address, you configure the malware detection IP group global object with the IP address of the FireEye appliance.

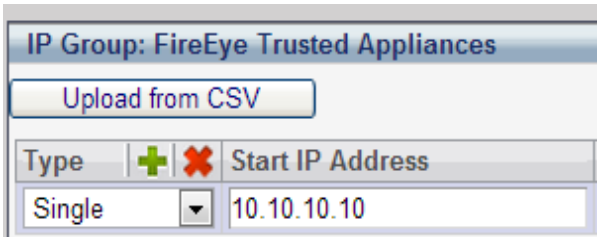
To configure the malware detection IP group:

1. In the **Main** workspace of the SecureSphere MX, select **Setup > Global Objects.**
2. In the **Scope Selection** menu, select **IP Groups**.

3. In the **Globals Tree** pane, under **IP Groups** select **FireEye Trusted Appliances**.



4. In the **IP Group: FireEye Trusted Appliances** pane, in the **Type** column heading, click . A new row is added to the table.
5. In the **Type** column, select **Single**.
6. In the **Start IP Address** column, enter the IP address of the FireEye appliance that sends data to SecureSphere. Do not enter values for other columns.



7. Click **Save**.

Configuring FireEye to Enable SecureSphere Malware Detection Support

After configuring SecureSphere to support malware detection, you configure FireEye MPS to pass information about malware-infected hosts to SecureSphere.

After completing SecureSphere and FireEye configuration, you verify that FireEye passes malware information to SecureSphere.

The following table lists the FireEye configuration tasks.

FireEye Configuration Task Overview

	Action	Description	For more information, see...

	Action	Description	For more information, see...
1	**Configure FireEye MPS to pass information about malware-infected hosts to SecureSphere.**	In FireEye, configure HTTP Server settings so that FireEye sends notifications to the SecureSphere MX.	Configuring FireEye MPS to Pass Information about Malware-infected Hosts to SecureSphere on page 446
2	**Verify that FireEye passes malware information to SecureSphere.**	Verify that: • FireEye generates a test event for SecureSphere. • SecureSphere displays the FireEye test event in **Lookup Data Set: Malware Infected Local IP Addresses**.	Verifying that FireEye Passes Malware Information to SecureSphere on page 448

Configuring FireEye MPS to Pass Information about Malware-infected Hosts to SecureSphere

To configure FireEye MPS to pass information about malware-infected hosts to SecureSphere:

1. Start FireEye.
2. In the top menu, click **Settings**.
3. In the left menu, click **Notifications**.
4. In the **Notification Settings** table, click the http column heading. Fields appear that enable you to add an HTTP Server.
5. In the **Add HTTP Server Name** field, enter a new name that represents the SecureSphere MX.



Note: You use the SecureSphere MX HTTP server name in the next procedure, step 5.

6. Click **Add HTTP Server**.
7. Enter the values to define an HTTP Server for the SecureSphere MX as shown in the table **HTTP Server Values** below.
8. Click **Update**.

HTTP Server Values

Field	Value
Server URL	Enter the following URL: https://<IP address>:8083 /SecureSphere/ui/externalIntegration.html ?source=fireEyeIntegration where <IP address>:8083 points to the SecureSphere MX. Enter this parameter on one line with no spaces. Verify that the URL specifies https, not http.
Username	Enter the malware detection feed user name defined in SecureSphere (malware_feed).
Password	Enter the malware detection feed user password you defined in SecureSphere.
Notification	Select All Events.
Delivery	(Recommended) Select Per event.
Message Format	Select JSON Normal.
SSL Enable	Select the SSL Enable checkbox.

To complete the FireEye configuration:

1. Use SSH to connect to FireEye MPS.
2. Execute the following command:

```
enable
```

3. (Optional) Execute the following command to display the configuration defined in the previous procedure.

   The configuration includes the SecureSphere MX HTTP server name that you defined in the previous procedure, step 5.

```
show fenotify http
```

4. Execute the following command to enter configuration mode:

```
configure terminal
```

5. In the following command, replace `[HTTP Server Name]` with the SecureSphere MX HTTP server name that you defined in the previous procedure, step 5.

   Verify that the rest of the command appears exactly as in the example.

   Enter the entire example on a single command line and execute the command.

```
fenotify http service [HTTP Server Name] provider generic message format
json_legacy-normal
```

6. Execute the following command to display the modified configuration:

```
show fenotify http
```

7. In the modified configuration, verify that *provider-generic-message-format json_legacy-normal* appears under the SecureSphere MX HTTP server name that you defined in the previous procedure, step 5.
8. Execute the following command to exit configuration mode:

```
exit
```

9. Execute the following command to save the configuration:

```
write memory
```



Note: If you make any changes in the FireEye user interface under **Settings>Notifications>http column heading** and click **Update**, you must run this procedure again.

Verifying that FireEye Passes Malware Information to SecureSphere

To verify that FireEye passes malware information to SecureSphere:

1. In FireEye, after completing the procedure configure FireEye MPS to pass information about malware-infected hosts to SecureSphere, select **domain-match** and click **Test-Fire** to generate a test event.
2. In **FireEye > Settings > Notifications**, verify that the following confirmation message appears: **Test-fired domain-match event successfully**
3. In SecureSphere, display Lookup Data Set: Malware Infected Local IP Addresses. See Displaying Local IP Addresses Identified as Infected by Malware on page 449.
4. In SecureSphere, verify that the Lookup Data Set window displays a row for a malware name that begins with **FireEye-TestEvent**, for example, *FireEye-TestEvent-SIG-DM*. You may need to refresh the Lookup Data Set window over a period of five minutes to display the test event malware name.

Understanding the List of Local IP Addresses Identified as Infected by Malware

SecureSphere displays the malware detection feed from FireEye in the **Lookup Data Set: Malware Infected Local IP Addresses** table. The Lookup Data Set lists the local IP addresses infected by malware and characteristics of the malware.

You can modify how the feed is displayed and handled. See Modifying Advanced Settings for Malware Detection Support on page 450.

Based on the threat level of the malware infection, you can apply SecureSphere policies that trigger a defensive action such as audit, alert, or block. See Applying Malware Detection Policies on page 454. Policy actions apply only to malware-infected IP addresses that are included in the **Lookup Data Set: Malware Infected Local IP Addresses** table.

The following table lists the tasks related to the list of local malware-infected IP addresses.

Local Malware-Infected IP Address List Task Overview

	Action	Description	For more information, see...
1	**Display local IP addresses identified as infected by malware.**	Display the **Lookup Data Set: Malware Infected Local IP Addresses** table.	Displaying Local IP Addresses Identified as Infected by Malware on page 449
2	**Understand the Lookup Data Set: Malware Infected Local IP Addresses table.**	Understand the fields of the **Lookup Data Set: Malware Infected Local IP Addresses** table, which lists local IP addresses infected by malware and characteristics of the malware.	Understanding the Malware Infected Local IP Addresses Table on page 449

Displaying Local IP Addresses Identified as Infected by Malware

To display local IP addresses identified as infected by malware:

1. In the **Main** workspace of the SecureSphere MX, select **Setup > Global Objects.**
2. In the **Scope Selection** menu, select **Lookup Data**.
3. In the **Lookup Data Set** pane, under **Data Set Name** select **Malware Infected Local IP Addresses**. The **Lookup Data Set: Malware Infected Local IP Addresses** pane appears.



Note: Do not delete the Malware Infected Local IP Addresses Lookup Data Set. Deleting the Malware Infected Local IP Addresses Lookup Data Set disables malware detection support.

Understanding the Malware Infected Local IP Addresses Table

The following table describes **Lookup Data Set: Malware Infected Local IP Addresses** fields.

Malware Infected Local IP Addresses Data

Field	Description
IP	The address of the local IP that is infected by malware.
Host	Not applicable.
Threat Level	The severity of the threat posed by the malware infection, based on the malware type. See Understanding Threat Levels on page 453.
ID	FireEye identifier for the malware infected local IP.
Feed Source	Location of the data that FireEye passes to SecureSphere.
Malware Type	The category of malware infection, such as malware object or malware callback. See Understanding Malware Types on page 452. Each malware type is assigned a threat level, which you can modify. See Modifying Threat Levels of Malware Types on page 452.
Malware Name	The name of the specific malware infection. To exclude a malware name from the Malware Infected Local IP Addresses list, see Specifying Names and Targets to Exclude from Malware Identification on page 450.
Malware Target	The destination to which the malware transfers information from the infected local IP. To exclude a malware target from the Malware Infected Local IP Addresses list, see Specifying Names and Targets to Exclude from Malware Identification on page 450.
Last Updated	The last date and time when the malware infection data for the local IP was updated in SecureSphere.

Modifying Advanced Settings for Malware Detection Support

SecureSphere displays the malware detection feed from FireEye in the **Lookup Data Set: Malware Infected Local IP Addresses** table. See Understanding the List of Local IP Addresses Identified as Infected by Malware on page 448.

You can use the following advanced settings to modify how the **Lookup Data Set: Malware Infected Local IP Addresses** table is displayed and handled:

- Specify names and targets to exclude from the Lookup Data Set.
- Modify the threat level mapped to malware types.

The following table lists the tasks for advanced settings of malware detection support.

Malware Detection Support Advanced Settings Task Overview

	Action	Description	For more information, see...
1	Specify names and targets to exclude from malware identification.	You can: • Modify the **Lookup Data Set: Malware Name White List**. • Modify the **Lookup Data Set: Malware Target White List**.	Specifying Names and Targets to Exclude from Malware Identification on page 450
2	Modify the threat level mapped to malware types.	After understanding malware threat levels, you can change the threat level mapped to malware types.	Modifying Threat Levels of Malware Types on page 452

Specifying Names and Targets to Exclude from Malware Identification

You may want to exclude certain malware names and targets from the **Lookup Data Set: Malware Infected Local IP Addresses** table. Policy actions will not apply to malware names and targets that are excluded from the table.

You exclude a malware item from the **Lookup Data Set: Malware Infected Local IP Addresses** table by adding it to the a Lookup Data Set white list:

- **Lookup Data Set: Malware Name White List**
- **Lookup Data Set: Malware Target White List**

To restore an excluded malware item to the **Lookup Data Set: Malware Infected Local IP Addresses** table, delete it from the Lookup Data Set white list.

The following table lists the tasks for excluding names and targets from malware identification.

Malware Identification Exclusion Task Overview

	Action	Description	For more information, see...

	Action	Description	For more information, see...
1	Modify the **Lookup Data Set: Malware Name White List.**	You can: • Add a Malware Name to the White List. • Delete a Malware Name from the White List.	Modifying the Malware Name White List on page 451
2	Modify the **Lookup Data Set: Malware Target White List**.	You can: • Add a Malware Target to the White List. • Delete a Malware Target from the White List.	Modifying the Malware Target White List on page 451

Modifying the Malware Name White List

To add a Malware Name to the White List:

1. In the **Main** workspace of the SecureSphere MX, select **Setup > Global Objects.**
2. In the **Scope Selection** menu, select **Lookup Data**.
3. In the **Lookup Data Set** pane, under **Data Set Name** select **Malware Name White List**. The **Lookup Data Set: Malware Name White List** pane appears.
4. In the menu bar, click . A new Malware Name field appears.
5. In the Malware Name field, enter a malware name.
6. Click **Save**.

To delete a Malware Name from the White List:

1. In the **Main** workspace of the SecureSphere MX, select **Setup > Global Objects.**
2. In the **Scope Selection** menu, select **Lookup Data**.
3. In the **Lookup Data Set** pane, under **Data Set Name** select **Malware Name White List**. The **Lookup Data Set: Malware Name White List** pane appears.
4. Under **Malware Name**, select a malware name to delete.
5. In the menu bar, click .
6. Click **Save**.

Modifying the Malware Target White List

To add a Malware Target to the White List:

1. In the **Main** workspace of the SecureSphere MX, select **Setup > Global Objects.**
2. In the **Scope Selection** menu, select **Lookup Data**.
3. In the **Lookup Data Set** pane, under **Data Set Name** select **Malware Target White List**. The **Lookup Data Set: Malware Target White List** pane appears.
4. In the menu bar, click . A new Malware Target field appears.
5. In the Malware Target field, enter a malware target.
6. Click **Save**.

To delete a Malware Target **from the White List:**

1. In the **Main** workspace of the SecureSphere MX, select **Setup > Global Objects.**
2. In the **Scope Selection** menu, select **Lookup Data**.

3. In the **Lookup Data Set** pane, under **Data Set Name** select **Malware Target White List**. The **Lookup Data Set: Malware Target White List** pane appears.
4. Under **Malware Target**, select a malware target to delete.
5. In the menu bar, click .
6. Click **Save**.

Modifying Threat Levels of Malware Types

FireEye categorizes each malware infection according to malware type.

SecureSphere assigns a default threat level to each malware type.

The threat level determines how SecureSphere handles the malware-infected IP address, such as the policy actions that may be taken and whether the IP address is displayed in the **Lookup Data Set: Malware Infected Local IP Addresses** table.

You can change the default threat level mapped to a malware type.

The following table lists the tasks for modifying threat levels of malware types.

Malware Type Threat Level Modification Task Overview

	Action	Description	For more information, see...
1	**Understand malware types.**	Understand the types of malware that SecureSphere handles.	Understanding Malware Types on page 452
2	**Understand threat levels.**	Understand how SecureSphere handles threat levels that can be assigned to malware types.	Understanding Threat Levels on page 453
3	**Change the threat level mapped to malware types.**	Display the FireEye Threat Level Mapping by Malware Type table and select a new threat level for one or more malware types.	Changing the Threat Level of a Malware Type on page 453

Understanding Malware Types

The following table describes the types of malware that SecureSphere handles.

Malware Type Descriptions

Type	Description
Domain Match	A lookup request was made to a known malicious domain. Note that depending on deployment, the source address for Domain Match events may be a Domain Name System (DNS) server.
Infection Match	An internal host made a request for an object that previously was determined to be malicious by the FireEye Multi-Vector Virtual Execution (MVX) engine.
Malware Callback	A machine is infected and is beaconing to its command and control server(s). The profiles responsible for generating Malware Callback events can come from the FireEye Dynamic Threat Intelligence (DTI) cloud and from locally generated profiles based on MVX dynamic callback extraction.
Malware Object	A malicious binary, DLL, or archive file was seen in the clear across the wire and was determined to be malicious by the MVX engine.
Unidentified Type	An unrecognized malware type. Used for new malware types and when FireEye did not identify the malware type.
Web Infection	A malicious web page or other high frequency object (or collection of objects) targeting the browser or a browser plugin was requested by an internal host and was determined to be malicious by the MVX engine.

Understanding Threat Levels

The following table describes:

- The threat levels that can be assigned to a malware type
- The policy actions available for the threat level
- Whether the **Lookup Data Set: Malware Infected Local IP Addresses** displays a local IP address that is identified with the threat level

Threat Level Descriptions

Level	Policy Actions	Displayed in Lookup Data Set?
Ignore	None.	No
Low	None.	Yes
Medium	• Audit sensitive data access only. • Alert when dangerous high risk scenarios occur.	Yes
High	• Audit all relevant data for forensics. • Alert when probable malware actions occur. • (Optional) Block dangerous high risk scenarios	Yes

Changing the Threat Level of a Malware Type

To change the threat level of a malware type:

1. In the **Admin** workspace of the SecureSphere MX, select **System Definitions**.

2. In the **System Definitions** pane, under Management Server Settings select **External Integration Settings**. The **External Integration Settings** window appears with the Fireeye Threat Level Mapping by Malware Type table. The columns display:
   - The malware types. For malware type values and descriptions, see Understanding Malware Types on page 452.
   - The value of the threat level assigned to each malware type. For threat level values and descriptions, see Understanding Threat Levels on page 453.
   - The default value of the threat level for each malware type
3. In the **Value** column, click a value to display a drop-down list and select a new value.
4. Click **Save**.

Applying Malware Detection Policies

SecureSphere enables you to apply policies with defensive actions against the malware-infected host IPs that FireEye MPS identifies, which are displayed in the **Lookup Data Set: Malware Infected Local IP Addresses** table.

Based on the threat level of the malware infection, you can apply recommended SecureSphere policies that trigger an audit, alert, or block.

You can also create and apply custom policies in SecureSphere using security match criteria that trigger a defensive action.

For more information on SecureSphere policies, see Security Policies on page 166.

The following table lists the malware detection policy tasks.

Malware Detection Policy Task Overview

	Action	Description	For more information, see...
1	**Understand recommended malware detection policies.**	Understand recommended policies that trigger a defensive action when malware-infected local IP addresses are identified.	Understanding Recommended Malware Detection Policies on page 454
2	**Apply recommended malware detection policies.**	Apply recommended policies in SecureSphere to trigger a defensive action when malware-infected local IP addresses are identified.	Applying Recommended Malware Detection Policies on page 455
3	**Define and apply custom policies.**	Define and apply custom policies in SecureSphere using security match criteria that trigger a defensive action when malware-infected local IP addresses are identified.	Defining Custom Malware Detection Policies on page 458

Understanding Recommended Malware Detection Policies

The following table describes recommended malware detection policies, including the policy type and the threat level that the policy addresses.

To achieve maximum protection against malware-infected IP host addresses, apply the recommended malware detection policies. Some policies require additional configuration as noted under Description.



Note: Policies that alert perform one of the following actions: a. Alert only; b. Alert and block. Policies with a blocking action are optional. For each policy with a blocking action, there is an equivalent policy with an alert only action. To ensure that policies operate efficiently, do not activate equivalent policies with different actions (*alert only* and *alert and block*) at the same time.

Recommended Malware Detection Policies for Web

Policy Type	Threat Level	Policy Name	Description
Alert	High	Infected Host - Web - Authentication URL	Alerts when an infected machine accesses a login page.
Alert	High	Infected Host - Web - Block Authentication URL	Alerts and blocks when an infected machine accesses a login page.
Alert	High	Infected Host - Web - POST Request	Alerts when POST requests originate from an infected machine.
Alert	High	Infected Host - Web - Block POST Request	Alerts and blocks when POST requests originate from an infected machine.
Enrichment	All	Malware Infected Hosts FireEye Information Enrichment (Web)	Adds fields from **Lookup Data Set: Malware Infected Local IP Addresses** to any event that matches an audit or alert policy.

Applying Recommended Malware Detection Policies

SecureSphere provides recommended malware detection policies. To use the recommended policies, you download the policies and individually import and apply each policy.

The following table lists the tasks for accessing and applying recommended malware detection policies.

Accessing and Applying Recommended Malware Detection Policies Task Overview

	Action	Description	For more information, see...
1	**Download recommended malware detection policies.**	Download recommended policies from the SecureSphere FTP site.	Downloading Recommended Malware Detection Policies on page 455
2	**Import recommended malware detection policies.**	Upload and import each recommended policy into SecureSphere.	Importing Recommended Malware Detection Policies on page 456
3	**Apply recommended malware detection policies.**	Apply each recommended policy to relevant services.	Applying Recommended Malware Detection Policies on page 457

Downloading Recommended Malware Detection Policies

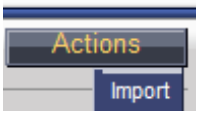
To download recommended malware detection policies:

1. Download the policy zip file from the Imperva FTP site ftp-us.imperva.com, under **/Downloads/SecureSphere_Patches/[SecureSphereVersion]/FireEye/**.
2. Extract the .imf files from the policy zip file.

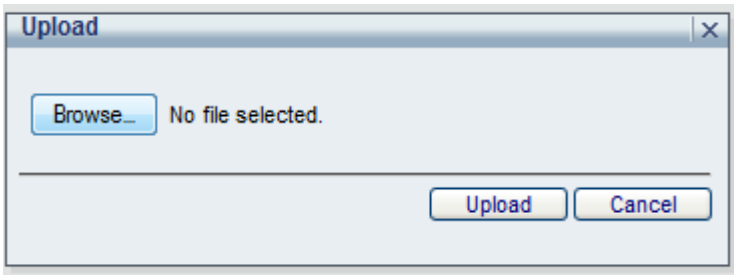
Importing Recommended Malware Detection Policies

To import each recommended malware detection policy:

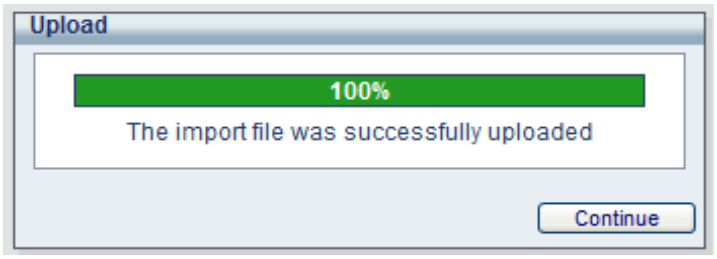
1. In the **Main** workspace of the SecureSphere MX, select **Policies**.
2. Click **Actions** and select **Import**.



The **Upload** window appears.

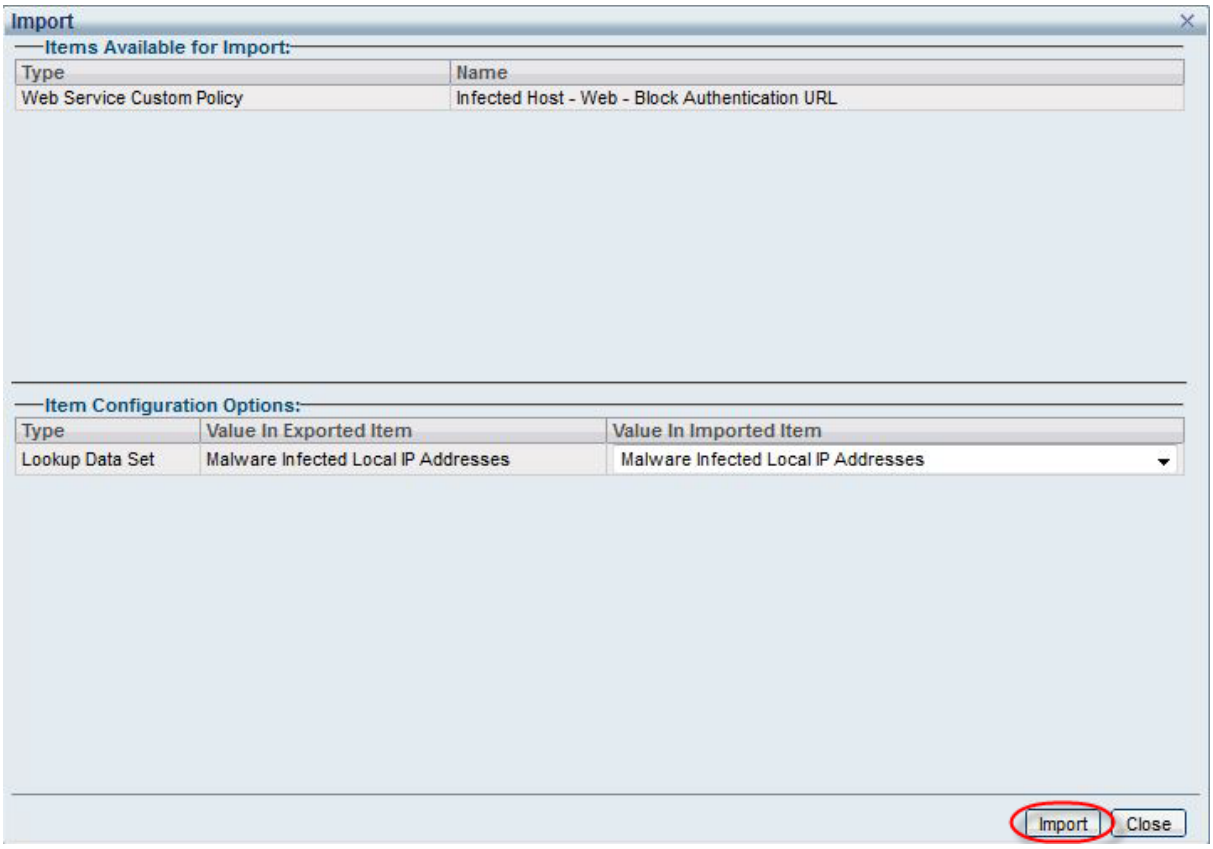


3. In the **Upload** window, click **Browse** to select a file to upload.
4. In the **File Upload** window, select an .imf file from the folder where you extracted the recommended policies, and click **Open**. The selected file name appears in the **Upload** window.
5. In the **Upload** window, click **Upload**.
6. In the **Upload** window, when the file is successfully imported click **Continue**.



The **Import** window displays details of the uploaded policy file.

7. In the **Import** window, click **Import**.



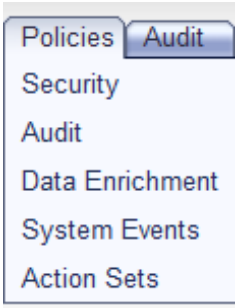
8. In the **Import Finished Successfully** message, click **Close**.

Applying Recommended Malware Detection Policies

Some recommended malware detection policies require additional configuration. See the policy descriptions in Understanding Recommended Malware Detection Policies on page 454. Complete the configuration before applying a policy.

To apply each recommended malware detection policy:

1. In the **Main** workspace of the SecureSphere MX, select **Policies** and the policy type tab.



   - For an Alert policy, select **Policies > Security**.
   - For an Audit policy, select **Policies > Audit**.
   - For an Enrichment policy, select **Policies > Data Enrichment**.

   To verify the policy type, see the **Policy Type** column in Understanding Recommended Malware Detection Policies on page 454.

2. In the Policies pane, select the policy name to display the policy details.

3. Apply the policy to relevant services. See Applying Policies on page 318.

Defining Custom Malware Detection Policies

To create and use a custom malware detection policy:

1. Create a user-defined policy. See Adding New Security Policies on page 179.
2. Configure the **Data Set: Attribute Lookup** match criteria with the settings in the table below.
3. Select one or more additional security match criteria.
4. Save and activate your custom policy.

Data Set: Attribute Lookup Match Criteria Settings

Setting	Value
Lookup Data Set	Malware Infected Local IP Addresses
Field	IP
Attribute	Threat Level
User Defined Values	Enter the applicable Threat Level value (in upper case): • MEDIUM • HIGH

CHAPTER 13

Reverse Proxies

A Reverse Proxy is a machine through which Web traffic passes enroute to and from a Web server. For Web traffic only, a non-sniffing SecureSphere gateway can be configured as a Reverse Proxy.

There are three kinds of Reverse Proxies:

- Transparent Reverse Proxy (TRP)
- Two kinds of non-Transparent Reverse Proxies:
  - Kernel Reverse Proxy (KRP)
  - Apache Reverse Proxy (ARP)

This section describes how to configure Transparent Reverse Proxy (TRP) and Kernel Reverse Proxy (KRP) in the SecureSphere GUI.

Additional configuration using `impcfg` is also required. For more information, see **SecureSphere Administration Guide**.

This chapter contains the following sections:

Kernel (Non-Transparent) Reverse Proxy 459

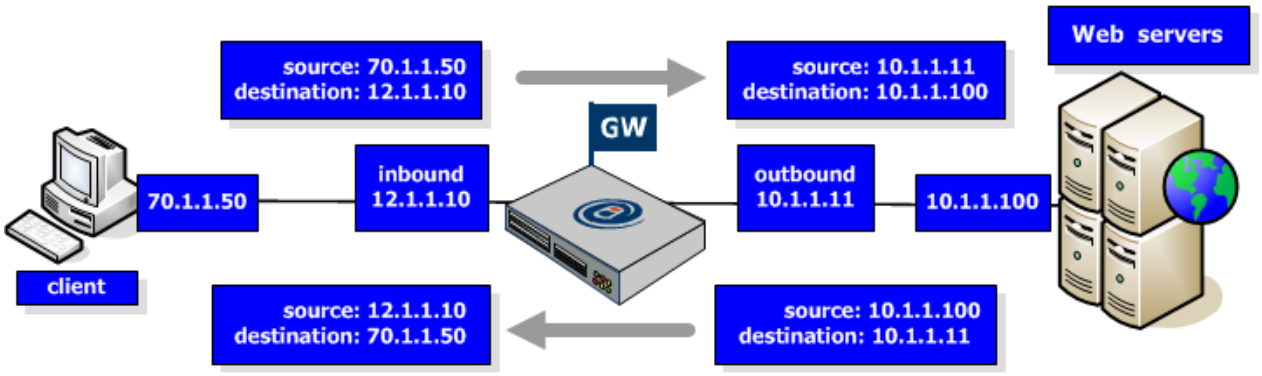
Transparent Reverse Proxy 462

Kernel (Non-Transparent) Reverse Proxy

In a Kernel Reverse Proxy deployment, the client begins the session by sending packets to the SecureSphere Gateway's inbound IP address. The SecureSphere Gateway inspects the packets and, based on the rules, opens a second connection to the server, changing the packet's source IP address to the SecureSphere Gateway's outbound IP address and the destination IP address to the real IP address of the server.

The server sends the return packets to the SecureSphere Gateway, which inspects the packets, changing the source IP address to the SecureSphere Gateway's inbound IP addresses and the destination IP address to that of the client, and sends the modified packets to the client.

Equation 1: Non-Transparent Reverse Proxy Deployment



Packets that do not match any of the decision rules are ignored. You may wish to define a default rule that sends these packets to a server that generates a customized error message for the client.

In addition, URL rewrite groups can be used to rewrite or redirect responses and/or requests. For more information, see URL Rewrite on page 428.

For more information about Transparent Reverse Proxy, including how it can be configured together with Apache Reverse Proxy on the same appliance, see **SecureSphere Administration Guide**.

Configuring a Kernel (Non-Transparent) Reverse Proxy 460

Configuring a Kernel (Non-Transparent) Reverse Proxy



Note: For information on configuring client certificate validation in Kernel Reverse Proxy mode, see Validating SSL Client Certificates on page 436.

To configure a Kernel (Non-Transparent) Reverse Proxy:

1. In `impcfg`, configure the Gateway appliance as a Kernel Reverse Proxy.

   For more information about `impcfg`, see the **SecureSphere Administration Guide**.
2. After the Gateway appliance successfully completes its reconfiguration as a Kernel Reverse Proxy, open the SecureSphere GUI.
3. In the **Main** workspace, select Setup > Gateways.
4. In the **Filter** pane, select **By Mode**, **By Platform** or **By Server Group**.
5. Select the Gateway you configured as a KRP in `impcfg`.
6. In the **Gateway Details** tab, configure the aliases.

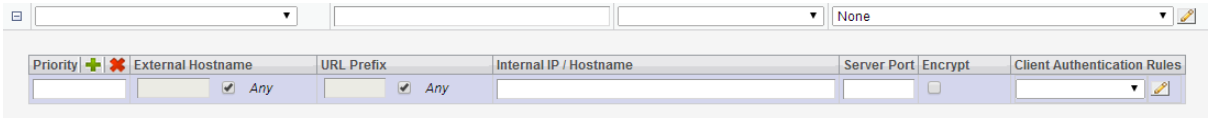
   An alias represents the inbound IP address and the outbound IP address for the KRP. Since networks tend to be dynamic, IP address allocation may change from time to time. Aliases eliminate the need to configure changes in two places (on the Gateway and on the Management Server). For multiple end points (servers), you can use port mapping (one Gateway IP address with multiple ports). Alternatively, use an alias representing a Reverse Proxy IP address per server so that whenever an alias is addressed, the communication will be forwarded to the corresponding server.

   For information on how to configure aliases and routing for KRP, see the "Gateways" chapter in the **SecureSphere Administration Guide**.

7. In the **Main** workspace, select Setup > Sites.
8. In the **Sites Tree** pane, select a server group and then a Web service.
9. In the **HTTP Service** pane, in the **Reverse Proxy** tab, click the icon to the right of the words **Gateway IP Alias**.
10. Enter the parameters, as described in the table **Kernel Reverse Proxy Parameters** below.

Note: A connection entering the Gateway which does not match any of these entries is ignored by the Gateway.

11. Next, create a decision rule which defines how to handle incoming Web traffic by clicking the icon to the right of the word **Priority**.



12. Enter the parameters, as described in the second table below titled **Kernel Reverse Proxy Decision Rules Parameters**.
13. Add additional decision rules as required.

Kernel Reverse Proxy Parameters

Parameter	Description
Gateway IP Alias	From the drop-down menu, select one of the aliases that were defined for this gateway in `impcfg`. For more information about `impcfg`, see **SecureSphere Administration Guide**.
Gateway Ports	Enter the port number on which the Kernel Reverse Proxy listens. If the Kernel Reverse Proxy listens on more than one port, and they are all encrypted or all not encrypted, then you can enter a comma-separated list of the ports. Otherwise, you must define a separate row for each port. You can use the same Gateway IP Alias name for more than one port.
Server Certificate	Select the certificate which SecureSphere should present to the client from the list of the certificates uploaded for the gateway. For more information, see Adding SSL Keys on page 73.
Client Authentication Authorities	Select a Certificate Authority Group to associate with this web server. In order you do this, you need to have imported Public Certificate Authority certificates and assigned them in a Certificate Authority Group. For more information see Associating Client Certificate Validation on page 438.
Report forwarded client IP in HTTP header	If selected, traffic forwarded to the Web server will include the client IP address in the header specified in **Header Name**. If not selected, the Web server will see this Reverse Proxy as the client.

Kernel Reverse Proxy Decision Rules Parameters

Parameter	Description
Priority	The decision rule's priority. You can define any number of decision rules, and they are evaluated sequentially in order of their priorities. You must assign different priorities to each rule in the group, in order to avoid ambiguity in the order of their application. Also, it is recommended that you assign priorities in multiples of ten, that is, 10, 20, 30 and so on, so that you can insert additional decision rules between the original decision rules at a later date, if required.
Host	Specify the host name for which this rule will be applied, or select **Any** for all hosts. **Note:** This field is understood by SecureSphere as a regular expression. So to specify "www.site.com" you should enter "www\.site\.com". "www\.site\.(org|co)\.uk" means "www.site.co.uk" or "www.site.org.uk". Wildcards (for example, "*" and "?") cannot be used in this field. For more information about regular expressions, see Regular Expressions on page 475.
URL prefix	Specify the prefix of URLs (for example, **/login/**) for which traffic is to be directed to **Server IP**, or select **Any** for all URLs. For more information about regular expressions, see Regular Expressions on page 475.
Internal IP / Hostname	The IP address or the hostname of the Web server to which traffic that meets the **Host** and **URL prefix** conditions is forwarded.
Server port	The port number on the Web server to which traffic that meets the **Host** and **URL prefix** conditions is forwarded.
Encrypt	Select **Encrypt** to encrypt the connection between the SecureSphere gateway and the backend (the real) Web server.
Client Authentication Rules	When working with client certificate authentication, select the Client Authentication Rules that determine the course of action taken when certificate validation succeeds or fails. For more information on Client Authentication Rules, see Configuring Client Authentication Rules on page 438.



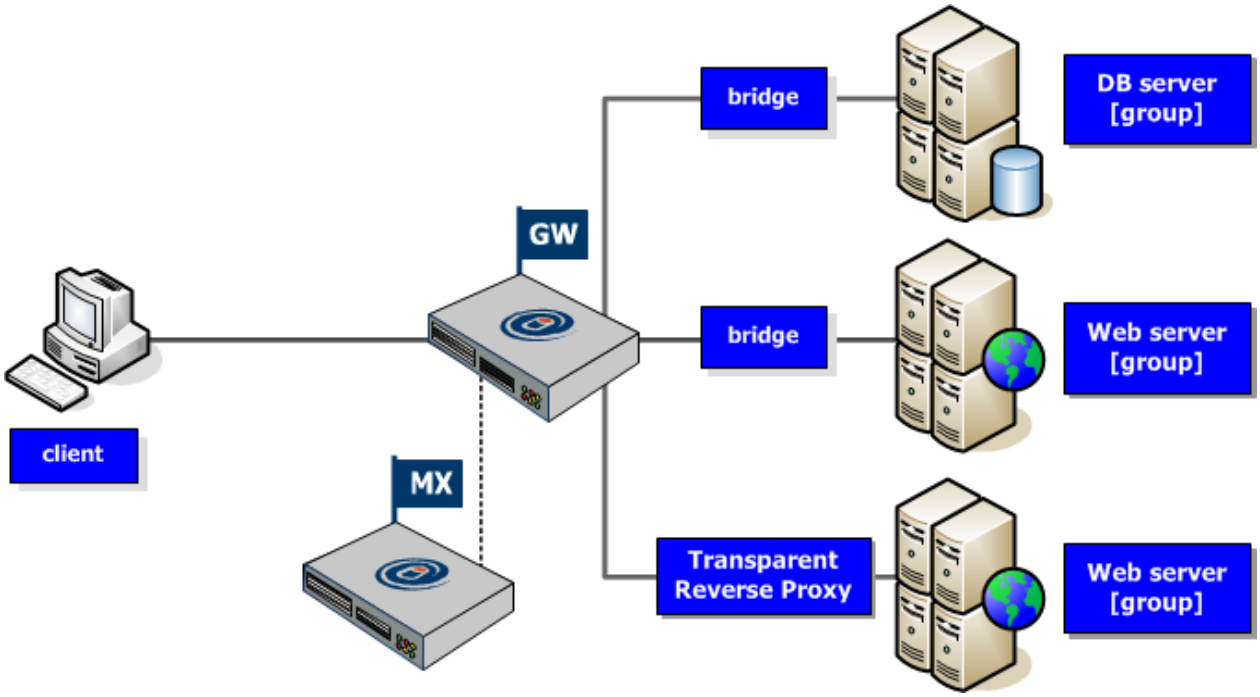
Note: When using the **Host** and **URL** prefix together, the operation between them is AND. While typically the operation between match criteria is OR..

Transparent Reverse Proxy

In Transparent Reverse Proxy mode, the client is unaware of the Reverse Proxy, and connects directly to the server. The SecureSphere gateway is not visible to the client or to the server, which see each others' real IP addresses.

The SecureSphere gateway is in bridge mode, but additionally functions as a Reverse Proxy (by transparently hijacking the client connection, terminating it and opening a new connection to the backend server on behalf of the client) for the servers specified in the SecureSphere GUI (in the **Transparent Reverse Proxy** section of the **Definitions** tab of the server group (**Main** > **Setup** > **Sites**). All other traffic, including non-Web traffic, is bridged through. In this way, the same SecureSphere gateway protecting many Web and database server groups can act as a Reverse Proxy for some or all of the Web server groups it protects, and as a bridge for the other server groups.

Equation 2: Transparent Reverse Proxy - Selected Web Servers



In addition, URL rewrite groups can be used to rewrite or redirect responses and/or requests. For more information, see URL Rewrite on page 428.

For more information about Transparent Reverse Proxy, see **SecureSphere Administration Guide**.

Configuring a Transparent Reverse Proxy

To configure a Transparent Reverse Proxy:

1. If the appliance is not already in bridge mode, then in `impcfg`, configure the appliance as a bridge.

   For more information about `impcfg`, see **SecureSphere Administration Guide**.
2. After the appliance successfully completes its reconfiguration as a bridge, open the SecureSphere GUI.
3. In the **Main** workspace, select Setup > Sites.
4. In the **Sites Tree** pane, select a server group and then a Web service.
5. In the **HTTP Service** pane, in the **Transparent Reverse Proxy** section of the **Definitions** tab, select **Enable Transparent Reverse Proxy**.
6. Next, create a decision rule which defines how to handle incoming Web traffic by clicking the  icon to the right of the words **Server IP**.
7. Enter the parameters, as described in the table below.

Notes:

- All traffic not directed to one of the specified Ports on Server IP address is bridged and inspected according to the server group, service and application applied rules.
- In the event of a conflict between a Transparent Reverse Proxy rule and another service's port configuration, the Transparent Reverse Proxy rule takes precedence.

8. Add additional rules as required.

Note: Ignore IP Group and **Limit monitoring to IP group** are ignored for Reverse Proxy.

Transparent Reverse Proxy Parameters

Parameter	Description
Server IP	The IP address of the Web server for which this SecureSphere gateway acts as a Transparent Reverse Proxy.
Ports	A comma-separated list of port numbers on which the Transparent Reverse Proxy listens.
Certificate	Select the certificate which SecureSphere should present to the client from the list of the certificates uploaded for the gateway. For more information, see Adding SSL Keys on page 73.
Server Side Port	The port on the Web server to which to direct Web traffic.
Encrypt Server Connection	Select **Encrypt** to encrypt the connection between the SecureSphere gateway and the Web server.

APPENDIX A

MXs Managed by SOM

SecureSphere Security Operations Manager (SOM) is a multi-domain management solution that dramatically improves the operational efficiency of managing SecureSphere deployments with multiple MX Management Servers.

This appendix presents how to work with SOM and it contains the following sections:

Working with SOM 465
Registering MXs 466

Working with SOM

SOM consolidates the management, visibility and reporting across multiple SecureSphere MX Management Servers, and provides system wide health metrics and statistics. Various objects can be configured on SOM and then downloaded to all the MXs registered to that SOM. Once the object was downloaded to MX, it is presented on MX as SOM object and it is locked for editing in most of cases.

You can apply the following objects defined on SOM: policies, action sets, reports, global objects and signatures. Objects downloaded from SOM are displayed in the MX GUI with the special icon which indicates that this object was defined on SOM and only limited changes are allowed.

Policies defined on SOM are applied to MXs using the following two flows:

- **Mandatory**: Enables sending the policies defined on SOM to MX and running them immediately on all the server groups/services/applications. In this flow policies downloaded from SOM cannot be unapplied.
- **Optional**: Enables sending the policies defined on SOM to MX and keeping them there without applying them, or applying them only to a part of the services there. Once the policy is on MX, you can decide if you want this policy to run now or later.

Scheduling for reports and audit policies can be defined on SOM and downloaded to MXs or defined locally on each MX.

For more information about working with SOM, refer to the SOM User Guide.

Registering MXs

In order to allow SOM to manage an MX, you need to register that MX on SOM.

All the MXs that have been registered on that SOM are presented in the Registered MXs table.

Registered MXs

Registered MXs Parameters	Description
Name	Read only MX name.
Status	SOM to MX communication status. The following options are available: • **Running**: The MX is registered on SOM and it is up and running. • **Disconnected**: The MX is registered on SOM and it is down. • **Not Registered**: The MX is not registered on SOM.
Synchronization Status	SOM to MX synchronization status. The following options are available: • **Synchronized**: MX successfully received all the data from SOM, so the configuration on MX is fully synchronized with SOM. • **Synchronizing**: MX is a process of receiving full configuration update from SOM. • **Not Synchronized**: There was an attempt to send a configuration update from SOM to MX and MX could not receive it. • **Permanently Not Synchronized**: After 10 unsuccessful attempts to send the configuration to MX, SOM stops trying.
Version	MX version number.
License	The product licenses that are installed on MX.
Last Update Time	When was the last time that MX responded to status check.

During the registration process you need to define the level of security required for the SOM - MX authentication and communication, as follows:

- **SSL with Certificate**
- **SSL without Certificate**
- **Do not use SSL**

Setting Up SSL with Certificate Authentication

Using SSL with Certificate provides authentication and data encryption. This method is recommended when SOM and MX are located in different local networks and sensitive data is sent.

In order to establish the communication between SOM and MX with authentication and encryption, SOM and MX must present certificates to each other. You need to import or generate a SOM certificate before registering a new MX. Once the certificate is ready, you need to export it in order to send it to MX.

Using SSL with Certificate Task Overview

	Action	Description	For more information, see...
1	**On SOM: Generate a SOM certificate**	Generate a SOM certificate that will be used to communicate with MXs.	Setting Up SSL with Certificate Authentication on MX on page 469
2	**On SOM: Export the new SOM certificate**	Export the new certificate into a file that will be used on MX and copy this file to the MX machine.	Setting Up SSL with Certificate Authentication on MX on page 469
3	**On MX: Set the Communication Port**	The port 8084 is usually used for SSL with Certificate communication option.	See the procedure **To define the communication port on MX** in Setting Up SSL with Certificate Authentication on SOM on page 467
4	**On MX: Generate an MX certificate**	Generate an MX certificate that will be used to communicate with SOM.	See the procedure **To generate an MX certificate** in Setting Up SSL with Certificate Authentication on MX on page 469
5	**On MX: Export the new MX certificate**	Export the new certificate into a file that will be used on SOM and copy this file to the SOM machine.	See the procedure **To export MX certificate** Setting Up SSL with Certificate Authentication on SOM on page 467
6	**On MX: Upload the SOM certificate to MX**	Upload the SOM certificate to MX.	See the procedure **To upload SOM certificate to MX** in Setting Up SSL with Certificate Authentication on SOM on page 467
7	**Optional - On SOM: Upload the MX certificate to SOM**	Upload the MX certificate to SOM.	See the procedure To upload MX certificate Setting Up SSL with Certificate Authentication on MX on page 469
8	**On SOM: Register a new MX**	Register a new MX on SOM and during the registration process select the SSL with Certificate option.	See the procedure **To register a new MX with SSL certificate** in Setting Up SSL with Certificate Authentication on MX on page 469

Setting Up SSL with Certificate Authentication on SOM

To generate a SOM certificate:

1. In the **Main** workspace, click **Setup** > **Settings** > **SOM to MX Communication Settings**. The **SOM to MX Communication Settings** pane appears.
2. In the **SOM Certificate details** section, click **Replace Certificate**. The **Generate New SOM Certificate** dialog box appears.
3. Click on **Click Here**, the **Upload Certificate** dialog box appears.
4. If you have a valid certificate in the PFX format, do one of the following:
   - Click **Browse** next to the **PFX file** box and type the password.
   - If a certificate has been previously installed on the machine and you want to overwrite it, click **Overwrite Existing Certificate**.

- Click **Upload**. The **Upload** dialog box appears with the status bar that presents the progress of the upload.

5. If you do not have a valid certificate, in the **Generate New SOM Certificate** dialog box, set the period of time in which the certificate is valid using the **From/To** options. The default for **From** is today. The default for **To** is a year from today.
6. Click **Generate**. The **Generate Certificate** status bar appears, presenting the progress of the certificate generation.

To export SOM certificate:

1. In the **Main** workspace, click **Setup** > **Settings** > **SOM to MX Communication Settings**. The **SOM to MX Communication Settings** pane appears.
2. In the **SOM Certificate details** section, click **Export Certificate**. The **Export Certificate** dialog box appears presenting the export status bar.
3. Once the export process is completed, click **Download Certificate** to open/save the certificate file. Depending on your browser settings, the certificate is downloaded either to a predefined folder or to the folder that you specify.
4. Send the file to the MX machine using your preferred file transfer method (use a thumb drive, shared drive or network file copy).

To upload MX certificate:

1. In the **Main** workspace, click **Setup** > **Settings** > **SOM to MX Communication Settings**. The **SOM to MX Communication Settings** pane appears.
2. In the **SOM Certificates** table, click **Create New**. The **Upload Trusted Certificates** dialog box appears.
3. In the **Upload Trusted Certificates** dialog box, click **Browse** and select the file to upload.
4. If you want the new certificate to overwrite an existing one, select **Overwrite Existing Certificate**.
5. Click **Upload**. The uploaded MX certificate appears in the **SOM Certificates** table.

To register a new MX with SSL certificate:

1. In the **Main** workspace, click **Setup** > **Registered MXs**. The **Registered MXs** window appears.
2. In the **Registered MXs** window, click **Add**. The **Add New MX** dialog box appears.
3. In the **Add New MX** dialog box, type the name of the MX that you want to add and click **Create**. The **Add New MX** dialog box closes and the MX details pane displays the name of the new MX.
4. In the MX details pane, set the following general settings:
   - **Name**: The name of the new MX as you want it to appear on SOM and on MX.
   - **Host Name/IP Address**: The host name or the IP address of the new MX.
   - **Username**: MX's admin username that is used to enter SecureSphere GUI.
   - **Password/Verify Password**: MX's admin password that is used to enter SecureSphere GUI.
5. Define the following **Communication Settings**, as described in the table below.
6. Define the following **Drilldown Settings**:
   - Drilldown Security Options: Enables navigating into the MX with the following methods:
     - **Use SSL**: Browsing MX using HTTPS.
     - **Do not Use SSL**: Browsing MX using HTTP.
   - Drilldown Port:The port that is used by SOM to navigate to MX.
7. To save your setting at any configuration stage, click **Save**.
8. To save your settings and to complete the first time registration, click **Save & Register**. The new MX appears in the **Registered MXs** table.
9. To save any changes that are performed on a registered MX, click **Save**.
10. To verify that the new settings have been applied, click **Test Connections**.

Communication Settings

Setting	Description
Authentication Options:	Select **SSL with Certificate**.
Verify MX Host Name:	Enables verifying the host. SOM makes sure that this host is really who it says it is, as certificate itself does not prove that. SOM verifies the host name of the received certificate by resolving its address using DNS and comparing it to sender's address. This option is recommended when you suspect that the MX certificate may be leaked to malicious users who may try to use it to establish connection with SOM. In this case use this option and ensure that DNS will correctly resolve the address of the MX.
Communication Port:	The port that is used to communicate with the MX. Select **8084**. You can set port numbers that is different from the default value. **Note:** The communication port numbers must be the same on SOM and MX. It is needed to set the communication port on MX in Setup>Settings>SOM to MX Communication Settings>Communication Port.

Setting Up SSL with Certificate Authentication on MX

To define the communication port on MX:

1. In the **Main** workspace, click **Setup** > **Settings** > **SOM to MX Communication Settings**. The **SOM to MX Communication Settings** pane appears.
2. Set the **Communication Port** parameter to 8084.



Note: The communication port numbers must be the same on SOM and MX. 8084 is the default port number. You can define a different port number for this authentication method on both, SOM and MX. For the instructions on how to use a communication port different from 8084 for this authentication method on MX, refer to the Knowledge Base or contact Imperva Technical Support.

To generate an MX certificate:

1. In the **Main** workspace, click **Setup** > **Settings** > **SOM to MX Communication Settings**. The **SOM to MX Communication Settings** pane appears.
2. In the **MX Certificate details** section, click **Replace Certificate**. The **Generate New MX Certificate** dialog box appears.
3. Click on **Click Here**, the **Upload Certificate** dialog box appears.
4. If you have a valid certificate in the PFX format, do one of the following:
   - Click **Browse** next to the **PFX file** box and type the password.
   - If a certificate has been previously installed on the machine and you want to overwrite it, click **Overwrite Existing Certificate**.
   - Click **Upload**. The **Upload** dialog box appears with the status bar that presents the progress of the upload.

5. If you do not have a valid certificate, in the **Generate New MX Certificate** dialog box, set the period of time in which the certificate is valid using the **From/To** options. The default for **From** is today. The default for **To** is a year from today.
6. Click **Generate**. The **Generate Certificate** status bar appears, presenting the progress of the certificate generation.

To export MX certificate:

1. In the **Main** workspace, click **Setup** > **Settings** > **SOM to MX Communication Settings**. The **SOM to MX Communication Settings** pane appears.
2. In the **MX Certificates Details** section, click **Export Certificate**. The **Export Certificate** dialog box appears presenting the export status bar.
3. Once the export process is completed, click **Download Certificate** to open/save the certificate file. Depending on your browser settings, the certificate is downloaded either to a predefined folder or to the folder that you specify.
4. Send the file to the SOM machine using your preferred file transfer method (use a thumb drive, shared drive or network file copy).

To upload SOM certificate to MX:

1. In the **Main** workspace, click **Setup** > **Settings** > **SOM to MX Communication Settings**. The **SOM to MX Communication Settings** pane appears.
2. In the **SOM Certificates** table, click **Create New**. The **Upload Trusted Certificates** dialog box appears.
3. In the **Upload Trusted Certificates** dialog box, click **Browse** and select the file to upload.
4. If you want the new certificate to overwrite an existing one, select **Overwrite Existing Certificate**.
5. Click **Upload**. The uploaded SOM certificate appears in the **SOM Certificates** table.

SSL Certificate Expiration

Certificates are valid for a restricted period of time. SecureSphere provides the following capabilities for dealing with certificate expiration:

- A System Event is generated as the certificate is close to the expiration day. The event is generated on SOM when the SOM certificate is about to reach its expiration, and on each MX as its own certificate is close to the expiration day.
- A new certificate may be automatically generated when the existing one expires.

To define SSL certificate expiration on SOM and on MX:

1. On SOM/MX, in the **Main** workspace, click **Setup** > **Settings** > **SOM to MX Communication Settings**. The **SOM to MX Communication Settings** pane appears.
2. In the **SOM to MX Communication Settings** pane, set the items in the table **SSL Certification Expiration Settings** below.
3. Click **Save**.

SSL Certification Expiration Settings

Parameter	Description
Certificate Validity (read-only):	Presents the validity that was defined when this certificate was created.
System Warning Before Certificate Validity Expires (In Days):	Enables to define how many days before the certificate expires a System Event will be generated
Generate Certificate Automatically When Certificate Validity Expires:	Enables generating a new certificate automatically once the validity has been expired. One day before the validity expires, SOM automatically generates a new certificate. The period for which the new certificate is valid is defined in the **Auto Certificate Validity**.
Auto Certificate Validity:	Enables to define certificate validity for SSL certificates that were automatically generated by SOM.

Setting Up the Other Authentication Options

In addition to the SSL with Certificate, the following authentication options are available:

- **SSL without Certificate**: This method is recommended when SOM and MX are located in the same local network, so that the authentication process is not required, but it is important to encrypt the sensitive data sent within this network.
- **Do not use SSL**: This method is recommended when SOM and MX are located in the same local network and there is no exchange of sensitive data between them.



Note: When using the Do not use SSL authentication option, MX re-connect to SOM after MX-HA Fail-Over will not succeed.

To register a new MX using SSL without Certificate/Do not use SSL options:

1. In the **Main** workspace, click **Setup** > **Registered MXs**. The **Registered MXs** window appears.
2. In the **Registered MXs** window, click **Add**. The **Add New MX** dialog box appears.
3. In the **Add New MX** dialog box, type the name of the MX that you want to add and click **Create**. The **Add New MX** dialog box closes and the MX details pane displays the name of the new MX.
4. In the MX details pane, set the following general settings:
   - **Name**: The name of the new MX as you want it to appear on SOM and on MX.
   - **Host Name / IP Address**: The host name or the IP address of the new MX.
   - **Username**: MX's admin username that is used to enter SecureSphere GUI.
   - **Password / Verify Password**: MX's admin password that is used to enter SecureSphere GUI.
5. Define the following **Communication Settings** as described in the table below.
6. Define the following **Drilldown Settings**:Enables navigating into the MX with the following methods:
   - **Drilldown Security Options**:
     - **Use SSL**: Browsing MX using HTTPS.
     - **Do not Use SSL**: Browsing MX using HTTP.
   - **Drilldown Port**:The port that is used by SOM to navigate to MX.

7. To save your setting at any configuration stage, click **Save**.
8. To save your settings and to complete the first time registration, click **Save & Register**. The new MX appears in the **Registered MXs** table.
9. To save any changes that are performed on a registered MX, click **Save**.
10. To verify that the new settings have been applied, click **Test Connections**.

Communication Settings

Parameter	Description
Authentication Options:	During the registration process you need to define the level of security required for the SOM - MX authentication and communication, as follows: • **SSL with Certificate**: Enables authentication and data encryption, see Setting Up SSL with Certificate Authentication on page 466. • **SSL without Certificate**: Enables data encryption, no authentication. Port 8083. • **Do not use SSL**: No authentication and no data encryption. Port 8080.
Communication Port:	The port that is used to communicate with the MX. This value depends on the communication option selected. The following are the default values: • **SSL with Certificate**: 8084. • **SSL without Certificate**: 8083. • **Do not Use SSL**: 8080. You can set port numbers that are different from the default values. **Note:** The communication port numbers must be the same on SOM and MX. It is needed to set the communication port in MX in Setup>Settings>SOM to MX Communication Settings>Communication Port.

Removing MXs from SOM

All the MX registered to SOM appear in the Registered MXs table on SOM. Removing an MX from SOM, simply means removing this MX from the Registered MXs table.

On MX you can see the SOM connection status in the Monitor > Dashboard, as follows:

- When MX is registered to SOM and they are connected, the SOM Status pane displays: Connected to SOM.
- When MX is removed from SOM using the Registered MX table option on SOM, the SOM Status pane does not appear in the Dashboard on MX.
- When the communication between SOM and MX is lost, MX users can detatch MX from SOM using the Disconnect option that appears in the SOM Status pane in the Dashboard on MX.

To remove MX from SOM:

1. On SOM, in the **Main** workspace, click **Setup** > **Registered MXs**. The **Registered MXs** window appears.
2. In the **Registered MXs** table, select the MX that you want to remove and click **Remove**.

APPENDIX B

Writing Signatures and Regular Expressions

This Appendix describes the SecureSphere signature language. SecureSphere includes a signature detection and prevention layer.

SecureSphere comes with a preconfigured database of signatures, which is updated automatically from the Imperva Web sites on a regular basis. In addition to these preconfigured and automatically downloaded signatures, users can write their own signatures to detect and block communication consisting of specific information. This appendix explains the language in which such signatures should be written. The language resembles a Snort™ signature language, though there are some differences.

This appendix reviews the following:

Writing Signatures 473
Regular Expressions 475

Writing Signatures

The following sections describe how to write signatures, including:

- Single Part Signatures on page 474
- Multi Part Signatures on page 474
- Absolute Modifiers on page 474
- Relative Modifiers on page 474

Single Part Signatures

The basic signature unit is part, which contains the actual signature text.

Example: The following signature searches for the text "hello world".

part="hello world"

Use \x<hex value> to add binary characters to the signature.

Example: In the following signature the white space is replaced with \x20.

part="hello\x20world"

To look for the backslash character, enter a double backslash.

Example: The following signature looks for the text "hello\world".

part="hello\\world"



Note: The length of the text searched for by **part** must be at least three characters. SecureSphere does not save signatures containing part lengths less than three characters.

Multi Part Signatures

You can include as many parts as required in a signature to search for a sequence of text strings.

Example: The following signature:

```
part="Imperva", part="Secure", part="Sphere"
```

can match either of the following strings:

- `abcdImpervaSecureSphere`
- `abcdImpervaabcdSecureSphere`
- `abcdImpervaabcdSecure---Sphere`

Different parts should be separated by a comma. White spaces are allowed before and after the comma.

Absolute Modifiers

Absolute modifiers limit the part to be matched to a specific area of a stream.

The absolute modifiers that are supported are:

- **amin**: The absolute position in the stream to start matching this part.
- **amax**: The absolute position in the stream to stop matching this part.

Example: The following signature searches for the string `cmd.exe` only from position 10 to position 20 in the stream.

```
part="cmd.exe", amin="10", amax="20"
```

You can include absolute modifiers after any part by using commas. You can either add a single modifier or both modifiers for each part.

Relative Modifiers

Relative modifiers limit the part to a specific area after the previous part.

The supported relative modifiers are:

- **rmin**: The position, relative to the previous part to start searching for this part.
- **rmax**: The position, relative to the previous part to stop searching for this part.

Example: The following signature includes two parts. The second part is searched for in a range of 10 characters from the first part.

```
part="cmd", part=".exe", rmax="10"
```

The string `cmd12345.exe` matches this signature. However the string `cmd12345678.exe` does not match the signature because the "`.exe`" part ends 12 characters after the "`cmd`" part.

You can include relative modifiers after any part by using commas. You can either add a single modifier or both modifiers for each part. You can also include both relative and absolute modifiers for the same part.

Relative modifiers are ignored for the first part, as the part must be relative to the previous part, which in this case does not exist.

Regular Expressions

The following sections describe how regular expressions, including:

Regular Expressions in SecureSphere 475
Using Regular Expressions in Signatures 477

Regular Expressions in SecureSphere

Regular expressions are used in SecureSphere for the following purposes:

	See also
Signatures	Using Regular Expressions in Signatures on page 477
Sensitive data to match credit card or Social Security Numbers	Configuring Sensitive Data Protection on page 350
U2P plugin	hpl_u2p and hpl_u2pp - URL to Parameters on page 487
Parameter type	Editing a URL Parameter Value Type on page 260
Host names and URL prefixes	Reverse Proxies on page 459

Regular Expression Syntax in Brief

SecureSphere supports a simplified form of standard regular expressions.

A regular expression is a text string defining the criteria for matching other text strings.

- Characters other than
  `. $ ^ { [ ( | ) ] } * + ? \`
  match themselves.
- A dot (`.`) matches almost any character, so `.gr.y` matches `grey` or `gray` or `gr#y` or `gr5y`
- `?` indicates that the preceding char is optional, so `</?script>` matches `<script>` or `</script>`

- + is a multiplier for 1 or more of what is before, so `[0-9]+` matches `1` or `123` or `353435`
- * is a multiplier for 0 or many of what is before, so `[0-9]*abc` matches `123abc` and `abc`



Note:

- When it is necessary to specify the minimum and maximum number of repeats explicitly, the bounds operator "`{}`" may be used.
- When using a regular expression, it is essential to remember that special characters must be escaped. For example, the characters ".", "*", "|" and others should be written as "\.", "*", "\|" *etc*. These are the standard rules for writing regular expressions and are not to specific to plugins or to SecureSphere.

Brackets

[ ]

- `[ ]` defines a list or a range of characters, for example:
- `[abc]` matches `a` or `b` or `c`
- `[a-c]` matches `a` or `b` or `c` (the - indicates a range)
- `[0-9]` matches `0` or `1` or `2` or `3`
- `[a-zA-Z]` matches `a` or `b` or `z` or `A` or `B` or `Z`
- `[a-zA-Z0-9]` matches any alphanumerical character
- `[a-cx-z]` matches `a` or `b` or `c` or `x` or `y` or `z` but not `d`
- `[Ss]ecure[Ss]here` matches all of the following:
- Securesphere
- SecureSphere
- secureSphere
- securesphere

Parentheses

( )

Parentheses serve two purposes:

- To group items together into a sub-expression, for example, `(ab)*` matches `abab`ab
- To group blocks of characters or words you want to retain for later use, typically for text replacement, for example, for reordering a list, as follows:

If the regular expression describing the original text is:

```
first second (fourth) (third)
```

Each group of ( ) is assigned a sequential number, as follows:

- `(fourth)` is the first group, denoted by $1
- `(third)` is the second group, denoted by $2

The replacement string might be:

```
first second $2 $1
```

This reverses the sequence of the third and fourth sub-strings in the text, which are denoted by $1 and $2 respectively.

Character classes

- '\s" matches white space (tabs, spaces, etc).
- '\d" matches decimal digits (0-9)
- '." matches any character
- '\w" matches alphanumeric characters and the underscore symbol ('_").
- '\W" matches non-alphanumeric characters.

Alternatives

Alternatives occur when the expression can match either one subexpression or another. The alternatives are separated by "|", or "\|".

The scope of an alternative is the largest possible previous sub-expression.



Note: This behavior is exactly the opposite of that of repeat operators, whose scope is the smallest previous sub-expression.

Escaping

To specify a special character, prefix the escape character (\).

Binary characters (\x) are also supported, as they are in the basic signature.

Anchors

An anchor matches the null string at the start or end of a line.

"^" matches the null string at the start of a line

"$" matches the null string at the end of a line

Using Regular Expressions in Signatures

The use of a regular expression part is optional and only applies to non-stream signatures (that is URL, parameters, parameter types, and headers). You can only include one regular expression part in a signature. The regular expression part must be the last part in the signature. Any number of non-regular expression parts can precede the regular expression part. SecureSphere searches for the regular expression only if all the preceding parts are found.



Note: The regular expression is searched for on the entire object (for example URL) and not on the text following the last part.

For example, the following signature searches the text string "cmd" and then searches the following ".exe" text string. If both text strings are found, the signature searches the regular expression "cmd\s*\.\s*exe" in the entire URL.

This regular expression specifies that between the text string "cmd" and ".exe" only white space appears.

```
part="cmd", part=".exe", rmax="10" rgxp="cmd\s*\.\s*exe"
```

APPENDIX C

Plugins

This Appendix describes SecureSphere plugins.

Plugins Overview on page 478
Configuring Plugins on page 479s
Custom Plugins Reference on page 487

Plugins Overview 478
Order of Plugin Execution 479
Configuring Plugins 479
Custom Plugins Reference 487

Plugins Overview

SecureSphere simplifies profiles by pre-processing application traffic and modifying the data presented to the profiling mechanism. The application sees the original unmodified traffic, so there is no need to modify the application, but the profile is greatly simplified and reduced in size.

For example, if user names are embedded in URLs, a plugin can extract the user name so that the profiling mechanism sees the user name as a URL parameter. Instead of tracking an ever-growing list of URLs of the form app/john, app/paul, app/george, app/ringo and so on, the profile tracks only one URL.



The plugin does not modify the application traffic. Only SecureSphere's profiling mechanism sees the plugin's output, while the application sees the original input traffic.

The plugin configuration is checked on the gateway. If there is an error, this information is displayed on the gateway screen for the gateways protecting the service on which the plugins are configured.

Order of Plugin Execution

Plugins are executed in sequence, according to their defined **Order**. This is the case for custom plugins as well.

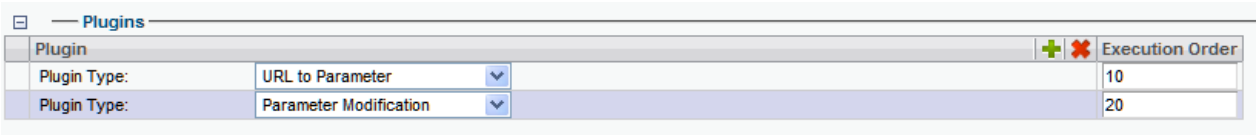


Figure 75: Plugin Order

Note: When defining plugin execution order, it is recommended that you enter numbers like 10, 20, 30 etc., so that you can later insert new plugins between existing plugins, as may be required.

Configuring Plugins

Plugins are gateway add-ons that do not require a version upgrade or any updates on the MX side. The plugins specified in the previous section come built-in with SecureSphere Gateway installation.

Configuring a plugin means configuring a plugin instance, that is, associating a plugin with a specific service, and then defining the plugin's parameters. It is possible to configure multiple instances of the same plugin, where each instance has different parameters and is associated with a different server group, and to control the order of plugin execution.

This section contains the following:

- Custom Plugins on page 479
- Associating a Plugin with a Service and Configuring the Plugin on page 480

Custom Plugins

Custom plugins are plugins which were defined in SecureSphere versions prior to version 9.5, for example, hpl_p2u or hpl_p2p. If you have defined such plugins in the past, there is nothing you need to do in order to continue using them in SecureSphere version 9.5 and higher. The plugins will continue to function in the same way they functioned in previous versions.

Beginning with SecureSphere version 9.5, web plugins are defined somewhat differently. There have been no changes to the database plugin hpl_db_source.

Plugin Correspondence (starting with SecureSphere version 9.5)

Custom Plugin ...	... corresponds to Plugin of Type
hpl_jsp	This plugin has been replaced by **Learn JSessionID parameters as regular parameters** in **Main** > **Sites** > **Service** > **Operations** tab > **Advanced HTTP Settings** section.
hpl_u2p	**URL to Parameter**
hpl_u2pp	This plugin is no longer available, as it is no longer necessary. The order of plugin execution is now defined explicitly (see below).
hpl_p2u	**Parameter to URL**
hpl_p2p	**Parameter Modification**
hpl_web_separator	**Special Parameter Delimiter**
hpl_db_source	There have been no changes to this database plugin.

Regular Expressions

When using a regular expression, it is essential to remember that special characters must be escaped. For example, the characters ".", "*", "|" and others should be written as "\.", "*", "\|" *etc*. These are the standard rules for writing regular expressions and are not to specific to plugins or to SecureSphere.

Associating a Plugin with a Service and Configuring the Plugin

To configure a plugin, you must associate it with a service and define its parameters.

To configure a plugin:

1. Select Main > Setup > Sites.
2. Select a server group.
3. Select a service.
4. In the **Definitions** tab, under **Plugins**, click .
5. Select the plugin's type from the dropdown menu, in the following figure .

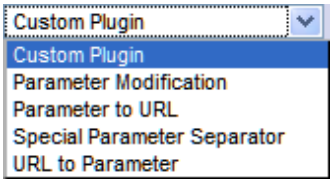


Figure 76: Plugin Types

Custom plugins are plugins which were defined in SecureSphere versions prior to version 9.5. You can continue to define custom plugins in SecureSphere version 9.5 and higher, but it is recommended that you use the new method described here.

6. In **Order**, enter the order in which the plugin is to be executed.

   Plugins are executed in ascending sequence of their **Order** numbers, for example, 10 followed by 20 followed by 30, *etc*.

Take into consideration the other plugins defined for the service. It is recommended that you enter numbers like 10, 20, 30 etc., so that you can later insert new plugins between existing plugins, as may be required. In any case, you can always change **Order** later as required.

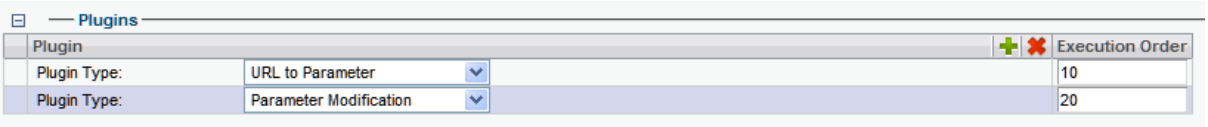


Figure 77: Plugin Order

7. Click **Save**. This associates the plugin with a specific service. Next, you must define the plugin's parameters.
8. Expand the plugin by clicking the **+** sign to the left of its name.
9. Under **Plugin Configuration**, enter the plugin's parameters.
10. Click **Save**. Settings are saved. If you are in delayed activation mode, you need to activate these settings. For further information, see Activating Settings on page 75.

Plugin Correspondence

Plugin Type	For a description of the plugin's parameters, see ...
URL to Parameter	URL to Parameter Plugin on page 481
Parameter to URL	Parameter to URL Plugin on page 483
Parameter Modification	Parameter Modification Plugin on page 485
Special Parameter Delimiter	Special Parameter Delimiter Plugin on page 486
Custom Plugin	Parameter to URL Plugin on page 483

URL to Parameter Plugin

A URL to Parameter plugin addresses the issue of URLs which contain an embedded username or hash value, that is, URLs of the form http://a/b/<RANDOM>/d/e, where <RANDOM> is of no significance for the profile. The plugin removes <RANDOM> from the URL and saves <RANDOM> as a URL parameter.

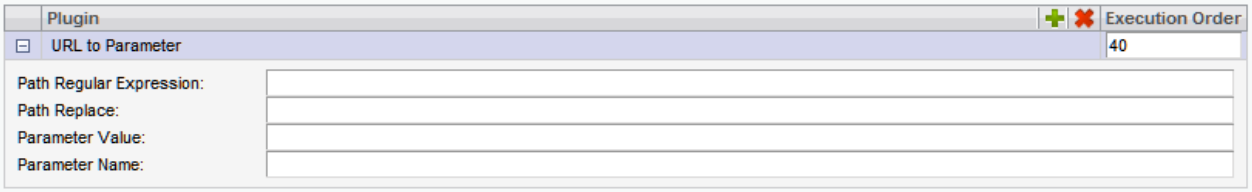


Figure 78: URL to Parameter Plugin

The URL to Parameter plugin parameters are listed in the following table.

URL to Parameter Plugin Parameters

Plugin Parameter	Description
Path Regular Expression	A regular expression to be run on the URL in order to map its dynamic parts. This mapping then becomes the basis for the part replacement. **Note:** Path Regular Expression parameters should delimited by commas and must not contain new lines. See the examples below.
Path Replace	The structure of the new URL.
Parameter Value	The value of the newly-added parameter. This is usually the dynamic element.
Parameter Name	The name of the newly-added parameter(s).

URL to Parameter Plugin Example - I

URL to Parameter Plugin Example

Plugin Parameter	Description
Path Regular Expression	(/iwebreports/.*)(a.*f/)(.*) **Note:** The regular expression should be written on a single line.
Path Replace	$1$3
Parameter Replace	$2
Parameter Name	sid

The regular expression defined in Path Regular Expression breaks the URL into three arguments: $1, $2 and $3. $2 is deleted from the URL and its value is specified as the value of the URL parameter named "sid".

The result is that SecureSphere learns the URL "/iwebreports/abc/def/ghi.asp"
as: "/iwebreports/ghi.asp" with the parameter "sid".

URL to Parameter Plugin Example - II

URL to Parameter Plugin Example

Plugin Parameter	Description
Path Regular Expression	(/\(.+\))(/.+) **Note:** The regular expression should be written on a single line.
Path Replace	$2
Parameter Replace	$1
Parameter Name	sid

The regular expression defined in Path Regular Expression breaks the URL into two arguments: $1 and $2. $1 is deleted from the URL and its value is specified as the value of the URL parameter named "sid".

So URLs of the form: "/(1209029802)/index.asp", "/(1209029803)/index.asp", and so on are learned as: "/index.asp"

Parameter to URL Plugin

A Parameter to URL plugin Extracts parameters and creates multiple URL (one for each function) based on the parameter values. It is used when a single URL serves multiple functions, with different parameters for each function.

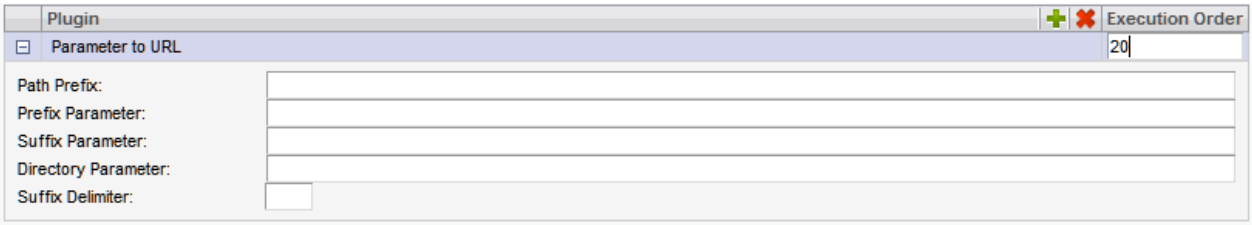


Figure 79: Parameter to URL Plugin

The Parameter to URL plugin parameters are listed in the following table.

Parameter to URL Plugin Parameters

Plugin Parameter	Description
Path Prefix	The plugin is activated on URLs the prefix of which matches this parameter. You can specify multiple values in a comma-separated list, where each value is enclosed in double quotes, for example, `"/a","/b","/c"`.
Prefix Parameter	When a URL parameter matches the value of the parameter, the value of this URL parameter is added as the prefix of the URL. You can specify multiple values in a comma-separated list, where each value is enclosed in double quotes, for example, `"a","b","c"`.
Suffix Parameter	When a URL parameter matches the value of the prefix parameter, the value of this URL parameter is added as the suffix of the URL. You can specify multiple values in a comma-separated list, where each value is enclosed in double quotes, for example, `"a","b","c"`.
Directory Parameter	When a URL parameter matches the value of the prefix parameter, the value of this URL parameter is added as the last directory of the URL. You can specify multiple values in a comma-separated list, where each value is enclosed in double quotes, for example, `"a","b","c"`.
Suffix Delimiter	The value of this parameter is used as a delimiter between the URL and the added suffix-parameter. The default value of this parameter is **&** and it is applicable when using the suffix parameter.

Parameter to URL Plugin Example - I

To have SecureSphere learn the URL
`/dispatcher?menuid=control&nextpageid=update&user=Jon`
as
`/control/update/dispatcher?user=Jon`
configure the plugin as follows:

Parameter to URL Plugin Example

Plugin Parameter	Description
Path Prefix	/dispatcher
Prefix Parameter	menuid
Suffix Parameter	
Directory Parameter	nextpageid
Suffix Delimiter	

Parameter to URL Plugin Example - II

To have SecureSphere learn the URL
`/dispatcher/home.asp?a=param`
as
`/dispatcher/param/home.asp`
configure the plugin as follows:

Parameter to URL Plugin Example

Plugin Parameter	Description
Path Prefix	/dispatcher
Prefix Parameter	
Suffix Parameter	
Directory Parameter	a
Suffix Delimiter	

Parameter to URL Plugin Example - III

To have SecureSphere learn the URL
`/g1/len1.asp?abc=param1&def=param2&gh=param3&ij=param7`
as:
`/param1/g1/param3/len1.asp.param2` with the parameters: `ij`
configure the plugin as follows:

Parameter to URL Plugin Example

Plugin Parameter	Description
Path Prefix	/g1
Prefix Parameter	abc
Suffix Parameter	def
Directory Parameter	gh
Suffix Delimiter	.

Parameter Modification Plugin

A Parameter Modification plugin removes unwanted text from a parameter in order to simplify profile tuning (by combining dynamic parameters into one parameter) and to reduce unnecessary growth in a profile's size. For example, the following URLs would be considered as different from the profiling point of view, even though they effectively point to the same page.

- ".../articles.aspx?FolderID123=456&docID=101813"
- ".../articles.aspx?FolderID321=789&docID=101813"

The Parameter Modification plugin ignores the specified part of the parameter and so "combines" many parameters in the profile so that they are treated as though they were identical.

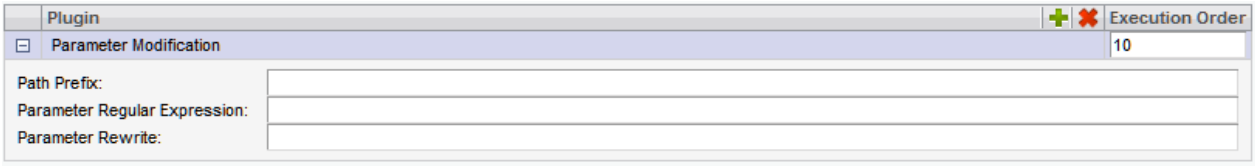


Figure 80: Parameter Modification Plugin

The Parameter Modification plugin parameters are listed in the following table.

Parameter Modification Plugin Parameters

Plugin Parameter	Description
Path Prefix	The plugin is activated on URLs the prefix of which matches this parameter. You can specify multiple values in a comma-separated list, where each value is enclosed in double quotes, for example, `"/a","/b","/c"`.
Parameter Regular Expression	A regular expression describing the parameter, including that part of the parameter to be ignored.
Parameter Rewrite	When a parameter matches the value of Path Prefix and Parameter Regular Expression, the parameter is rewritten as defined by Parameter Rewrite.

Parameter Modification Plugin Example

Parameter Modification Plugin Example

Plugin Parameter	Description
Path Prefix	/articles.aspx
Parameter Regular Expression	(.)older(.)(.+)
Parameter Rewrite	$1_test_$2_$3

SecureSphere learns the URL: /articles.aspx?FolderID=378&docID=87771&lang=EN
as having the parameter "F_test_I_D" instead of "FolderID". This is internal to SecureSphere: the Web server sees the original URL.

Special Parameter Delimiter Plugin

Some applications use a delimiter other than the standard ampersand "&" between parameters, so that multiple parameters appear to SecureSphere as a single parameter. A Special Parameter Delimiter plugin, by replacing the application-specific delimiter with ampersand, enables SecureSphere to correctly parse the parameters. This is internal to SecureSphere: the Web server sees the original URL.



Figure 81: Special Parameter Delimiter Plugin

The Special Parameter Delimiter plugin parameters are listed in the following table.

Special Parameter Delimiter Plugin Parameters

Plugin Parameter	Description
Special Parameter Delimiter	The delimiter used by the application - a replacement character for "&".

Special Parameter Delimiter Plugin Example:

To have SecureSphere learn the URL:

`/a/b/x.asp?param1=tom+param2=jerry`

as `/a/b/x.asp` with the parameters `param1` and `param2`

configure the plugin as follows:

Special Parameter Delimiter Plugin Example

Plugin Parameter	Description
Special Parameter Delimiter	+

Custom Plugin

To configure a custom plugin, enter the following parameters:

Custom Plugin Parameters

Plugin Parameter	Description
Name	Enter one of the custom plugin names. For more information, see Custom Plugins Reference on page 487.
Plugin Configuration	Enter the appropriate parameters for the custom plugin, as described in Custom Plugins Reference on page 487.



Figure 82: Custom Plugin Configuration

Custom Plugins Reference

This section contains the following information:

Custom Web Plugins on page 487

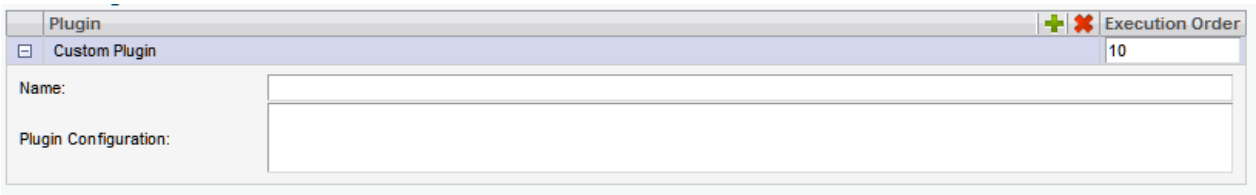


Figure 83: Custom Plugin Configuration

Custom Web Plugins

This section describes custom plugins for Web traffic:

- hpl_u2p and hpl_u2pp - URL to Parameters on page 487
- hpl_p2u on page 490
- hpl_p2p on page 491
- hpl_web_separator on page 492

hpl_u2p and hpl_u2pp - URL to Parameters

These plugins are used when a URL has a random element, which makes the profile grow rapidly in size and complexity to no benefit. The plugin extracts the random element from the URL and inserts it as a URL parameter.

hpl_u2p plugin

Item	Description
Plugin Name	hpl_u2p
Addresses Problem	URLs in HTTP traffic are of the form `http://a/b/<RANDOM>/d/e`, where `<RANDOM>` is of no significance for the profile.
Typical Case	A URL contains an embedded username or hash value.
Problem's Symptom	The profile grows rapidly in size and complexity to no benefit.
Action	Remove `<RANDOM>` from the URL and save `<RANDOM>` as a URL parameter. **Note:** It is recommended that you consult Imperva support for guidance before configuring this plugin, as this can be a delicate task.
Parameters	**path-regexp**: A regular expression to be run on the URL in order to map its dynamic parts. This mapping then becomes the basis for the part replacement. **Note: path-regexp** parameters should delimited by commas and should not contain new lines. See the examples below. **path-replace**: This parameter specifies the structure of the new URL. **parameter-replace**: The value of the newly-added parameter. This is usually the dynamic element. **parameter-name**: The name of the newly-added parameter(s).
Examples	**EXAMPLE 1:** path-regexp="(/iwebreports/.*)(a.*f/)(.*)",path-replace="$1$3",parameter-replace="$2",parameter-name="sid" **Note:** The regular expression should be written on a single line. The regular expression defined in path-regexp breaks the URL into three arguments: $1, $2 and $3. $2 is deleted from the URL and its value is specified as the value of the URL parameter named "sid". The result is that SecureSphere learns the URL "/iwebreports/abc/def/ghi.asp" as: "/iwebreports/ghi.asp" with the parameter "sid". **EXAMPLE 2:** path-regexp="(/\(.+\))(/.+)",path-replace="$2",parameter-replace="$1",parameter-name="sid" **Note:** The regular expression should be written on a single line. The regular expression defined in path-regexp breaks the URL into two arguments: $1 and $2. $1 is deleted from the URL and its value is specified as the value of the URL parameter named "sid". So URLs of the form: "/(1209029802)/index.asp", "/(1209029803)/index.asp", and so on are learned as: "/index.asp"

hpl_u2pp



Note: The hpl_u2pp is described here for references purposes only. It is no longer possible to define this plugin, nor is it necessary, because plugin execution order is now determined explicitly by the **Order** parameter (see Order of Plugin Execution on page 479).

hpl_u2pp is identical in its configuration and operation to hpl_u2p, with one difference: the point in the traffic processing path in which they run:

- hpl_u2p runs before URL security policies and rules (for example, signatures and static file exclusion) are executed.
- hpl_u2pp runs after they are executed.

hpl_u2pp can be chained to run after hpl_p2u. This capability is useful when for processing URI parameter with structured values, where one part must be collected and profiled as part of the URI, and another part must be profiled as a parameter value proper. In this case, the hpl_p2u plugin can make the entire parameter value part of the URI, and hpl_u2pp can extract from that URI section that part of the original value that is to be profiled as parameter value.

For example, if the original URI is of the form `/folder/something.asp?p=subapp/username`,

SecureSphere can process the URI as `/folder/subapp/something.asp?user=username`

To do this:

1. Use hpl_p2u to transform the URI to `/folder/subapp/username/something.asp?`
2. Use hpl_u2pp to get to the required result of `/folder/subapp/something.asp?user=username`



Note: If both hpl_p2u and hpl_u2pp are applied to the same service, hpl_p2u is always executed before hpl_u2pp.

hpl_p2u

hpl_p2u plugin

Item	Description
Plugin Name	hpl_p2u
Addresses Problem	Multiple functions for a single URL, with different parameters for each function.
Typical Case	
Problem's Symptom	Many parameters for the same URL, all profiled as not required.
Action	Extracts parameters and creates multiple URL (one for each function) based on the parameter values
Parameters	**path-prefix**: The plugin is activated on URLs the prefix of which matches this parameter. You can specify multiple values in a comma-separated list enclosed in double quotes, for example, "/a,/b,/c". **prefix-parameter**: When a URL parameter matches the value of the parameter, the value of this URL parameter is added as the prefix of the URL. You can specify multiple values in a comma-separated list enclosed in double quotes, for example, "a,b,c". **suffix-parameter**: When a URL parameter matches the value of the prefix parameter, the value of this URL parameter is added as the suffix of the URL. You can specify multiple values in a comma-separated list enclosed in double quotes, for example, "a,b,c". **dir-parameter**: When a URL parameter matches the value of the prefix parameter, the value of this URL parameter is added as the last directory of the URL. You can specify multiple values in a comma-separated list enclosed in double quotes, for example, "a,b,c". **suffix-delimiter**: The value of this parameter is used as a delimiter between the URL and the added suffix-parameter. The default value of this parameter is **/** and it is applicable when using the suffix parameter.
Example	Plugin Configuration: path-prefix="/dispatcher", prefix-parameter="menuid", dir-parameter="nextpageid" SecureSphere learns the URL: /dispatcher?menuid=control&nextpageid=update&user=Jon as: control/dispatcher/update?user=Jon
Example	Plugin Configuration: path-prefix="/dispatcher", dir-parameter="a" SecureSphere learns the URL: /dispatcher/home.asp?a=param as: /dispatcher/param/home.asp
Example	Plugin Configuration: path-prefix="/g1", prefix-parameter="abc", suffix-parameter="def", dir-parameter="gh", suffix-delimiter="." SecureSphere learns the URL: /g1/len1.asp?abc=param1&def=param2&gh=param3&ij=param7 as: param1/g1/param3/len1.asp.param2?ij=param7

hpl_p2p

hpl_p2p plugin

Item	Description
Plugin Name	hpl_p2p
Addresses Problem	Remove unwanted text from a parameter in order to reduce unnecessary growth in a profile's size.
Typical Case	For example, the following URLs would be considered as different from the profiling point of view, even though they effectively point to the same page. The plugin ensures that they are treated as though they were identical. • "/articles.aspx?FolderID123=456&docID=101813" • "/articles.aspx?FolderID321=789&docID=101813"
Problem's Symptom	Profile is excessively large because it contains many parameters which are essentially duplicates.
Action	Ignores the specified part of the parameter and so "combines" many parameters in the profile.
Parameters	**path-prefix**: The plugin is activated on URLs the prefix of which matches this parameter. You can specify multiple values in a comma-separated list enclosed in double quotes, for example, "/a,/b,/c". **parameter-regexp:** A regular expression describing the parameter, including that part of the parameter to be ignored. **parameter-rewrite**: When a parameter matches the value of **path-prefix** and **parameter-regexp**, the parameter is rewritten as defined by **parameter-rewrite**.
Example	Plugin Configuration: **path-prefix**="/articles.aspx",**parameter-regexp**="(.)older(.)(.+)",**parameter-rewrite**="$1_test_$2_$3" **Note:** The parameters should all be written on a single line, separated by commas. SecureSphere learns the URL: /articles.aspx?FolderID=378&docID=87771&lang=EN as having the parameter "F_test_I_D" instead of FolderID

hpl_web_separator

hpl_web_separator plugin

Item	Description
Plugin Name	hpl_web_separator
Addresses Problem	Some applications use a separator other than the usual ampersand "&" between parameters. Without knowledge of the separator, the SecureSphere system will not parse the parameters correctly.
Typical Case	
Problem's Symptom	Multiple parameters appear as a single parameter.
Action	Replaces the user-defined separator with ampersand so that the parser will parse the parameters correctly.
Parameters	The new separator - a replacement character for "&".
Example	Plugin Configuration: Type in the new char to be used as a separator. For example: + SecureSphere learns the URL: /a/b/x.asp?param1=tom+param2=jerry as: /a/b/x.asp with the parameters param1 and param2

Custom Database Plugins

This section describes custom plugins for DB traffic.

hpl_db_source

This plugin enables the administrator to recombine data from the application in order to structure the profile differently.

hpl_db_source plugin

Parameter	Description
Plugin Name	hpl_db_source
Addresses Problem	The SecureSphere administrator wants the profile parameters to reflect combinations of data in the database traffic.
Typical Case	The SecureSphere administrator wants the user in the profile to be a combination of the user and application in the application traffic.
Problem's Symptom	The profiled traffic separates data fields which the profile must combine.
Action	Recombine source data in a form required by the profile.
Parameters	**DB_USER**: The database user as seen in the database traffic. **OS_USER**: The operating system user as seen in the database traffic. **APPLICATION**: The application as seen in the database traffic. **HOST**: The hostname as seen in the database traffic. **SOURCE_IP**: The source IP address of the application traffic. This parameter is read-only and cannot be replaced.
Example	Plugin Configuration: DB_USER="$(DB_USER)@$(APPLICATION)",APPLICATION="$(APPLICATION)-$(SOURCE_IP)" **Note:** The text above should be written on a single line. Plugin Effect: The application will be added to the DB user and the source IP address will be added to the application, so that the profiling mechanism sees the user name as a combination of user name and application (with **@** as a delimiter), and application as a combination of application and source IP address (with **-** as a delimiter) in the traffic.

hpl_dq2q

MSSQL and Sybase allow the use of the double-quote characters (") as delimiters. This plugin replaces the double quotes by single quotes in strings processed by SecureSphere. The application sees the original text of the queries.

This plugin requires no configuration.

APPENDIX D

Placeholders

This Appendix describes SecureSphere placeholders, including:

Working with Placeholders on page 494

Working with Placeholders 494

Working with Standard Placeholders 495

Working with Collection Placeholders 495

Working with Placeholders

Certain SecureSphere Action Sets enable you to utilize placeholders that call variables from the system instead of displaying static values. This enables you to integrate dynamic information into action interface messages which result in more valuable and actionable information to users.



Notes:

- Placeholders start with a dollar sign and enclosed by rounded brackets ${}.
- To hide a placeholder if no data is returned, add an exclamation point '!' immediately after the dollar sign to the placeholder. For example: $!{Event.struct.webCorrelationInfo.webItemType}.
- You may want to enclose each parameter with its string within identifiers such as a comma (,), colon (:) or any other character to later parse this information and extract specific items. This indicates to a script or a parser where value start and end.

SecureSphere offers a number of placeholders for use with certain action sets. There are two primary types of placeholders available in SecureSphere:

- **Standard Placeholders**: List an individual value regarding a single item. For more information see Working with Standard Placeholders on page 495

- **Collection Placeholders**: List a number of values for a collection of related items for a single event. For more information see .Working with Collection Placeholders on page 495



Note: SYSLOG messages can be sent from the Gateway to the SYSLOG server, depending on the configured followed action.

Working with Standard Placeholders

Standard placeholders enable you to query system data and generate messages with this data in Action Interfaces message fields. This can be valuable to provide specific information surrounding an event such as an alert. For a list of standard placeholders by action interface, see List of Standard Placeholders on page 498.

EXAMPLE: ACTION INTERFACE MESSAGE USING PLACEHOLDERS

Lets say you want to create an action interface that generates a syslog message and sends it to a syslog sever. You may additionally want to include details regarding the event. You would create a new Action Set and name it SendEventInformationtoSyslog. You could use a combination of free text and placeholders to create a message. Free text is displayed in the syslog message as is, while the placeholder is replaced with data regarding the item.

Example of a message with free text and placeholders:

```
Alert Number;${Alert.dn}; of type;${Alert.alertType}; took place
at;${Alert.createTime}; the alert description is;${Alert.description};
```

Resulting syslog message: Alert Number;1143; of type;Protocol; took place at;2009-01-14 11:38:06.0; the alert description is; Login Statement Error by scott from 10.1.1.94;

Example of message with only placeholders:

```
${Alert.dn};${Alert.alertType};${Alert.createTime};${Alert.description}
```

Resulting syslog message: 11146;Protocol;2009-01-14 11:41:01.0;Login Statement Error by scott from 10.1.1.94

Working with Collection Placeholders

Collection placeholders are placeholders that can list a number of values for a collection of related items for a single event. For example, you may want to list all the HTTP headers that were part of a single violation. SecureSphere provides a number of collection placeholders that enable you display such information.

Collection placeholders use lists. Lists can have multiple items per row. For example the HTTP request header collection lists all headers in an event's HTTP request. For each header three items are available, the header name, the header value and whether the header contains a violation.



Note: Collection placeholders are not available as part of followed actions.

SecureSphere offers two types of collection placeholders:

- **Single Collection Placeholders**: Used to list values related to a single item. For more information see Understanding Single Collection Placeholders on page 496.
- **Nested Collection Placeholder**: Used when a list item is a collection, for example used to list all details (nested list) of all violations (parent list) for an event. For more information see Understanding the Nested Collection Placeholder on page 496.



Note: For a list of available collection placeholders, see List of Collection Placeholders on page 521.

Understanding Single Collection Placeholders

(#list)

The syntax for using single collection placeholders differs depending on if you want to list a single item for this placeholder or multiple items.

- To list a single item: `#list(${list_name} "item_name 1")`
- To list multiple items: `#list(${list_name} "${item.item_name_1} ${item.item_name_2}")`

This following is a breakdown of the items in the list:

Description of Collection Placeholder Components

Component	Description
#list	Macro that accesses the items held by the collection placeholder "list_name".
"Item_name_1" and "Item_name_2"	Names of available items

EXAMPLE: DISPLAY ALL HTTP HEADERS IN A VIOLATION

SecureSphere has detected a violation in web traffic and you want to display the HTTP headers related to this violation. You'd use a collection placeholder with the following syntax:

To display the first header:

```
#list(${Event.struct.httpRequest.headers} "name")
```

This would result in the header names in the specific violation being displayed.

To display all headers in the related to the violation:

```
#list(${Event.struct.httpRequest.headers} "${item.name} ${item.value}")
```

This would result in a list of the HTTP header names with their values.

Understanding the Nested Collection Placeholder

(#innerlist)

Additionally, a special syntax is used when a list item itself is a collection. This placeholder can contain nested collections.

Syntax: #innerlist(${list_1_name} "${item1.item1_name_1}" "list_2_name" "${item2.item2_name_1} ${item2.item2_name_2}")

The following table lists items used in the syntax of the nested collection placeholder.

Description of Collection Placeholder Components

Component	Description
#innerlist	Macro that accesses the items ("item#_name_#") held by the collections "list_name_#" in the nested collection placeholder.
list_name_1	A collection containing a nested collection list_name_2, meaning that along with the other items in list_name_1 there is a collection item, list_name_2, for each list entry.
"item1_name_1"	An item in the list_name_1 collection.
"item2_name_1", "item2_name_2"	Items in the list_name_2 collection.

The items for the first collection, list_name_1, are available with item1 as a prefix, and the items in the second collection are available with item2 as a prefix. The items from both collections can be used in any order anywhere in the message created. For example:

#innerlist(${list_1_name} "list_2_name" "${item2.item2_name_1} ${item2.item2_name_2} are part of collection ${item1.item1_name_1}")

EXAMPLE: USING NESTED COLLECTIONS

Syntax: `#innerlist(${Event.violations} "${item1.metadata.displayName}:" "infoAttributes" "${item2.name}=${item2.value}")`

Sample Output

The following is an example of data sent to a syslog server when a violation placeholder is used:

Signature Violation:attack-id=1002, location=parameters, pattern-id=401002, signature-pattern=part="cmd.exe", policy-id=992347469600906248, dictionary-name=Worms and Critical Vulnerabilities for Web Applications, dictionary-id=-1091025908323764786, position=1, description=WEB-IIS cmd.exe access, matched-text=cmd.exe.



Note: You can also use nested collections to display event irregular behavior.
Syntax: #innerlist(${Event.irregularBehaviors} "${item1.metadata.displayName}:" "infoAttributes" "${item2.name}=${item2.value}")

List of Standard Placeholders

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Alert.dn}	The alert number	v	v	v	v	v									
${Alert.createTime}	The alert creation date	v	v	v	v	v									
${Alert.lastUpdateTime}	The alert update time	v	v	v	v	v									
${Alert.description}	The alert description	v	v	v	v	v									
${Alert.username}	The app user name	v	v	v	v	v									
${Alert.alertType}	The alert type (Firewall, Signature, Protocol)	v	v	v	v	v									
${Alert.sourceGeoLocation}	The GeoLocation from where traffic that cause the alert originated	v			v	v									
${Alert.sourceIp}	The source IP of the client	v	v	v	v	v									
${Alert.simulationMode}	Whether the alert was generated in simulation mode	v	v	v	v	v									
${Alert.serverGroupName}	The name of the attacked server group	v	v	v	v	v									
${Alert.severity}	The severity of the alert	v	v	v	v	v									
${Alert.aggregationInfo.occurrences}	How many violations are aggregated in the alert	v	v	v	v	v									
${Alert.immediateAction}	The immediate action taken	v	v	v	v	v									
${Alert.serviceName}	The name of the attacked service	v	v	v	v	v									
${Alert.applicationName}	The name of the attacked application	v	v	v	v	v									
${Alert.alertMetadata.alertName}	The name of the alert	v	v	v	v	v									

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.dn}	The event ID	v	v	v	v	v	v	v							
${Event.createTime}	The event creation time	v	v	v	v	v	v	v							
${Event.gateway}	Gateway on which the event was detected	v	v	v	v	v									
${Event.lastUpdateTime}	The event update time	v	v	v	v	v		v							
${Event.eventType}	The event type (HTTP or network)	v	v	v	v	v		v							
${Event.sourceInfo.eventSourceType}	Source of the Event.		v	v	v		v								
${Event.sourceInfo.sourceIp}	The source IP of the client	v	v	v	v	v	v								
${Event.sourceInfo.sourcePort}	The source port of the attacker	v	v	v	v	v	v								
${Event.sourceInfo.geoIpLocation}	The source location from where the attack that generated the event originated.	v	v	v	v	v									
${Event.sourceInfo.ipProtocol}	The protocol of the event (IP,TCP, UDP,ICMP)	v	v	v	v	v									
${Event.destInfo.serverIp}	The destination IP address of the server	v	v	v	v	v	v								
${Event.destInfo.serverPort}	The destination port of the server	v	v	v	v	v	v								
${Event.arrivalTime}	The event arrival time	v	v	v	v	v									
${Event.ApplicationDefaultURL}							v								
${Event.serviceName}	The name of the attacked service	v	v	v	v	v	v								
${Event.applicationName}	The name of the attacked application	v	v	v	v	v	v								

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.struct.query.parsedQuery}	The parsed query that generated the alert		v			v	v								
${Event.struct.query.affectedRows}	The number of affected rows in the database		v			v	v								
${Event.struct.userGroup}	Name of the user group.		v				v								
${Event.struct.applicationUser}	The application username.		v				v								
${Event.struct.serviceType}	Service type of the audited item.		v				v								
${Event.struct.eventType}	Event type being audited.		v				v								
${Event.struct.host.host}	The source host name		v			v	v								
${Event.struct.host.hostAvailability}	Is the source host name available? (True\False)		v			v									
${Event.struct.user.user}	Extracts the database or web application username.	v	v			v	v								
${Event.struct.user.userType}	The type of user (Valid, OpenMode, NotAvailable)		v			v	v								
${Event.struct.user.authenticated}	Is the user authenticated? (True\False)		v			v	v								
${Event.struct.application.application}	The name of the source application		v			v	v								
${Event.struct.application.applicationAvailability}	Is the source application name available? (True\False)		v			v									
${Event.struct.complete.responseSize}	The size of the database response		v			v	v								
${Event.struct.complete.responseTime}	The time the database response took		v			v	v								

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.struct.complete.errorMessage}	The error message returned by the database		v			v									
${Event.struct.complete.complete}	Is the database response available? (True\False)		v			v									
${Event.struct.complete.completeSuccessful}	Has an exception occurred? (True\False)		v			v	v								
${Event.struct.complete.errorValue}	The exception string message		v			v	v								
${Event.struct.queryGroup.queryGroup}	The query group that the query belongs to		v			v	v								
${Event.struct.queryGroup.queryGroupType}	The type of the query group (Regular, Privileged,GeneralOperation)		v			v									
${Event.struct.queryGroup.queryGroupStatus}	The status of the query group (InLearning,InStaticProtection, InDynamicProtection)		v			v									
${Event.struct.osUser.osUser}	The source OS username		v			v	v								
${Event.struct.osUser.osUserAvailability}	Is the source OS username available? (True\False)		v			v									
${Event.struct.rawUser.rawUser}	The database username's raw string		v			v	v								
${Event.struct.rawUser.rawUserAvailability}	Is the database username's raw string available? (True\False)		v			v									
${Event.struct.rawData.encoding}	The encoding of the raw data		v			v									
${Event.struct.rawData.rawData}	The raw transaction buffer		v				v								

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.struct.webCorrelationInfo.sessionId}	The correlated web session ID		v			v	v								
{Event.struct.webCorrelationInfo.webEventId}	The correlated web event number		v			v	v								
${Event.struct.webCorrelationInfo.webClientIP}	The correlated web source IP address		v			v	v								
${Event.struct.user.username}	The user (login) that conducted the operation on a CIFS server.			v	v	v	v								
${Event.struct.user.domain}	The domain of the user that conducted the operation			v	v	v	v								
${Event.struct.user.fullName}	The full name of the user that conducted the operation			v	v	v	v								
${Event.struct.user.department}	The department of the user that conducted the operation			v	v	v	v								
${Event.struct.originalFile.folder}	The name of the folder before operation was conducted			v	v	v	v								
${Event.struct.originalFile.name}	The name of the file before the operation was conducted			v	v	v	v								
${Event.struct.originalFile.extension}	The extension of the file before the operation was conducted			v	v	v	v								
${Event.struct.targetFile.destFolder}	The name of the folder after the operation was conducted			v	v	v	v								
${Event.struct.targetFile.destName}	The name of the file after the operation was conducted			v	v	v	v								
${Event.struct.targetFile.destExtension}	The extension of the file after the operation was conducted			v	v	v	v								

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.struct.operation.displayName}	The type of operation conducted on the file or folder. For example, read or modify.			v	v	v	v								
${Event.struct.isRead}	Was the operation on the file or folder read? (True/False)			v	v	v	v								
${Event.struct.isWrite}	Was the operation on the file or folder write? (True/False)			v	v	v	v								
${Event.struct.originalOwner.name}	The original owner of the file or folder before the operation was conducted			v	v	v	v								
${Event.struct.originalOwner.department}	The department of the original owner before the operation was conducted			v	v	v	v								
${Event.struct.newOwner.name}	The name of the new owner of the file or folder			v	v	v	v								
${Event.struct.newOwner.department}	The department of the new owner of the file or folder			v	v	v	v								
${Event.struct.webCorrelationInfo.webItemType}	The type of the correlated event URL (URL\SOAP)		v			v									
${Event.struct.webCorrelationInfo.soapAction}	The SOAP action of the correlated event		v			v									
${Event.struct.webCorrelationInfo.webUserName}	The correlated web application username		v			v									
${Event.struct.webCorrelationInfo.additionalWebUserName}	Additional web users if exist		v			v									
${Event.struct.webCorrelationInfo.webApplicationId}	The correlated web application internal ID		v			v									
${Event.struct.webCorrelationInfo.urlPath}	The URL that generated the query		v			v	v								

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.struct.webCorrelationInfo.webUserAuthenticated}	Is the web user authenticated? (True/False)		v			v									
${Event.struct.session.id}	The web session ID	v				v									
${Event.struct.session.creationTime}	The time the web session was initiated	v				v									
${Event.struct.session.isVerified}	Is the web session verified? (True\False)	v				v									
${Event.struct.user.available}	Is the web username available? (True\False)	v				v									
${Event.struct.user.additionalUser}	The additional web user name	v				v									
${Event.struct.responseSize}	The size of the HTTP response	v				v									
${Event.struct.responseTime}	The time the HTTP response took	v				v									
${Event.struct.httpRequest.url.method}	The HTTP method	v				v									
${Event.struct.httpRequest.url.path}	The URL path	v				v									
${Event.struct.httpRequest.url.host}	The HTTP host	v				v									
${Event.struct.httpRequest.url.queryString}	The complete query string of the event	v				v									
${Event.struct.httpRequest.url.fullPath}	The URL full path	v				v									
${Event.struct.httpRequest.url.urlCRC}	The CRC of the URL	v				v									
${Event.struct.httpRequest.soapAction.soapAction}	The SOAP action of the event	v				v									

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.struct.httpRequest.soapAction.profileItemId}	The identifier of the violated SOAP item	v				v									
${Event.struct.httpRequest.soapAction.soapCRC}	The CRC of the SOAP item	v				v									
${Event.struct.httpRequest.soapAction.soapProfileItemType}	The SOAP profile item type (URL\SOAP action)	v				v									
${Event.struct.httpResponse.responseCode}	The HTTP response code	v				v									
${Event.struct.complete}	Has the transaction been completed? (True\False)	v				v									
${Event.struct.soap}	Is this a SOAP event? (True\False)	v				v									
${Event.struct.rawData}	The raw query used in an event	v	v	v	v	v									
${Event.struct.networkDirection}	The direction of the packet	v	v	v	v	v									
${Event.irregularBehaviors.complexSetMainTag}	The allowed methods and parameters	v	v	v	v	v									
${Event.irregularBehaviors.alert.dn}	The alert number	v	v	v	v	v									
${Event.irregularBehaviors.alert.createTime}	The alert creation date	v	v	v	v	v									
${Event.irregularBehaviors.alert.lastUpdateTime}	The alert update time	v	v	v	v	v									
${Event.irregularBehaviors.alert.username}	The app user name	v	v	v	v	v									
${Event.irregularBehaviors.alert.alertType}	The alert type (Firewall, Signature, Protocol)	v	v	v	v	v									
${Event.irregularBehaviors.alert.sourceIp}	The source IP of the client	v	v	v	v	v									

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.irregularBehaviors.alert.simulationMode}	Was the alert generated in simulation mode? (True/False)	v	v	v	v	v									
${Event.irregularBehaviors.alert.serverGroupName}	The name of the attacked server group	v	v	v	v	v									
${Event.irregularBehaviors.alert.severity}	The severity of the alert	v	v	v	v	v									
${Event.irregularBehaviors.alert.aggregationInfo.occurrences}	How many violations are aggregated in the alert	v	v	v	v	v									
${Event.irregularBehaviors.alert.immediateAction}	The immediate action taken	v	v	v	v	v									
${Event.irregularBehaviors.alert.serviceName}	The name of the attacked service	v	v	v	v	v									
${Event.irregularBehaviors.alert.applicationName}	The name of the attacked application	v	v	v	v	v									
${Event.violations.complexSetMainTag}	The allowed methods and parameters	v	v	v	v	v									
${Event.violations.alert.severity}	The severity of the violation	v	v	v	v	v									
${Event.violations.alert.aggregationInfo.occurrences}	Number of violations aggregated by the alert	v	v	v	v	v									
${Event.violations.alert.immediateAction}	The immediate action taken	v	v	v	v	v									
${Event.violations.alert.serviceName}	The name of the attacked service	v	v	v	v	v									
${Event.violations.alert.applicationName}	The name of the attacked application	v	v	v	v	v									
${Alert.actions.actionSetName}	The name of the followed action taken as a result of the alert	v	v	v	v	v									

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Alert.actions.actionTime}	The time of the followed action taken as a result of the alert	v	v	v	v	v									
${Alert.actions.description}	The description of the followed action taken as a result of the alert	v	v	v	v	v									
${Alert.actions.duration}	The duration of the followed action taken as a result of the alert	v	v	v	v	v									
${Alert.flag}	The alert flag (Important/Acknowledged/Dismissed/None)	v	v	v	v	v									
${Event.customTags.name}		v	v	v	v	v									
${Event.customTags.value}		v	v	v	v	v									
${Event.irregularBehaviors.action}		v	v	v	v	v									
${Event.irregularBehaviors.alert.alertMetadata.alertName}	The name of the alert	v	v	v	v	v									
${Event.irregularBehaviors.alert.description}	The description of the alert	v	v	v	v	v									
${Event.irregularBehaviors.alert.flag}	The alert flag (Important/Acknowledged/Dismissed/None)	v	v	v	v	v									
${Event.irregularBehaviors.description}	Description of the irregular behavior	v	v	v	v	v									
${Event.irregularBehaviors.infoAttributes.name}		v	v	v	v	v									
${Event.irregularBehaviors.infoAttributes.value}		v	v	v	v	v									
${Event.irregularBehaviors.metadata.displayName}		v	v	v	v	v									

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.irregularBehaviors.violatedItem}	The violated item	v	v	v	v	v									
${Event.originGWDn}	The ID of the gateway that generated the event	v	v	v	v	v									
${Rule.parent.displayName}	Policy display name	v	v	v	v	v									
${Rule.dn}	Represents a numerical value which is used for rules/alerts in SecureSphere, providing a unique numerical ID per policy	v	v	v	v	v									
${Event.irregularBehaviors.profileItemId}	The ID of the profile item	v	v			v									
${Event.irregularBehaviors.profileItemType}	The type of the profile item	v	v			v									
${Event.struct.tableGroups.tableGroupDn}	The table group ID		v			v									
${Event.struct.databases.schemaName}	The schema name		v			v	v								
${Event.struct.operations.storedProcedure}	Is the operation is a stored procedure? (True/False)		v			v	v								
${Event.struct.operations.operationId}	The operation ID		v			v									
${Event.struct.operations.name}	SQL Operation		v			v	v								
${Event.struct.operations.operationType}	Operation Type		v				v								
${Event.struct.operations.objects.name}	Object name		v			v	v								
${Event.struct.operations.objectType}	Object Type		v			v	v								

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.struct.operations.subjects.name}	Subject		v			v	v								
${Event.struct.databases.databaseName}	Database name from which the audited item originated	v	v			v	v								
${Event.struct.tableGroups.displayName}	Table Group to which the audited information is associated		v			v	v								
${Event.struct.tableGroups.sensitive}	Is the item is sensitive? (True/False)		v			v	v								
${Event.struct.operations.privileged}	Is the item is privileged? (True/Flase)		v			v	v								
${Event.sourceInfo.streamId}	Stream ID.		v				v								
${Event.struct.rawData.bindVariables}	Bind variables in audited information		v			v	v								
${Event.struct.operations.commandGroups.commandGroupName}	The name of the command group		v			v									
${Event.struct.operations.subjectType}	The subject type of the operation		v			v									
${Event.struct.agentName}	Agent name from which the audited item originated.		v				v								
${Event.struct.logCollectorName}	Log collector name.		v				v								
${Event.struct.databases.databaseAvailability}	Is the database available? (True/False)		v			v									
${Event.struct.databases.schemaAvailability}	Is the schema available? (True/False)		v			v									
${Event.struct.sqlStructQueryValue.decodedRawData}	The raw query used in an event decoded		v			v									

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.struct.sqlStructQueryValue.parsedQuery}	The parsed query used in an event		v			v									
${Event.struct.sqlStructQueryValue.rawData}	The raw query used in an event		v			v									
${Event.struct.storedProcedures.storedProcedureName}	The stored procedure used in an event		v			v									
${Event.struct.tableGroups.black}	Is the table group black? (True/False)		v			v									
${Event.struct.httpRequest.cookies.name}	The cookie's name	v				v									
${Event.struct.httpRequest.cookies.traceabilityState}	The cookie's traceability state	v				v									
${Event.struct.httpRequest.cookies.value}	The cookie's value	v				v									
${Event.struct.httpRequest.headers.name}	The header's name	v				v									
${Event.struct.httpRequest.headers.value}	The header's value	v				v									
${Event.struct.httpRequest.headers.violatedHeader}	Is the header violated? (True/False)	v				v									
${Event.struct.httpRequest.parameters.bindingClassification}	The binding classification of the parameter	v				v									
${Event.struct.httpRequest.parameters.charType}	The character type of the parameter	v				v									
${Event.struct.httpRequest.parameters.ignored}	Is the parameter ignored? (True/False)	v				v									
${Event.struct.httpRequest.parameters.localeSize}	The size of the parameter	v				v									

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.struct.httpRequest.parameters.name}	The name of the parameter	v				v									
${Event.struct.httpRequest.parameters.paramOrigin}	The origin of the parameter	v				v									
${Event.struct.httpRequest.parameters.value}	The value of the parameter	v				v									
${Event.struct.httpRequest.parameters.violatedParameter}	Is the parameter violated? (True/False)	v				v									
${Event.struct.httpRequest.soapAction.soapParameters.charType}	The character type of the soap parameter	v				v									
${Event.struct.httpRequest.soapAction.soapParameters.localeSize}	The size of the soap parameter	v				v									
${Event.struct.httpRequest.soapAction.soapParameters.name}	The name of the soap parameter	v				v									
${Event.struct.httpRequest.soapAction.soapParameters.value}	The value of the soap parameter	v				v									
${Event.struct.httpResponse.cookies.domain}	The cookie's domain	v				v									
${Event.struct.httpResponse.cookies.expiration}	The cookie's expiration time	v				v									
${Event.struct.httpResponse.cookies.name}	The cookie's name	v				v									
${Event.struct.httpResponse.cookies.path}	The cookie's path	v				v									
${Event.struct.httpResponse.cookies.persistent}	Is the cookie persistent? (True/False)	v				v									

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.struct.httpResponse.cookies.secure}	is the cookie secure? (True/False)	v				v									
${Event.struct.httpResponse.cookies.value}	The cookie's value	v				v									
${Event.struct.httpResponse.headers.name}	The header's name	v				v									
${Event.struct.httpResponse.headers.value}	The header's value	v				v									
${Event.struct.httpResponse.headers.violatedHeader}	Is the header violated? (True/False)	v				v									
${Event.struct.isFolder}	Is the file a folder? (True/False)			v	v	v									
${Event.struct.originalFile.Path}				v	v		v								
${Event.struct.originalFile.parentPath}	The parent path of the original file (not including the file itself)			v	v	v									
${Event.struct.targetFile.path}	The path of the target file			v	v	v									
${Event.struct.profileItemId}	The ID of the profile item	v	v			v									
${Event.struct.profileItemType}	The type of the profile item	v	v			v									
${Event.violations.action}		v	v	v	v	v									
${Event.violations.alert.alertMetadata.alertName}	The name of the alert	v	v	v	v	v									
${Event.violations.alert.alertType}	The type of the alert	v	v	v	v	v									
${Event.violations.alert.createTime}	The alert update time	v	v	v	v	v									

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Event.violations.alert.description}	The alert description	v	v	v	v	v									
${Event.violations.alert.dn}	The alert number	v	v	v	v	v									
${Event.violations.alert.flag}	The alert flag (Important/Acknowledged/Dismissed/None)	v	v	v	v	v									
${Event.violations.alert.lastUpdateTime}	The update time of the alert	v	v	v	v	v									
${Event.violations.alert.serverGroupName}	The name of the attacked server group	v	v	v	v	v									
${Event.violations.alert.simulationMode}	Whether the alert was generated in simulation mode	v	v	v	v	v									
${Event.violations.alert.sourceIp}	The source IP of the alert	v	v	v	v	v									
${Event.violations.alert.username}	The app user name	v	v	v	v	v									
${Event.violations.description}	The description of the violation	v	v	v	v	v									
${Event.violations.infoAttributes.name}		v	v	v	v	v									
${Event.violations.infoAttributes.value}		v	v	v	v	v									
${Event.violations.metadata.displayName}	The display name of the violation	v	v	v	v	v									
${Event.violations.profileItemId}	The ID of the profile item	v	v			v									
${Event.violations.profileItemType}	The type of the profile item	v	v			v									
${Event.violations.violatedItem}	The violated item	v	v	v	v	v									

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${SecureSphereVersion}	The current version of SecureSphere	v	v	v	v	v	v	v	v	v			v	v	v
${Time}						v									
${ViolationDescription}	The description of the violation	v	v	v	v	v									
${Event.struct.rawData.rawData}	The raw transaction buffer.		v			v									
${Event.message}	The message of the system event	v	v	v	v			v							
${Event.username}	The user name that generated the system event	v	v	v	v			v							
${Event.subsystem}	The sub-system of the system event (for example: profile/audit/gateway)	v	v	v	v			v							
${Event.severity}	The severity of the system event	v	v	v	v			v							
${Event.severity.displayName}	The severity of the system event	v	v	v	v			v							
${ListOfIds}	List of Ids of the referenced items of this system event	v	v	v	v			v							
${Job.type}	The type of job.	v	v	v	v				v						
${Job.status}	The job's status.	v	v	v	v				v						
#time(${Job.creationTime})	The time the job was created.	v	v	v	v				v						
${Job.attachment}	The item to which the job was attached.	v	v	v	v				v						
${JobData.jobName}	The name of the job.	v	v	v	v				v						
${JobData.nextRunTime}	Next time the job is to be run.	v	v	v	v				v						

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${JobData.specificJobData.policy.name}	The job's policy name for assessments.	v	v	v	v				v						
${JobData.specificJobData.policy.displayName}	The job's policy name for active modules.	v	v	v	v				v						
${JobData.specificJobData.policy.description}	The Active Module job's policy description.	v	v	v	v				v						
${JobData.specificJobData.policy.parent.displayName}	The Active Module Name.	v	v	v	v				v						
${JobData.specificJobData.policy.parent.description}	The active module description.	v	v	v	v				v						
${JobData.specificJobData.policy.parent.version}	The Active Module version.	v	v	v	v				v						
${JobData.specificJobData.policy.parent.oldVersion}	The Active Module old version.	v	v	v	v				v						
${JobData.specificJobData.policy.parent.scriptType}	Script type (OS/SQL/Both).	v	v	v	v				v						
${JobData.specificJobData.policy.group.name}	Assessment policy group name.	v	v	v	v				v						
${JobData.jobName}	The assessment job's name.	v	v	v	v				v						
${JobData.specificJobData.osConnections.ip}	The job's OS connection IP.	v	v	v	v				v						
${JobData.specificJobData.reportItem.displayName}	The job's report name.	v	v	v	v				v						
${JobData.specificJobData.reportItem.reportDescription}	The job's report description.	v	v	v	v				v						

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${JobData.specificJobData.reportItem.userItem.name}	The job's report's owner.	v	v	v	v				v						
${JobData.actionSet.displayName}	The job's actions.	v	v	v	v				v						
${JobData.scheduleItem.scheduleType}	The job's schedule type.	v	v	v	v				v						
#date(${JobData.scheduleItem.startDate}	) The job's schedule start time.	v	v	v	v				v						
${JobData.scheduleItem.timeOfDay}	The job's schedule time.	v	v	v	v				v						
${JobData.scheduleItem.oneTimeDate}	The job's one time schedule time.	v	v	v	v				v						
${JobData.scheduleItem.recurrenceType}	The job's schedule reoccurrence type.	v	v	v	v				v						
${JobData.scheduleItem.everyXDays}	The interval in days the job is scheduled for.	v	v	v	v				v						
${JobData.scheduleItem.daysOfWeek}	The days of the week the job is scheduled for.	v	v	v	v				v						
${JobData.scheduleItem.everyXWeeks}	The interval in months the job is scheduled for.	v	v	v	v				v						
${JobData.scheduleItem.monthlyRecurrenceType}	The job's monthly reoccurrence type.	v	v	v	v				v						
${JobData.scheduleItem.everyXMinutes}	The interval in minutes the job is scheduled for.	v	v	v	v				v						
${JobData.scheduleItem.dayOfMonth}	The day of the month the job is scheduled for.	v	v	v	v				v						
${JobData.scheduleItem.everyXMonthsForDayRecurrency}	The monthly interval.	v	v	v	v				v						
${JobData.scheduleItem.weekOfMonth}	The week of the month the job is scheduled for.	v	v	v	v				v						

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${JobData.scheduleItem.monthlyDayOfWeek}	The day of the week the monthly job is scheduled on.	v	v	v	v				v						
${JobData.scheduleItem.everyXMonthsForDayOfWeekRecurrency}	The monthly interval.	v	v	v	v				v						
${JobHistory.startTime}	Start time of the job run	v	v	v	v				v						
${JobHistory.endTime}	End time of the job run	v	v	v	v				v						
${JobHistory.status}		v	v	v	v				v						
${JobHistory.warnings.message}		v	v	v	v				v						
${JobHistory.warnings.severity}		v	v	v	v				v						
${Job.file}		v	v	v	v				v						
${Job.policyName}		v	v	v	v				v						
${JobData.lastStatus}		v	v	v	v				v						
${JobData.specificJobData.databaseConnections.connectionData.dbName}			v						v						
${JobData.specificJobData.databaseConnections.connectionData.dbUsername}			v						v						
${JobData.specificJobData.databaseConnections.connectionData.port}			v						v						
${JobData.specificJobData.databaseConnections.connectionData.serverName}			v						v						

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${JobData.specificJobData.databaseConnections.ip.ip}			v						v						
${JobData.specificJobData.jobName}		v	v	v	v				v						
${JobData.specificJobData.policy.parent.installed}		v	v		v				v						
${JobData.specificJobData.policy.parent.moduleFrontEnd}		v	v		v				v						
${JobData.specificJobData.policy.parent.schema}		v	v		v				v						
${ReportLink}	A url containing a link to the report's result	v	v	v	v				v		v				
#wfdTags(${Event} ${Alert})		v										v			
${Task.name}	Name of the task.	v	v	v	v								v		
${Task.subsystem}	Subsystem to which the task belongs.	v	v	v	v								v		
${Task.status}	Status of the task.	v	v	v	v								v		
${Task.priority}	Priority of the task.	v	v	v	v								v		
${Task.owner.name}	Owner of the ask.	v	v	v	v								v		
${Task.assignee.name}	Person to whom the task is assigned.	v	v	v	v								v		
${Task.previousAssignee.name}	Previous person to whom the task was assigned if it has been reassigned.	v	v	v	v								v		
${Task.assignmentDate}	Date the task was assigned.	v	v	v	v								v		
${Task.dueDate}	Date the task is due.	v	v	v	v								v		

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${Task.closingDate}	Date the task was closed.	v	v	v	v								v		
${Task.statusChangedFollowedAction.displayName}	Name of the task followed action whose status was changed.	v	v	v	v								v		
${Task.assigneeChangedFollowedAction.displayName}	Name of the task followed action whose assignee was changed.	v	v	v	v								v		
${Task.overdueFollowedAction.displayName}	Name of overdue tasks.	v	v	v	v								v		
${Task.notifyOwner}	Owner of task.	v	v	v	v								v		
${Task.assignee.dn}	Id of the person to whom the task is assigned.	v	v	v	v								v		
${Task.comments.contents}	Contents of the comment posted on the task	v	v	v	v								v		
${Task.comments.postingTime}	Posting time of the comment posted on the task	v	v	v	v								v		
${Task.comments.postingUser}	The user name that posted the comment on the task	v	v	v	v								v		
${Task.referredResource}	A name of an external resource reffered to by the task (for example: report result)	v	v	v	v								v		
${am}	The name of the active module	v	v	v	v					v					
${endDate}	The end date of the active module run	v	v	v	v					v					
${PolicyName}	The name of the policy of the active module	v	v	v	v		v			v				v	
{SourceOfActivity}							v								
${startDate}	The start date of the active module run	v	v	v	v					v				v	

Placeholders		SecureSphere Product				Event type									
Placeholder Name	Description	WAF	DAM	CIFS	SharePoint	Security Event	Audit Event	System Event	Job Run	Active Modules Archiving	Report	FPS	Task	Archive Files	Vulnerability
${gatewayName}	The name of the gateway that archived the files		v	v	v									v	
${splitIndex}			v	v	v									v	
${Item.Asset}	The asset of the vulnerability	v	v												v
${Item.Asset.Credentials.Port}	The port of the database in which the vulnerability was found	v	v												v
${Item.Asset.ServerGroup.DisplayName}	The display name of the server group in which the vulnerability was found	v	v												v
${Item.Asset.Service.DisplayName}	The display name of the service in which the vulnerability was found	v	v												v
${Item.Asset.Site.DisplayName}	The display name of the site in which the vulnerability was found	v	v												v
${Item.Owner.DisplayName}	The name of the owner of the vulnerability	v	v												v
${Item.Risk}	Risk calculated for this vulnerability	v	v												v
${Item.SerialNumber}	Serial number of the vulnerability	v	v												v
${Item.Status}	The status of the vulnerability (Open/Closed/Mitigated/ Deleted/Excluded)	v	v												v
${Item.Type.DisplayName}	The name of the type of the vulnerability	v	v												v
${Item.TypeDescription}	The description of the type of the vulnerability	v	v												v

List of Collection Placeholders

The following table lists all available collection placeholders. Syntax for using these placeholders can be seen at Working with Collection Placeholders on page 495.

Collection Placeholders

Placeholder Name	List Name	Item Name	Description
Alert Actions	Alert.actions	• description • actionSetName • duration • actionTime	Lists details regarding actions taken about a violation that generated an alert.
Job Warning	JobHistory.warnings	• message • severity	Lists Job Warnings and their properties that were involved in an event.
Task Comments	Task.comments	• contents • postingUser • postingTime	Lists Task Comments and their properties that were involved in an event.
Irregular Behaviors	Event.irregularBehaviors	• metadata.displayName • alert.alertMetadata.alertName • alert.description • action • infoAttributes (nested collection access using #innerlist)	Lists all irregular behavior on an event including learning, monitoring and security violations, etc. Note that this contains a nested collection infoAttributes that provides access to all the events details. For more information, see Understanding the Nested Collection Placeholder on page 496.
Violations	Event.violations	• metadata.displayName • alert.alertMetadata.alertName • action • infoAttributes (nested collection access using #innerlist) • alert.dn (violation ID) • alert.createTime • alert.lastUpdateTime • alert.description • alert.username (app user name) • alert.alertType (violation type: Firewall, Signature, Protocol, Profile) • alert.sourceIp • alert.simulationMode • alert.serverGroupName	List Security violations on an event. Note that this contains a nested collection infoAttributes that provides access to all the events details. For more information, see Understanding the Nested Collection Placeholder on page 496.

Placeholder Name	List Name	Item Name	Description
Irregular Behavior and Violation Collection Attributes	infoAttributes	• Name • Value	Accessed only using the #innerlist macro from within the Events.irregularBehaviour or Events.violations collections. For more examples, see Working with Collection Placeholders on page 495.
Database Connection	JobData.specificJobData.databaseConnections	• ip.ip • connectionData.dbUserncon nectionData.ame • dbName • connectionData.port • connectionData.serverName	Lists a database job and properties that were involves with the event. For use with assessments and active modules.
Stored Procedure	Event.struct.storedProcedures	• storedProcedureName	Lists stored procedures that were involved in an event.
Table Group	Event.struct.tableGroups	• displayName • sensitive • black	Lists table groups and their properties that were involved in an event.
Database	Event.struct.databases	• database • databaseName • schema • databaseAvailability • schemaAvailability	Lists data bases and their properties that were involved in an event.
Operation	Event.struct.operations	• name • privileged • storedProcedure • objectType • subjectType • commandGroups • subjects • objects	Lists operations and their properties that were involved in an event
HTTP Request Headers	Event.struct.httpRequest.headers	• name • value • violatedHeader (Boolean)	Lists HTTP Request Headers and their properties that were involved in an event.

Placeholder Name	List Name	Item Name	Description
HTTP Request Parameters	Event.struct.httpRequest.parameters	• name • value • paramOrigin • bindingClassification (hit, miss, na) • ignored • violatedParameter (Boolean) • charType • localeSize (=value length) • origin (numeric)	Lists HTTP Request Parameters and their properties that were involved in an event.
HTTP Request Soap Action Parameter	Event.struct.httpRequest.soapAction.soapParameters	• name • value • charType • localeSize	Lists HTTP Request SOAP Action Parameters and their properties that were involved in an event.
HTTP Response Header	Event.struct.httpResponse.headers	• name • value • violatedHeader (Boolean)	Lists HTTP Response Headers and their properties that were involved in an event.
HTTP Request Cookies	Event.struct.httpRequest.cookies	• name • value • traceabilityState (unchanged,hit or miss)	Lists HTTP Request Cookies and their properties that were involved in an event.
HTTP Response Cookies	Event.struct.httpResponse.cookies	• name • value • path • domain • expiration (string) • persistent (Boolean) • secure (Boolean)	Lists HTTP Response Cookies and their properties that were involved in an event.

APPENDIX E

Rule.dn Associated IDs

This Appendix describes IDs that are used with the Rule.DN placeholder that can be part of a SecureSphere event sent out in syslog format. This section reviews the following:

About Rule.dns 525
List of Rule.Dn Associated IDs 525

About Rule.dns

The Rule.dn placeholder represents a numerical value which is used to map the severity of rule/alert in SecureSphere. It can be added to an event syslog message and its output provides a unique numerical ID per policy. This information can assist in analyzing events and their sources.

List of Rule.Dn Associated IDs

The below is a list of rule.dn associated IDs.



Note: Some Rule.dns are in alphabetic and not numeric format.

Rule DN	Rule Name	Policy Name
1	Unknown Parameter	Web Profile Policy

Rule DN	Rule Name	Policy Name
10	Too Many Cookies in a Request	Web Protocol Policy
100	TCP - Bad State	Network Protocol Violations Policy
101	TCP - TCP Unexpected SYN	Network Protocol Violations Policy
102	UDP - Micro-Fragment	Network Protocol Violations Policy
103	SQL protocol - invalid header	SQL Protocol Policy
104	SQL protocol - invalid flags	SQL Protocol Policy
105	SQL protocol - invalid header size	SQL Protocol Policy
106	SQL protocol - invalid authentication	SQL Protocol Policy
107	SQL protocol - invalid login	SQL Protocol Policy
108	SQL protocol - invalid statement	SQL Protocol Policy
109	SQL protocol - invalid length	SQL Protocol Policy
11	Too Many of the Same Response Code	Web Correlation Policy
110	SQL protocol - invalid message type	SQL Protocol Policy
114	SQL protocol - unauthorized operation	SQL Protocol Policy
118	Illegal Content Type	Web Protocol Policy
119	Illegal Chunk Size	Web Protocol Policy
12	Too Many URL Parameters	Web Protocol Policy
120	Illegal Byte Code Character in Method	Web Protocol Policy
121	NULL Character in Method	Web Protocol Policy
122	Illegal Byte Code Character in URL	Web Protocol Policy
123	NULL Character in Url	Web Protocol Policy
124	Illegal Byte Code Character in Query String	Web Protocol Policy
125	NULL Character in Query String	Web Protocol Policy
126	Illegal Byte Code Character in Parameter Value	Web Protocol Policy
127	NULL Character in Parameter Value	Web Protocol Policy
13	URL is Above Root Directory	Web Protocol Policy
14	Unknown HTTP Request Method	Web Protocol Policy
15	Malformed URL	Web Protocol Policy
16	Malformed HTTP Header Line	Web Protocol Policy
17	Abnormally Long Request	Web Protocol Policy
18	Abnormally Long Header Line	Web Protocol Policy
19	Illegal URL Path Encoding	Web Protocol Policy
2	Required Parameter Not Found	Web Profile Policy
20	Illegal Parameter Encoding	Web Protocol Policy

Rule DN	Rule Name	Policy Name
20000001	Custom Violation	Anti Google Hacking - 1
20000002	Custom Violation	Anti Google Hacking - 2
20000016	Custom Violation	Unauthorized Privileged Operation - Deprecated
20000017	Custom Violation	Direct changes to data dictionary
20000018	Custom Violation	Automated Site Reconnaissance/Access
20000019	Custom Violation	Automated Vulnerability Scanning
20000020	Custom Violation	Malformed HTTP Attack (Non compatible HTTP Results Error code)
20000021	Custom Violation	Suspected parameter tampering - Deprecated
20000022	Custom Violation	Oracle - Attempt to Execute Database Export
20000025	Custom Violation	Sensitive Error Messages Leakage
20000026	Custom Violation	Directory Browsing Detection
20000027	Custom Violation	Data Leakage - U.S Social Security Number
20000028	Custom Violation	Privacy Violation - Credit Card Number Insertion
20000029	Custom Violation	Privacy Violation - Credit Card Number Insertion by Internal IP Address
20000030	Custom Violation	Privacy Violation - Credit Card Number Insertion by non Internal IP Address
20000033	Custom Violation	Unsuccessful Directory Browsing
20000034	Custom Violation	Apache Expect Header XSS
20000035	Custom Violation	MSSQL Data Leakage through Errors
20000036	Custom Violation	Webdav Method Detection
20000037	Custom Violation	Oracle - Attempt to Create Wrapped Object
20000038	Custom Violation	Suspicious Response Code
20000039	Custom Violation	Oracle - PL/SQL Code Tampering
20000040	Custom Violation	Cross Site Request Forgery
20000043	Custom Violation	IIS Code Upload
20000044	Custom Violation	WEB MISC Unauthorized File Access
20000045	Custom Violation	Hazardous HTTP request methods
20010002	Custom Violation	SOX - Unauthorized data changes
20010003	Custom Violation	SOX - Data changes by administrator
20010004	Custom Violation	SOX - Manual data changes
20010005	Custom Violation	SOX - Direct changes to data dictionary
20020001	Custom Violation	EBS - Suspected Activity in User Admin

Rule DN	Rule Name	Policy Name
		Tables
20020002	Custom Violation	EBS - Suspected Access by a Non-Default User
20020003	Custom Violation	EBS - Suspected Connection by a Default User
20020004	Custom Violation	EBS PCI - Unauthorized access to credit card no.
20020005	Custom Violation	EBS PCI - Violation to credit card number table
20020006	Custom Violation	EBS PCI - Unauthorized access to credit cardholder
20020007	Custom Violation	EBS PCI - Violation to credit cardholder
20030001	Custom Violation	PCI - Unauthorized access to cardholder information
20030002	Custom Violation	PCI - Usage of default user accounts
20030003	Custom Violation	PCI - Violations to a cardholder information table
20030005	Custom Violation	PCI - Violations of privileged commands
20030006	Custom Violation	PCI - Existence alerts of Track data
20030007	Custom Violation	PCI - Failed privileged operations on users and privileges management
20030008	Custom Violation	PCI - Attempted users and privileges management privileged operations by non-privileged user
20030009	Custom Violation	PCI - Violation to a cardholder information table
20030010	Custom Violation	PCI - Violations caused by admin
20040001	Custom Violation	SAP - Suspected Activity in User Administration Tables
20040002	Custom Violation	SAP - Suspected Activity in Data Dictionary Tables
20040003	Custom Violation	SAP - Suspected Activity in Confidential Financial Data and Vendor Tables
20040005	Custom Violation	SAP - Suspected Activity in User Master Data Tables
20040006	Custom Violation	SAP - Suspected Activity in Accounting Documents Tables
20040007	Custom Violation	SAP - Suspected Activity in Personal Info Tables
20040008	Custom Violation	SAP - Querying of Tables by Non-Default User
20040009	Custom Violation	SAP - Suspected modifications in Tables by

Rule DN	Rule Name	Policy Name
		Non-Default User
20040010	Custom Violation	SAP - Querying of Tables with a Non-Default Application
20040011	Custom Violation	SAP - Suspected Access to Tables with a Non-Default Application
20040012	Custom Violation	SAP - Suspected modification of SAP Metadata
20040013	Custom Violation	SAP PCI - Unauthorized access to payment card no.
20040014	Custom Violation	SAP PCI - Violation to payment card number table
20040015	Custom Violation	SAP PCI - Unauthorized access to credit cardholder
20040016	Custom Violation	SAP PCI - Violation to payment cardholder
20050001	Custom Violation	HIPAA - Unauthorized data access
20050002	Custom Violation	HIPAA - Unauthorized data modification
20050003	Custom Violation	HIPAA - Attempt to backup database
20060001	Custom Violation	PeopleSoft - Access to PeopleSoft Schema
20060002	Custom Violation	PeopleSoft - Querying Tables by Non-Default Users
20060003	Custom Violation	PeopleSoft - Querying with Unknown Application
20060004	Custom Violation	PeopleSoft - Querying Tables from Unknown IPs
20060005	Custom Violation	PeopleSoft - Changes by Non-Default Users
20060006	Custom Violation	PeopleSoft - Changes by Non-Default Applications
20060007	Custom Violation	PeopleSoft - Changes from Unknown IPs
20060008	Custom Violation	PeopleSoft - Changes to Login Tables
20060009	Custom Violation	PeopleSoft - Changes to User Admin Tables
20060010	Custom Violation	PeopleSoft - Query of Payment Card Info Columns
20060011	Custom Violation	PeopleSoft - Query of Identification Info Columns
20060012	Custom Violation	PeopleSoft - Query of Personal Info Columns
20060013	Custom Violation	PeopleSoft - DB Management by Non-Default Users
20060014	Custom Violation	PeopleSoft - DB Management by Non-Default Apps
20060015	Custom Violation	PeopleSoft - DB Management from Unknown IPs

Rule DN	Rule Name	Policy Name
20440001	Custom Violation	ThreatRadar - Malicious IPs
20440002	Custom Violation	ThreatRadar - Anonymous Proxies
20440003	Custom Violation	ThreatRadar - Phishing URLs
20440004	Custom Violation	ThreatRadar - TOR IPs
21	Extremely Long Parameter	Web Protocol Policy
21000002	Custom Violation	HTTP Response Splitting Vulnerability
21000003	Custom Violation	WEB-FRONTPAGE-Access to Sensitive Internal Information
21000004	Custom Violation	WEB-FRONTPAGE- External Access to Internal Information
21000005	Custom Violation	eMail Hoarding
21000007	Custom Violation	WEB-FRONTPAGE- Access to Internal Information
21000008	Custom Violation	IE Discussion Bar- Access to Internal Information
21000009	Custom Violation	Data Leakage - Application Source Code
21000010	Custom Violation	Data Leakage - Developer Comments
21000011	Custom Violation	Data Leakage - Visa, Short Credit Card Numbers
21000012	Custom Violation	Data Leakage - Visa, Long Credit Card Numbers
21000013	Custom Violation	Data Leakage - Diner's Club / Carte Blanche Credit Card Numbers
21000014	Custom Violation	Data Leakage - enRoute Credit Card Numbers
21000015	Custom Violation	Data Leakage - JCB Credit Card Numbers
21000016	Custom Violation	Data Leakage - American Express Credit Card Numbers
21000017	Custom Violation	Data Leakage - MasterCard Credit Card Numbers
21000018	Custom Violation	Data Leakage - Discover Credit Card Numbers
21000019	Custom Violation	Directory Traversal (In Cookies/Parameters Value)
21000020	Custom Violation	Directory Traversal (In URL)
21000021	Custom Violation	Directory Traversal (In URL) - Basic Rule
21000023	Custom Violation	Plain Vanilla Scanner Detection
21000024	Custom Violation	OS Command Injection
21000025	Custom Violation	Fullwidth/Halfwidth Unicode Decoding
21000026	Custom Violation	File Download Injection

Rule DN	Rule Name	Policy Name
22	NULL Character in Header Name	Web Protocol Policy
23	NULL Character in Parameter Name	Web Protocol Policy
24	Illegal Byte Code Character in Header Name	Web Protocol Policy
25	Illegal Byte Code Character in Parameter Name	Web Protocol Policy
26	Illegal Host Name	Web Protocol Policy
261	Illegal Content Length	Web Protocol Policy
27	Double URL Encoding	Web Protocol Policy
28	Redundant UTF-8 Encoding	Web Protocol Policy
29	Too Many Headers per Request	Web Protocol Policy
3	Parameter Value Length Violation	Web Profile Policy
30	Illegal HTTP Version	Web Protocol Policy
301-36	Sql Signature Violation	Recommended Policy for Database Applications - Legacy
301-39	Sql Signature Violation	Recommended Policy for Database Applications - Legacy
301-42	Sql Signature Violation	Recommended Policy for Database Applications - Legacy
302-37	Stream Signature Violation	Recommended Policy for General Applications - Legacy
302-40	Stream Signature Violation	Recommended Policy for General Applications - Legacy
302-43	Stream Signature Violation	Recommended Policy for General Applications - Legacy
31	Illegal Response Code	Web Protocol Policy
32	Too Many Headers per Response	Web Protocol Policy
33	Parameter Type Violation	Web Profile Policy
34	Parameter Read Only Violation	Web Profile Policy
35	Web Worm	Web Worm Policy
36	Unauthorized Access to Service	Firewall Policy
37	Unauthorized Method for Known URL	Web Profile Policy
38	Untraceable Database User	SQL Profile Policy
39	Unauthorized Host	SQL Profile Policy
40	Unauthorized OS User	SQL Profile Policy
41	Unauthorized Database and Schema	SQL Profile Policy
42	Login Statement Error	Oracle SQL Protocol Policy
43	Excessive Attempts of Database Login	SQL Correlation Policy

Rule DN	Rule Name	Policy Name
44	Excessive Login Attempts in Mid Database Session	SQL Correlation Policy
45	Access to a black-listed table	SQL Profile Policy
46	Unauthorized Sensitive Query Group	SQL Profile Policy
47	Unauthorized Table/Operation Access	SQL Profile Policy
48	Unauthorized Sensitive Table	SQL Profile Policy
49	Unauthorized Sensitive Query	SQL Profile Policy
5	Unauthorized Source IP Address	SQL Profile Policy
50	SSL Untraceable Connection	Network Protocol Violations Policy
5000	Incorrect cookie signature	Cookie Signing Policy
5002	Missing cookie signature	Cookie Signing Policy
5035	Cross-site scripting	Web Correlation Policy
5037	SQL injection	Web Correlation Policy
56	Unauthorized URL Access	Web Profile Policy
57	Cookie Tampering	Web Profile Policy
58	Cookie Injection	Web Profile Policy
59	Reuse of Expired Session's Cookie	Web Profile Policy
6	Unauthorized Database User	SQL Profile Policy
60	Time of Day Violation	SQL Profile Policy
61	Unauthorized Source Application	SQL Profile Policy
62	Unauthorized SOAP Action	Web Profile Policy
63	SOAP Element Value Length Violation	Web Profile Policy
64	SOAP Element Value Type Violation	Web Profile Policy
65	Required XML Element Not Found	Web Profile Policy
66	Unknown SOAP Element	Web Profile Policy
67	Malformed SOAP Message	Web Protocol Policy
68	Unauthorized Request Content Type	Web Protocol Policy
69	SOAP Access to a Non-SOAP URL	Web Profile Policy
7	Unauthorized Query Group	SQL Profile Policy
70	Non-SOAP Access to a SOAP Only URL	Web Profile Policy
71	Redundant HTTP Headers	Web Protocol Policy
72	Fragmented Packet	Network Protocol Violations Policy
75	PCI Compliance - Card Track Data Detection	SQL Correlation Policy
76	Session Attribute Changes	Web Correlation Policy
77	Forceful Browsing	Web Correlation Policy

Rule DN	Rule Name	Policy Name
78	IP - Bad Total Length	Network Protocol Violations Policy
79	Bad Source IP Address	Network Protocol Violations Policy
8	Unauthorized Query	SQL Profile Policy
80	Invalid IP Flags	Network Protocol Violations Policy
80000001-35	HTTP Signature Violation	Recommended Policy for Web Applications - Legacy
80000001-38	HTTP Signature Violation	Recommended Policy for Web Applications - Legacy
80000001-41	HTTP Signature Violation	Recommended Policy for Web Applications - Legacy
80000003-41	HTTP Signature Violation	Recommended Signatures Policy for Web Applications
80000003-88	HTTP Signature Violation	Recommended Signatures Policy for Web Applications
80000003-91	HTTP Signature Violation	Recommended Signatures Policy for Web Applications
80000003-92	HTTP Signature Violation	Recommended Signatures Policy for Web Applications
80000003-93	HTTP Signature Violation	Recommended Signatures Policy for Web Applications
80000003-95	HTTP Signature Violation	Recommended Signatures Policy for Web Applications
80000004-42	Sql Signature Violation	Recommended Signatures Policy for Database Applications
80000004-86	Sql Signature Violation	Recommended Signatures Policy for Database Applications
80000004-89	Sql Signature Violation	Recommended Signatures Policy for Database Applications
80000005-43	Stream Signature Violation	Recommended Signatures Policy for General Applications
80000005-87	Stream Signature Violation	Recommended Signatures Policy for General Applications
80000005-90	Stream Signature Violation	Recommended Signatures Policy for General Applications
81	IP - Micro-Fragment	Network Protocol Violations Policy
82	IP - Bad Fragment	Network Protocol Violations Policy
83	Bad IP Option Padding	Network Protocol Violations Policy
84	Unknown IP Option	Network Protocol Violations Policy
85	Bad IP Option Length	Network Protocol Violations Policy
86	IP - Too Small TTL	Network Protocol Violations Policy
87	ICMP - LAND Attack	Network Protocol Violations Policy

Rule DN	Rule Name	Policy Name
88	TCP - LAND Attack	Network Protocol Violations Policy
89	TCP - Micro-Fragment	Network Protocol Violations Policy
9	Attempt to Execute Privileged Operation	SQL Profile Policy
90	TCP - SYN with FIN	Network Protocol Violations Policy
91	TCP - SYN with Data	Network Protocol Violations Policy
92	TCP - SYN with RST (not SYN ACK)	Network Protocol Violations Policy
93	TCP - SYN with Non Zero ACK Sequence	Network Protocol Violations Policy
94	TCP - Non Zero Reserved Flags	Network Protocol Violations Policy
95	Bad TCP Option Length	Network Protocol Violations Policy
96	Bad TCP Option Padding	Network Protocol Violations Policy
97	Unknown TCP Option	Network Protocol Violations Policy
98	Illegal TCP Option	Network Protocol Violations Policy
99	Modified TCP Retransmission Data (client to server only)	Network Protocol Violations Policy
http-event-too-long	Extremely Long HTTP Request	Web Protocol Policy
http-illegal-byte-code-header-value	Illegal Byte Code Character in Header Value	Web Protocol Policy
http-null-char-header-value	NULL Character in Header Value	Web Protocol Policy
http-post-missing-content-type	Post Request - Missing Content Type	Web Protocol Policy
invalid-ocsp-asn1-encoding	Invalid ASN.1 Request Encoding	OCSP Protocol Policy
invalid-ocsp-base64-encoding	Invalid Base64 Request Encoding	OCSP Protocol Policy
invalid-ocsp-content-type	Invalid Message Content Type	OCSP Protocol Policy
invalid-ocsp-request-structure	Invalid Request Structure	OCSP Protocol Policy
invalid-ocsp-version	Invalid version	OCSP Protocol Policy
ocsp-message-too-long	OCSP Message Too Long	OCSP Protocol Policy
ocsp-too-many-queries	Too many queries within a Request	OCSP Protocol Policy
sql-too-long	Extremely Long SQL Request	SQL Protocol Policy
tcp-incompatible-data-len	TCP - Invalid Data Length in Header	Network Protocol Violations Policy

APPENDIX F

ArcSight Placeholders and Syntax

The integration of ArcSight into the SecureSphere workflow is based on the sending of syslog messages specially formatted with placeholders that are configured in the message field of System Log properties (as described in Configuring Integration with ArcSight). These placeholders are used to define a syslog based event using the ArcSight Common Event Format. The basic format of the syslog event is as follows:

CEF:Version|DeviceVendor|DeviceProduct|DeviceVersion|deviceEventClassId|Name|Severity|Extension

- For a description of available macros that can be used with ArcSight placeholders, see Macros for use with ArcSight Placeholders on page 535
- For example of syslog event messages, see Examples of Syslog Events for ArcSight on page 536
- For a list of fields that can be used in building a syslog message for ArcSight, see CEF Fields on page 537

Macros for use with ArcSight Placeholders 535
Examples of Syslog Events for ArcSight 536
CEF Fields 537

Macros for use with ArcSight Placeholders

CEF message contains 2 parts: a message and an extension. The message includes everything up to and including the severity, while the extension contains all other information. There are certain symbols that are not supported in ArcSight. So in order for ArcSight to be able to properly parse information in messages and extensions, they need to use supported characters.

The below placeholders output data generated in SecureSphere for ArcSight converting symbols into their supported format. These placeholders can be used in action sets, then attached as followed actions to Security policies meant for ArcSight.



Note: For a list of ArcSight fields and the equivalent SecureSphere placeholders, see Standard Fields on page 538 and Extended Fields on page 540.

Macro	Description
#cefEscapeMessage(${placeholder})	Converts a single backlash (\) in a message into a double backslash (\\), or converts a single pipe (|) into a backslash pipe (\\|).
#cefEscapeMessageList(${placeholder} "<full_field_name>")	Converts the single backlashes (\) in a list of messages into double backslashes (\\), or converts single pipes (|) into backslash pipes (\\|), where **placeholder** points to the list and **<full_field_name>** points to the value of each element in the list. **Note:** There should be a space before "<full_field_name>".
#cefEscapeExtension(${placeholder})	Converts a single backlash (\) in a message into a double backslash (\\), Converts an equal sign (=) in extensions to a backslash equal sign (\=), and converts a new line character (\n) or (\r) to a new line string (\n) or (\r).
#cefEscapeExtensionList(${placeholder} "<full_field_name>")	Converts the equal signs (=) in a list of extensions to backslash equal signs (\=), where **placeholder** points to the list and **<full_field_name>** points to the value of each element in the list. **Note:** There should be a space before "<full_field_name>".
#arcsightDate(${placeholder})	Converts dates to ArcSight supported format: MMM dd yyyy HH:mm:ss
#arcsightDateList(${placeholder} "<full_field_name>"	Converts a list of dates to ArcSight supported format: MMM dd yyyy HH:mm:ss, where **placeholder** points to the list and **<full_field_name>** points to the value of each element in the list. **Note:** There should be a space before "<full_field_name>".

EXAMPLE: DISPLAYING POLICY DISPLAY NAME

To output a list of policy names using supported symbols, use the following placeholder.

```
#cefEscapeMessage(${Rule.parent.displayName})
```

Examples of Syslog Events for ArcSight

A number of types syslog events are available, they include:

Regular Security Events

Security events indicate that a security policy violation has taken place. The following is an example of syntax used to build a syslog message for reporting a **regular** security event to ArcSight.

```
CEF:0|Imperva Inc.|SecureSphere|[SecureSphere version #]|
${Alert.alertType}|${Alert.alertMetadata.alertName}|${Alert.severity} |act=${Alert.immediateAction}
dst=${Event.destInfo.serverIp} dpt=${Event.destInfo.serverPort} duser=${Alert.username}
src=${Event.sourceInfo.sourceIp} spt=${Event.sourceInfo.sourcePort} proto=${Event.sourceInfo.ipProtocol}
rt=#arcsightDate(${Alert.createTime}) cat=Alert cs1=${Rule.parent.displayName} cs1Label=Policy
cs2=${Alert.serverGroupName} cs2Label=ServerGroup cs3=${Alert.serviceName} cs3Label=ServiceName
cs4=${Alert.applicationName} cs4Label=ApplicationName cs5=${Alert.description} cs5Label=Description
```

Custom Security Events

Security events indicate that a security policy violation has taken place. The following is an example of syntax used to build a syslog message for reporting a **custom** security event to ArcSight.

```
CEF:0|Imperva Inc.|SecureSphere|[SecureSphere version #]|
${Rule.parent.displayName}|${Rule.parent.displayName}|${Alert.severity} |act=${Alert.immediateAction}
dst=${Event.destInfo.serverIp} dpt=${Event.destInfo.serverPort} duser=${Alert.username}
src=${Event.sourceInfo.sourceIp} spt=${Event.sourceInfo.sourcePort} proto=${Event.sourceInfo.ipProtocol}
rt=#arcsightDate${Alert.createTime}) cat=Alert cs1=${Rule.parent.displayName} cs1Label=Policy
cs2=${Alert.serverGroupName} cs2Label=ServerGroup cs3=${Alert.serviceName} cs3Label=ServiceName
cs4=${Alert.applicationName} cs4Label=ApplicationName cs5=${Alert.description} cs5Label=Description
```

Firewall Security Events

Firewall Security events indicate a Firewall related issue has occurred. The following is an example of syntax used to build a syslog message for reporting a **firewall** event to ArcSight.

```
CEF:0|Imperva Inc.|SecureSphere|[SecureSphere version #]|
${Alert.alertType}|${Alert.alertMetadata.alertName}|${Alert.severity} |act=${Alert.immediateAction}
dst=${Event.destInfo.serverIp} dpt=${Event.destInfo.serverPort} duser=${Alert.username}
src=${Event.sourceInfo.sourceIp} spt=${Event.sourceInfo.sourcePort} proto=${Event.sourceInfo.ipProtocol}
rt=#arcsightDate(${Alert.createTime}) cat=Alert cs1=${Rule.parent.displayName} cs1Label=Policy
cs2=${Alert.serverGroupName} cs2Label=ServerGroup cs3=${Alert.description} cs3Label=Description
```

System Events

System events indicate a system related issue has occurred. The following is an example of syntax used to build a syslog message for reporting a **system** event to ArcSight.

```
CEF:0|Imperva
Inc.|SecureSphere|8.0|${Event.eventType}|${Event.message}|${Event.severity.displayName}|suser=${Event.us
ername} rt=#arcsightDate(${Event.createTime}) cat=SystemEvent
```

CEF Fields

CEF fields are added in the message field of System Log properties. These fields are used to create a syslog message that can be read by ArcSoft. There are two categories of CEF fields that can be used in syslog messages:

- Standard Fields on page 538
- Extended Fields on page 540

Standard Fields

Standard fields enables the reporting of core syslog functions from SecureSphere to ArcSight. The following tables details standard CEF fields and its corresponding SecureSphere placeholders.

CEF Field Name	*Version*
CEF Definition	Version is an integer that identifies the version of the CEF format. Event consumers use this information to determine the other fields of the message. Currently only version 0 (zero) is established in the CEF format. The other fields might need to be added to the "prefix" and therefore require a version number change. Adding new formats is handled through the standards body.
Configuration in SecureSphere	0
SecureSphere Definition	N/R

CEF Field Name	*DeviceVendor*
CEF Definition	Device Vendor, Device Product, and Device Version are strings that uniquely identify the type of device that sends events. Two products cannot use the same device-vendor and device product pair. There is no central authority managing these pairs. Event producers have to ensure that they assign a unique name to each pair.
Configuration in SecureSphere	Imperva Inc.
SecureSphere Definition	Company Name

CEF Field Name	*DeviceProduct*
CEF Definition	Device Vendor, Device Product, and Device Version are strings that uniquely identify the type of device that sends events. Two products cannot use the same device-vendor and device product pair. There is no central authority managing these pairs. Event producers have to ensure that they assign a unique name to each pair.
Configuration in SecureSphere	SecureSphere
SecureSphere Definition	Product Name

CEF Field Name	*DeviceVersion*

CEF Definition	Device Vendor, Device Product and Device Version are strings that identify the type of device that sends events. Two products cannot use the same device-vendor and device product pair. There is no central authority managing these pairs. Event producers have to ensure that they assign a unique name to each pair.
Configuration in SecureSphere	6.0.6 and up
SecureSphere Definition	Product Version

CEF Field Name	*deviceEventClassId*
CEF Definition	DeviceEventClassId is a unique identifier for each event type. This can be a string or an integer. DeviceEventClassId represents the type of event reported. In the intrusion detection system (IDS) world, each signature or rule that detects certain activity has a unique deviceEventClassId assigned. This is a requirement for other types of devices as well, and helps correlation engines deal with the events.
Configuration in SecureSphere	${Alert.alertType} for security alerts other than custom policy alerts ${Rule.parent.displayName} for custom policy security alerts ${Event.eventType} for system events
SecureSphere Definition	${Alert.alertType} is the alert type (firewall, signature, protocol, profile, or correlation) ${Rule.parent.displayName} is the name of the custom policy ${Event.eventType} is the type of system event

CEF Field Name	*Name*
CEF Definition	Name is a string that represents a human-readable and understandable description of the event. The event name must not contain information that is specifically mentioned in other fields. For example, "Port scan from 10.0.0.1 targeting 20.1.1.1" is not a good event name. The name should be: "Port scan". The rest of information is redundant and can be picked up from the other fields.
Configuration in SecureSphere	${Alert.alertMetadata.alertName} for security alerts other than custom policy alerts ${Event.message} for system events ${Rule.parent.displayName} for custom policy security alerts
SecureSphere Definition	${Alert.alertMetadata.alertName} is the alert name ${Event.message} is the message of the event ${Rule.parent.displayName} is the name of the custom policy

CEF Field Name	*Severity*

CEF Definition as appears in CEF documentation	Severity reflects the importance of the event.
Configuration in SecureSphere	${Alert. severity} for alerts ${Event.severity.displayName} for system events
SecureSphere Definition	Severity should not be set to Informative when CEF is used. Use Low instead. ${Alert.severity} is the severity of the alert in text format: Low, Medium, High. ${Event.severity.displayName} is the severity of the event in text format: Low, Medium, High.

CEF Field Name	*Extension*
CEF Definition as appears in CEF documentation	Extension is a collection of key-value pairs. Each key is a part of a predefined set. The standard allows you to include additional keys as outlined later. An event can contain any number of key-value pairs in any order separated by spaces (" "). A field can include spaces, i.e. in case of file name.
Configuration in SecureSphere	Security Event (Alert): act=${Alert.immediateAction} dst=${Event.destInfo.serverIp} dpt=${Event.destInfo.serverPort} duser=${Alert.username} src=${Event.sourceInfo.sourceIp} spt=${Event.sourceInfo.sourcePort} proto=${Event.sourceInfo.ipProtocol} rt=#arcsightDate(${Alert.createTime}) cat=Alert cs1=${Rule.parent.displayName} cs1Label=Policy cs2=${Alert.serverGroupName} cs2Label=ServerGroup cs3=${Alert.serviceName} cs3Label=ServiceName cs4=${Alert.applicationName} cs4Label=ApplicationName cs5=${Alert.description} cs5Label=Description Firewall Event (Alert): act=${Alert.immediateAction} dst=${Event.destInfo.serverIp} dpt=${Event.destInfo.serverPort} duser=${Alert.username} src=${Event.sourceInfo.sourceIp} spt=${Event.sourceInfo.sourcePort} proto=${Event.sourceInfo.ipProtocol} rt=#arcsightDate(${Alert.createTime}) cat=Alert cs1=${Rule.parent.displayName} cs1Label=Policy cs2=${Alert.serverGroupName} cs2Label=ServerGroup cs3=${Alert.description} cs3Label=Description System Event: User: ${Event.username} Creation Time: #arcsightDate (${Event.createTime}) cat=SystemEvent
SecureSphere Definition	The definition of each placeholder is listed under the Extension Field Dictionary.

Extended Fields

Extended fields provide the ability to use the CEF key-value pairs for additional information on the event. The following table details a CEF key and its corresponding SecureSphere placeholders.

CEF Key	*deviceFacility*
CEF Definition	The facility generating the event.
SecureSphere Placeholder	N/A
SecureSphere Definition	Choose the desired facility.
SecureSphere Event Type	All events

CEF Key	*act*
CEF Definition	Action mentioned in the event.
SecureSphere Placeholder	${Alert.immediateAction}
SecureSphere Definition	The immediate action performed, either block transaction (event) or no action.
SecureSphere Event Type	Security event.

CEF Key	*dst*
CEF Definition	Identifies the destination an event refers to in an IP network in IPv4 format. For example: "192.168.10.1".
SecureSphere Placeholder	${Event.destInfo.serverIp}
SecureSphere Definition	The destination IP address.
SecureSphere Event Type	Security events.

CEF Key	dpt
CEF Definition	Identifies the destination by port number. Valid port numbers are between 0 and 65535.
SecureSphere Placeholder	${Event.destInfo.serverPort}
SecureSphere Definition	The destination port.
SecureSphere Event Type	Security events.

CEF Key	duser
CEF Definition	Identifies the destination user by name. This parameter represents the user associated with the event's destination.
SecureSphere Placeholder	${Alert.username}

SecureSphere Definition	The destination user. In web applications it refers to the application user logged into the application. In database applications it refers to the database user.
SecureSphere Event Type	Security events.

CEF Key	src
CEF Definition	Identifies source an event refers to in an IP network in IPv4 format. For example: "192.168.10.1".
SecureSphere Placeholder	${Event.sourceInfo.sourceIp}
SecureSphere Definition	The source IP address.
SecureSphere Event Type	Security events.

CEF Key	spt
CEF Definition	Identifies the source by port number. Valid port numbers are between 0 and 65535.
SecureSphere Placeholder	${Event.sourceInfo.sourcePort}
SecureSphere Definition	The source port.
SecureSphere Event Type	Security events.

CEF Key	proto
CEF Definition	Identifies the Layer-4 protocol used. Possible values are protocol names, i.e. TCP or UDP.
SecureSphere Placeholder	${Event.sourceInfo.ipProtocol}
SecureSphere Definition	The protocol used.
SecureSphere Event Type	Security events.

CEF Key	rt
CEF Definition	The time when the event started. The format is MMM dd yyyy HH:mm:ss.
SecureSphere Placeholder	#arcsightDate(${Alert.createTime})
SecureSphere Definition	The alert time.
SecureSphere Event Type	Security events.

CEF Key	cat
CEF Definition	Represents the category assigned to the originating device. Usually devices use their own categorization schema to classify events.
SecureSphere Placeholder	Alert
SecureSphere Definition	The type of event.
SecureSphere Event Type	Security events.

CEF Key	cs1
CEF Definition	Custom field used to map fields that do not fit into any other field available in the CEF dictionary.
SecureSphere Placeholder	${Rule.parent.displayName}
SecureSphere Definition	The violated policy's name.
SecureSphere Event Type	Security events.

CEF Key	cs2
CEF Definition	Custom field used to map fields that do not fit into any other field available in the CEF dictionary.
SecureSphere Placeholder	${Alert.serverGroupName}
SecureSphere Definition	The server group name.
SecureSphere Event Type	Security events.

CEF Key	cs3
CEF Definition	Custom field used to map fields that do not fit into any other field available in the CEF dictionary.
SecureSphere Placeholder	For Firewall events: ${Alert.description} For other security events: ${Alert.serviceName}
SecureSphere Definition	${Alert.description} is the alert description ${Alert.serviceName} is the service name
SecureSphere Event Type	Security events.

CEF Key	cs4
CEF Definition	Custom field used to map fields that do not fit into any other field available in the CEF dictionary.
SecureSphere Placeholder	For non firewall security events: ${Alert.applicationName}
SecureSphere Definition	${Alert.applicationName} is the application name.
SecureSphere Event Type	Security events.

CEF Key	cs5
CEF Definition	Custom field used to map fields that do not fit into any other field available in the CEF dictionary.
SecureSphere Placeholder	For non firewall security events: ${Alert.description}
SecureSphere Definition	${Alert.description} is the alert description
SecureSphere Event Type	Security events.

CEF Key	cs1Label
CEF Definition	All custom fields have a corresponding label field for the description of the field.
SecureSphere Placeholder	Policy
SecureSphere Definition	Policy label.
SecureSphere Event Type	Security events.

CEF Key	cs2Label
CEF Definition	All custom fields have a corresponding label field for the description of the field.
SecureSphere Placeholder	ServerGroup
SecureSphere Definition	ServerGroup Label.
SecureSphere Event Type	Security events.

CEF Key	cs3Label
CEF Definition	All custom fields have a corresponding label field for the description of the

	field.
SecureSphere Placeholder	For non Firewall alerts Service For Firewall Alerts: Description
SecureSphere Definition	Service is Service Label. Description is Description Label
SecureSphere Event Type	Security events.

CEF Key	cs4Label
CEF Definition	All custom fields have a corresponding label field for the description of the field.
SecureSphere Placeholder	For non Firewall alerts Application
SecureSphere Definition	Application is Service Label.
SecureSphere Event Type	Security events.

CEF Key	cs5Label
CEF Definition	All custom fields have a corresponding label field for the description of the field.
SecureSphere Placeholder	For non Firewall alerts Description
SecureSphere Definition	Description is Description Label.
SecureSphere Event Type	Security events.

CEF Key	suser
CEF Definition	Identifies the source user by name. This field represents the user associated with the event's source.
SecureSphere Placeholder	${Event.username}
SecureSphere Definition	The system user who caused the event. It can be a specific user who logged into the system or a system user.
SecureSphere Event Type	System events.

CEF Key	cat
CEF Definition	Represents the category assigned to the originating device. Usually devices use their own categorization schema to classify events.
SecureSphere Placeholder	System Event
SecureSphere Definition	The type of the event.
SecureSphere Event Type	System events.

CEF Key	rt
CEF Definition	The time when the activity the event referred to started. The format is MMM dd yyyy HH:mm:ss.
SecureSphere Placeholder	#arcsightDate(${Event.createTime})
SecureSphere Definition	The system event time.
SecureSphere Event Type	System events.

APPENDIX G

Event Types

This Appendix describes SecureSphere event types. The following topics provide information regarding event types:

About Event Types 547
List of SecureSphere Events 547

About Event Types

Event Types are used to assist you in a number of locations in SecureSphere. They are used to create system event policies, are displayed in alert and violations, and can be included in reports. SecureSphere uses two primary events:

- **Specific Events**: Standard categories of events that are used in SecureSphere and displayed from the new policy event dropdown menu.
- **Regular Events**: General events that are used across SecureSphere and are identified by message strings. For more information, see the SecureSphere User Guide.

The following table lists the various events available in SecureSphere, their subsystem, severity, and message. When creating policies you can search for "Matching Text Segments" which are part of these messages. For more information see Creating a System Events Policy on page 411.

List of SecureSphere Events

The following table provides a comprehensive list of SecureSphere events and their associated message. These messages can subsequently be used when creating policies by using the Matching Text Segments option.

Type	Sub System	Severity	Deployment Type	Message
Access violation	Security	High	Both	Access Violation: {}
Active modules	ADC	Low	MX	Install of {} started.
Active modules	ADC	Low	MX	Install of {} has succeeded.
Active modules	ADC	Low	MX	Install of {} has failed. Error information: {}
Active modules	ADC	Low	MX	Uninstall of {} started.
Active modules	ADC	Low	MX	Uninstall of {} has succeeded.
Active modules	ADC	Low	MX	Uninstall of {} has failed. Error information: {}
Active modules	ADC	Low	MX	{} upgrade started.
Active modules	ADC	Low	MX	{} upgrade has succeeded.
Active modules	ADC	Low	MX	{} upgrade has failed. Error information: {}
Active modules	ADC	Low	MX	Import of {} started.
Active modules	ADC	Low	MX	Import of {} has succeeded.
Active modules	ADC	Low	MX	Import of {} has failed. Error information: {}.
Active modules	ADC	Low	MX	Export of {} from {} to {} started.
Active modules	ADC	Low	MX	Export of {} from {} to {} has succeeded.
Active modules	ADC	Low	MX	Export of {} from {} to {} has failed. Error information: {}.
Active modules	ADC	Low	MX	Failed to load the AM jars
Active modules	ADC	Low	MX	Process of {} started.
Active modules	ADC	Low	MX	Process of {} has succeeded.
Active modules	ADC	Low	MX	Process of {} has failed for targets: {}.
Active modules	ADC	Low	MX	Process of {} was aborted.
Active modules	ADC	Low	MX	Process of {} has failed. Error information: {}.
Active modules	ADC	Low	MX	Clean up of {} started.
Active modules	ADC	Low	MX	Clean up of {} has succeeded.
Active modules	ADC	Low	MX	Clean up of {} has failed. Error information: {}.
Active modules	ADC	Low	MX	Export of {} from {} to {} started.
Active modules	ADC	Low	MX	Export of {} from {} to {} has succeeded.
Active modules	ADC	Low	MX	Export of {} from {} to {} has failed. Error information: {}.
ADC automatic upload checked	ADC	Low	Both	ADC automatic upload checked
ADC download failure	ADC	High	Both	ADC content upload has failed. Error information: {}.
ADC download succeeded	ADC	Medium	Both	ADC content upload has succeeded.
Agent CPU usage	Agent	High	MX	Agent "{}" at IP {} (id={}) CPU usage exceeds {}%.
Agent data interfaces changed	Agent	Medium	MX	New agent channel found: Service Type={}, Channel Type = {}, Server Id = {}, Server Path = {}, Listener Name = {},

Type	Sub System	Severity	Deployment Type	Message
				Ip Address = {}, Port = {}, Interface Name = {}, No Configuration = {}, Is Online = {}, Is Encrypted = {}, Is Monitored = {}
Agent data interfaces changed	Agent	Medium	MX	Agent channel updated: Service Type={}, Channel Type = {}, Server Id = {}, Server Path = {}, Listener Name = {}, Ip Address = {}, Port = {}, Interface Name = {}, No Configuration = {}, Is Online = {}, Is Encrypted = {}, Is Monitored = {}
Agent data interfaces changed	Agent	Medium	MX	Agent channel deleted: Service Type={}, Channel Type = {}, Server Id = {}, Server Path = {}, Listener Name = {}, Ip Address = {}, Port = {}, Interface Name = {}, No Configuration = {}, Is Online = {}, Is Encrypted = {}, Is Monitored = {}
Agent Move To GW Group Failed	Agent	Low	MX	Failed to Move Agent {} to GW Group {}
Agent Move To GW Group Succeeded	Agent	Low	MX	Agent {} was moved to GW Group {}
Agent Operation Finished successfully	Agent	Low	MX	Agent: {} finished running operation: {}. Status : {}.
Agent registration	Agent	Low	MX	New agent created - "{}" at IP {} (id={}), registered successfully to gateway {}.
Agent registration	Agent	Low	MX	Agent "{}" at IP {} (id={}) registered successfully to secondary gateway {}.
Agent registration	Agent	Low	MX	Failed to create agent "{}" at IP {} and register to gateway {}.
Agent registration	Agent	Low	MX	Failed to create agent "{}" at IP {} and register to gateway {}.
Agent registration	Agent	Low	MX	New agent created - "{}" at IP {} (id={}), registered successfully to gateway {} in cluster {}.
Agent registration	Agent	Low	MX	Failed to create agent "{}" at IP {} and register to gateway {} in cluster {}.
Agent status changed	Agent	High	MX	Status of agent "{}" at IP {} (id={}) changed from {} to {}.
Agent status changed	Agent	High	MX	Status of agent "{}" at IP {} (id={}) changed from {} to {}. Error messages: {}.
Agent status changed	Agent	High	MX	Agent "{}" at IP {} (id={}) started.
Agent status changed	Agent	High	MX	The status of agent {} in gateway {} changed to {}
Agent status changed	Agent	High	MX	Agent "{}" at IP {} (id={}) started. Error messages: {}.
Alert Flag Changed	Alerts	Information	MX	Alert {} flag has changed to {}
Appliance became disconnected	Appliance	High	Both	Gateway {} became disconnected.
Appliance became disconnected	Appliance	High	Both	MX {} became disconnected.
Archiving has ended	Maintenance	Medium	Both	Failed to archive {}

Type	Sub System	Severity	Deployment Type	Message
	Tasks			
Archiving has ended	Maintenance Tasks	Medium	Both	Archived {} successfully
Assessment error	Assessment	High	MX	$errorMessage
Audit archive ended successfully	Audit	Low	MX	Audit archive created for policy: {}, time frame: {} - {}
Audit error	Audit	High	MX	Failed to retrieve audit index data to management server, for policy: {}
Audit error	Audit	High	MX	Failed to archive data for policy: {}
Audit error	Audit	High	MX	Audit fast viewing generation failed
Audit error	Audit	High	MX	Failed to purge data on all gateways
Audit error	Audit	High	MX	Failed to upgrade audit legacy data on gateway {}
Audit error	Audit	High	MX	Failed to retrieve audit index data to management server due to table-space limit, for policy: {}
Audit error	Audit	High	MX	Audit archive operation on gateway: {} failed, for policy: {}
Audit error	Audit	High	MX	Audit purge operation on gateway: {} failed
Audit error	Audit	High	MX	Failed to retrieve audit index data from gateway: {}, for policy: {}
Audit fast viewing ended successfully	Audit	Low	MX	Audit fast viewing data was generated
Audit fields phase settings	Audit	Low	MX	Audit fields phase settings changed: {}
Audit gateway legacy data upgraded	Audit	Low	MX	Audit legacy data on gateway {} was upgraded successfully
Audit load warnings	Audit	High	MX	Audit table-space in management server database reached 80% usage
Audit load warnings	Audit	High	MX	Policy {} on gateway {} reached 80% of its quota
Audit load warnings	Audit	High	MX	Gateway {} reached 80% of disk space
Audit load warnings	Audit	High	MX	Policy {} on gateway {} reached 95% of its quota
Audit load warnings	Audit	High	MX	Fast view generation time exceeded one hour
Audit load warnings	Audit	High	MX	Policy {} on gateway {} lost audit data due to quota
Audit load warnings	Audit	High	MX	Audit table-space in management server database reached 80% usage
Audit load warnings	Audit	High	MX	Gateway {} lost audit data due to disk space limit
Audit load warnings	Audit	High	MX	Gateway {} lost audit data due to high system load
Audit load warnings	Audit	High	MX	Fast view generation time exceeded one hour
Audit load warnings	Audit	High	MX	Gateway {} reached 95% of disk space
Audit load warnings	Audit	High	MX	Audit table-space in management server database reached 95% usage

Type	Sub System	Severity	Deployment Type	Message
Audit load warnings	Audit	High	MX	Policy {} on gateway {} lost audit data due to quota
Audit load warnings	Audit	High	MX	Gateway {} reached 95% of disk space
Audit load warnings	Audit	High	MX	Gateway {} lost audit data due to high system load
Audit load warnings	Audit	High	MX	Policy {} on gateway {} reached 80% of its quota
Audit load warnings	Audit	High	MX	Policy {} on gateway {} reached 95% of its quota
Audit load warnings	Audit	High	MX	Gateway {} reached 80% of disk space
Audit load warnings	Audit	High	MX	Gateway {} lost audit data due to disk space limit
Audit load warnings	Audit	High	MX	Audit table-space in management server database reached 95% usage
Audit purge ended successfully	Audit	Low	MX	Audit data for policy {} created before {} was purged
Audit purge ended successfully	Audit	Low	MX	Audit data for policy {} created before {} was purged partially (several gateways not purged)
AWS snapshot creation failed	Config	High	Both	Snapshot creation failed for Management Server instance in Amazon Web Services account. See logs for more details
AWS snapshot was created	Config	Low	Both	A snapshot with ID {} was created for Management Server instance in Amazon Web Services account
Batch update	Vulnerability	Low	MX	Batch {} items was executed. New value is {}. Reason: {}
Blocked ambiguous packets	Gateway	High	MX	Gateway {} has blocked an average of {} ambiguous packets per minute in the last 5 minutes.
Collector down due to error	Gateway	High	MX	Collector {} is down due to error on gateway {}.
Collector running	Gateway	Low	MX	Collector {} is running on gateway {}.
Collector running	Gateway	Low	MX	Collector {} is running on gateway {}.
Collector running with errors	Gateway	Medium	MX	Collector {} is running with errors on gateway {}.
Collector terminated unexpectedly	Gateway	High	MX	Collector {} on gateway {} has terminated unexpectedly.
Collector terminated unexpectedly	Gateway	High	MX	Collector {} on gateway {} has terminated unexpectedly.
Community Defense was activated	Community Defense	Medium	MX	Community Defense was activated
Community Defense was deactivated	Community Defense	Medium	MX	Community Defense was deactivated because ThreatRadar services are not enabled for this MX
Configuration detached from MX	Config	Medium	MoM	The upgrade process has made {} not compatible with MXs of version prior to {}. This item has been detached from {}: {}.
Configuration detached from MX	Config	Medium	MoM	ADC content contains {} that is not compatible with MXs of version prior to {}. This item has been detached from {}: {}.
Controlling Gateway	Gateway	High	MX	The following Agent/s have changed their controlling

Type	Sub System	Severity	Deployment Type	Message
Changed				Gateway from {} to {}: {}
Create new Remedy incident	Management	Low	Both	Created new Remedy Incident ({}) for {}
Data overflow high threshold	Maintenance Tasks	High	Both	The Tablespace {} has run out of space. Threshold of {}% has exceeded. Used space: {}%
Data overflow low threshold	Maintenance Tasks	Medium	Both	The Tablespace {} is running low on space. Warning Threshold of {}% has exceeded. Used space: {}%
Db2 on z/OS Agent Version Changed	Agent	Low	MX	Agent version changed from {} to {}, and does not support policy '{}'.
Detection of Optimization issues was run	Profile	Low	MX	Detection of Optimization issues was run on {}.
Discovery: batch operation	Discovery	Low	MX	Batch {} was executed. New value is {}. Reason: {}
Do not learn any URLs without parameters	Profile	Low	MX	Application "{}" is now learning only URLs with parameters.
Entitlement run failed	Entitlement	Low	MX	Entitlement process run failed.
Entitlement run failed	Entitlement	Low	MX	Policy {} run failed on {} ({}@{}).
Entitlement run succeeded	Entitlement	Low	MX	Policy {} finished successfully on {} ({}@{}).
Environment failed registration	MX	Low	MoM	MX {} failed registration on SOM.
Environment failed re-registration	MX	Low	MoM	MX {} failed re-registration on SOM.
Environment registration	MX	Low	MoM	MX {} successfully registered to SOM.
Environment removed	MX	Low	MoM	MX {} successfully removed from SOM.
Environment re-registered	MX	Low	MoM	MX {} successfully re-registered to SOM.
External Authorization mode was changed	User	Medium	Both	External Authorization mode was disabled. All Externally Authorized users were transformed to Externally Authenticated users
External Authorization mode was changed	User	Medium	Both	External Authorization mode was enabled. All Externally Authenticated users were transformed to Externally Authorized users
Failed to apply full SOM configuration	Config	High	MX	Full SOM configuration failed.
Failed to create Remedy incident	Management	Medium	Both	Failed to create Remedy incident for {}
Failed to detach configuration from MX	Config	High	MoM	The upgrade process has made {} not compatible with MXs of version prior to {}. There was a failure in detaching {} from {}: {}. Please manually detach from {}: {}.
Failed to detach	Config	High	MoM	ADC content contains {} that is not compatible with MXs of

Type	Sub System	Severity	Deployment Type	Message
configuration from MX				version prior to {}. There was a failure in detaching {} from {}: {}. Please manually detach from {}: {}.
Failed to load Adc content	ADC	High	Both	ADC content file {} failed to load. Error information: {}.
Failed to localize default objects	User	High	Both	Failed to localize default objects after login
Failed to remove user from group	File User Rights	High	MX	Failed to revoke user(or group): {} from group: {}.
Failed to revoke permissions	File User Rights	High	MX	Failed to revoke permissions: {} of grantee: {} from: {}.
Failed to send full configuration to CX	Config	High	MX	Failed to send full configuration to CX {}.
Failed to send full configuration to gateway	Config	High	MX	Failed to send full configuration to gateway {}.
Failed to update dictionary	Config	High	Both	Update {} "{}" failed: {}
Failed to update lookup data set	Config	High	Both	Update {} "{}" failed: {}
Failed to upload license	Management	High	Both	License upload has failed.
Fail-over occurred	Appliance	Medium	MX	Fail-over occurred to the secondary server.
Fail-over occurred for MX	MX	Medium	MoM	Fail-over occurred for MX {}.
Fatal error. Monitoring table switch failed.	Maintenance Tasks	High	MX	Failed to execute table switch for monitoring statistic table: {} to index: {} . Please contact support.
Fatal error. Monitoring table switch failed.	Maintenance Tasks	High	MX	Failed to execute main monitoring table switch to index: {}. Reconciliation process failed. Please contact support.
File classification scan failed	File Classification	Low	MX	File Classification scan failed: {}
File classification scan failed	File Classification	Low	MX	File Classification scan failed - policy: {}, ip: {}, service: {}, error: {}.
File classification scan finished with warnings	File Classification	Low	MX	File Classification scan {} finished with warnings on {} ({})
File classification scan succeeded	File Classification	Low	MX	File Classification scan {} finished successfully on {} ({})
File user rights process failed	File User Rights	Low	MX	File URM process run failed: {}.
File user rights process failed	File User Rights	Low	MX	File URM process run failed - policy: {}, ip: {}, service: {}, error: {}.
File user rights process failed	File User Rights	Low	MX	File URM process run failed - policy: {}, error: {}.

Type	Sub System	Severity	Deployment Type	Message
File user rights process finished with warnings	File User Rights	Low	MX	Policy {} finished with warnings on {} ({})
File user rights process succeeded	File User Rights	Low	MX	Policy {} finished successfully on {} ({}).
Full configuration applied	Config	Low	MX	Full SOM configuration successfully applied.
Full configuration received	Config	Low	MX	Full configuration received from SOM.
Full configuration sent	Config	Low	MoM	Full configuration sent to MX {}.
Gateway block on overloaded streams	Gateway	Medium	MX	{} streams blocked on gateway {} due to overload.
Gateway counters reached max capacity	Config	Medium	MX	{}: amount of collected data reached max capacity. Collection will be stopped until some entries are removed. To reduce amount of data, you can turn off collection for certain appliances.
Gateway CPU utilization	Gateway	High	MX	The CPU utilization of gateway {} is {}%.
Gateway entered bypass	Gateway	High	MX	Gateway {} has entered bypass mode (fail-open). Traffic is no longer being monitored.
Gateway Group created automatically during gateway registration	Config	Information	MX	Gateway Group {} was created during automatic gateway registration
Gateway left bypass	Gateway	Medium	MX	Gateway {} has left bypass mode (fail-open) and resumed regular operation. Traffic is now being monitored.
Gateway low free space	Gateway	Medium	MX	On gateway {}: Low disk space. Data loss will occur if minimum disk space is reached.
Gateway low free space	Gateway	Medium	MX	On gateway {}: Low disk space. Data loss will occur if minimum disk space is reached.
Gateway minimal free space	Gateway	High	MX	On gateway {}: Data loss occurred due to minimum disk space limit.
Gateway Move Failed	Gateway	Low	MX	failed to move gateway {} from cluster {} to gateway group {} - {}
Gateway Move Failed	Gateway	Low	MX	failed to move gateway {} from gateway group {} to cluster {} - {}
Gateway Move Failed	Gateway	Low	MX	failed to move gateway {} from gateway group {} to gateway group {} - {}
Gateway Move Failed	Gateway	Low	MX	failed to move gateway {} from cluster {} to cluster {} - {}
Gateway Move Success	Gateway	Low	MX	Gateway {} was moved from cluster {} to cluster {} - successfully
Gateway Move Success	Gateway	Low	MX	Gateway {} was moved from cluster {} to gateway group {} - successfully

Type	Sub System	Severity	Deployment Type	Message
Gateway Move Success	Gateway	Low	MX	Gateway {} was moved from gateway group {} to cluster {} - successfully
Gateway Move Success	Gateway	Low	MX	Gateway {} was moved from gateway group {} to gateway group {} - successfully
Gateway overflow	Gateway	High	MX	The throughput of gateway {} has been exceeded.
Gateway registration failed	Management	High	MX	Gateway registration from {} has failed. {}
Gateway state change	Gateway	High	MX	The following error has occurred on gateway {}: {}. The gateway"s current status is {}.
Gateway state change	Gateway	High	MX	The status of the gateway {} is {}. Please check the gateways section of the user interface for additional information.
Gateway state change	Gateway	High	MX	Gateway {} is now up.
Gateway state change	Gateway	High	MX	Gateway {} resumed normal operation; all modules are running properly.
Gateway state change	Gateway	High	MX	The following error has occurred on gateway {}: {}. The gateway"s current status is {}.
Gateway state change	Gateway	High	MX	Gateway {} resumed normal operation; all modules are running properly.
Gateway state change	Gateway	High	MX	Gateway {} is now up.
Gateway throughput	Gateway	Medium	MX	The throughput of gateway {} is {} Mbps.
GeoLocation license has expired	Maintenance Tasks	High	MX	GeoLocation license has expired. You can renew the license from the ThreatRadar dashboard.
Hardware Failure	Appliance	High	MX	RAID hard-disk failure in {} device "{}" on {}. Details: {}
Hardware monitor failure	Appliance	High	MX	Appliance status retrieval failed on {}
Import external classifications process failed	File Classification	Low	MX	Import external classification process for file: {} failed.
Import external classifications process succeeded	File Classification	Low	MX	Import external classifications process for file: {} finished successfully.
Incomplete configuration sent to CX	Management	Medium	MX	Incomplete configuration sent to cx
Incomplete configuration sent to gateways	Management	Medium	MX	Incomplete configuration sent to gateways
Item expires Today	Config	Medium	MX	{}: {} expires today.
Item will expire soon	Config	Low	MX	{}: {} will expire in {} days.
Key exported	Security	High	Both	Key exported
Key imported	Security	High	Both	Key imported

Type	Sub System	Severity	Deployment Type	Message
Learn all URLs	Profile	Low	MX	Application "{}" is now learning all URLs.
Learn only non-static URLs	Profile	Low	MX	Application "{}" is now learning URLs without parameters only if they are not static.
License for Fraud Prevention Service has expired	Maintenance Tasks	High	MX	ThreatRadar Fraud Prevention Services - support license has expired - {}
License module about to expire	Management	Low	Both	The following licensed Modules will expire in {} days: {}
License uploaded successfully	Management	High	Both	License upload has succeeded.
Load ADC content succeeded	ADC	Medium	Both	ADC content file {} successfully loaded
Login failed	User	High	Both	Login failed for user {} (IP: {}) Reason: credentials expired
Login failed	User	High	Both	Login failed for External user {} (IP: {}) Reason: an internal user with the same name already exist
Login failed	User	High	Both	Login failed for user {} (IP: {}) Reason: User account is locked
Login failed	User	High	Both	Login failed for user {} (IP: {}) Reason: bad credentials
Login failed	User	High	Both	Login failed for user {} (IP: {}) Reason: Bad Connection
Login failed	User	High	Both	Drilldown failed for user {} (IP: {})
Login failed	User	High	Both	Login failed for user {} (IP: {}) Reason: account disabled
Login failed	User	High	Both	Login failed for External user {} (IP: {}) Reason: insufficient permissions
Monitoring table switch failed. Trying to reconcile.	Maintenance Tasks	High	MX	Failed to switch statistics monitoring table: {} to index: {}. Reconciliation process will be executed.
Monitoring table switch failed. Trying to reconcile.	Maintenance Tasks	High	MX	Failed to switch main monitoring tables to index: {}. Data overflow risk - please contact support.
MX Registered to SOM	Management	Low	MX	MX successfully registered to SOM {}.
MX Removed from SOM	Management	Low	MX	MX successfully removed from SOM {}.
MX Re-registered to SOM	Management	Low	MX	MX successfully re-registered to SOM {}.
MX synchronization status changed	MX	Low	MoM	Synchronization status of MX {} changed from {} to {}.
MX unregistered itself from SOM	MX	Low	MoM	MX unregistered itself from SOM.
Network Interface	Config	High	MX	Network interface {} was removed from gateway {} but is still in use by {}
Network interface was removed	Config	Low	MX	Network interface {} on gateway {} was removed

Type	Sub System	Severity	Deployment Type	Message
New MX trusted SSL Key	Config	Low	MoM	New trusted SSL Key from MX {} loaded for SOM-to-MX communication.
New SOM trusted SSL Key	Config	Low	MX	New SOM trusted SSL Key loaded for SOM-to-MX communication.
New SSL key	Config	Low	Both	New MX SSL Key loaded for SOM-to-MX communication.
New SSL key	Config	Low	Both	New SOM SSL Key loaded for SOM-to-MX communication.
New Trusted Certificate loaded in MX	Config	Low	MX	External system certificate loaded (SOM/LDAP)
New Trusted Certificate loaded in SOM	Config	Low	MoM	External system certificate loaded (MX/LDAP)
Old ADC content	ADC	Medium	Both	The ADC content has not been updated for the last {} days.
Optimization adjustment was executed	Profile	Low	MX	Adjustment "{}" executed on optimization issue "{}".
Package Distribution Complete	Software Update	Low	MX	Package {} distributed to component {}.
Package Distribution Failed	Software Update	High	MX	Failed to distribute package {} to component {}.
Package Distribution Start	Software Update	Information	MX	Distributing package {} to component {}.
Package Installation Complete	Software Update	Low	MX	Version {} installed on component {}.
Package Installation Failed	Software Update	High	MX	Failed to install version {} on component {}.
Package Installation Start	Software Update	Information	MX	Installing new version {} on component {}.
Password changed	User	Medium	Both	Password changed
Password reset	User	High	Both	Password reset for {}
Permissions changed	Permissions	Low	Both	User {} was granted {} permissions on item {}
Permissions changed	Permissions	Low	Both	Role {} was granted {} permissions on item {}
Permissions changed	Permissions	Low	Both	User {} permissions on item {} have been revoked
Permissions changed	Permissions	Low	Both	Role {} permissions on item {} have been revoked
Permissions changed	Permissions	Low	Both	User {} was granted {} permissions on group {}
Permissions changed	Permissions	Low	Both	Role {} was granted {} permissions on group {}
Permissions changed	Permissions	Low	Both	User {} permissions on group {} have been revoked
Permissions changed	Permissions	Low	Both	Role {} permissions on group {} have been revoked
Permissions changed	Permissions	Low	Both	User {} was granted permission on url {}
Permissions changed	Permissions	Low	Both	Role {} was granted permission on url {}

Type	Sub System	Severity	Deployment Type	Message
Permissions changed	Permissions	Low	Both	User {} privileges for url {} have been revoked
Permissions changed	Permissions	Low	Both	Role {} privileges for url {} have been revoked
Permissions changed	Permissions	Low	Both	User {} was granted permission to perform Activate Settings unconditionally
Permissions changed	Permissions	Low	Both	User {} was granted permission to perform Activate Settings if authorized to view all changes that will be activated
Permissions changed	Permissions	Low	Both	Role {} was granted permission to perform Activate Settings unconditionally
Permissions changed	Permissions	Low	Both	Role {} was granted permission to perform Activate Settings if authorized to view all changes that will be activated
Permissions changed	Permissions	Low	Both	User {} privileges to perform Activate Settings have been revoked
Permissions changed	Permissions	Low	Both	Permissions: {} of grantee: {} were revoked from: {}.
Permissions changed	Permissions	Low	Both	Failed to revoke permissions: {} of grantee: {} from: {}.
Permissions changed	Permissions	Low	Both	User {} was granted {} permissions on item {}
Permissions changed	Permissions	Low	Both	Role {} was granted {} permissions on item {}
Permissions changed	Permissions	Low	Both	User {} permissions on item {} have been revoked
Permissions changed	Permissions	Low	Both	Role {} permissions on item {} have been revoked
Permissions changed	Permissions	Low	Both	User {} was granted {} permissions on group {}
Permissions changed	Permissions	Low	Both	Role {} was granted {} permissions on group {}
Permissions changed	Permissions	Low	Both	User {} permissions on group {} have been revoked
Permissions changed	Permissions	Low	Both	Role {} permissions on group {} have been revoked
Permissions changed	Permissions	Low	Both	User {} was granted permission on url {}
Permissions changed	Permissions	Low	Both	Role {} was granted permission on url {}
Permissions changed	Permissions	Low	Both	User {} privileges for url {} have been revoked
Permissions changed	Permissions	Low	Both	Role {} privileges for url {} have been revoked
Permissions changed	Permissions	Low	Both	User {} was granted permission to perform Activate Settings unconditionally
Permissions changed	Permissions	Low	Both	User {} was granted permission to perform Activate Settings if authorized to view all changes that will be activated
Permissions changed	Permissions	Low	Both	Role {} was granted permission to perform Activate Settings unconditionally
Permissions changed	Permissions	Low	Both	Role {} was granted permission to perform Activate Settings if authorized to view all changes that will be activated
Permissions changed	Permissions	Low	Both	User {} privileges to perform Activate Settings have been revoked

Type	Sub System	Severity	Deployment Type	Message
Permissions changed	Permissions	Low	Both	Permissions: {} of grantee: {} were revoked from: {}.
Permissions changed	Permissions	Low	Both	Failed to revoke permissions: {} of grantee: {} from: {}.
Permissions were revoked	File User Rights	Low	MX	Permissions: {} of grantee: {} were revoked from: {}.
Platform account ID Validation failed	Management	High	MX	Cannot validate license: {}
Policy applied to MX	MX	Low	MoM	Policy {} applied to MXs {}.
Policy applied to services on MX	MX	Low	MoM	Policy {} applied to {} on MXs: {}.
Policy applied to services on MX	MX	Low	MoM	Policy {} applied to {}.
Policy changed	Config	Low	Both	Policy "{}" changed
Policy removed from MX	MX	Low	MoM	Policy {} removed from MXs {}.
Policy removed from services on MX	MX	Low	MoM	Policy {} removed from {} on MXs: {}.
Policy removed from services on MX	MX	Low	MoM	Policy {} removed from {}.
Profile size limit	Config	Low	MX	You have reached the allowed limit
Purging items has ended	Maintenance Tasks	Medium	Both	Purged {} successfully
Purging items has ended	Maintenance Tasks	Medium	Both	Failed to purge {}
Regular	Archive	Low	Both	Default archive settings changed
Regular	Config	Low	Both	Virtual router {} on gateway {} was changed
Regular	Config	Low	Both	Dictionary Group {} was created.
Regular	Config	Low	Both	Server Group Collection {} was deleted.
Regular	Config	Low	Both	Server Group {} - {} was created.
Regular	Config	Low	Both	URL Prefixes/Directories Group {} was deleted.
Regular	Config	Low	Both	Site {} was created.
Regular	Config	Low	Both	Susceptible Directory Group {} was created.
Regular	Config	Low	Both	Dictionary Group {} was deleted.
Regular	Config	Low	Both	Policy "{}" applied to {}
Regular	Config	Low	Both	Action Set "{}" changed
Regular	Config	Low	Both	IP address {} on gateway {} was created
Regular	Config	Low	Both	Privileged Stored Procedures And Operations List {} was created.
Regular	Config	Low	Both	Web Service {} - {} - {} was deleted.
Regular	Config	Low	Both	Command Group {} was created.

Type	Sub System	Severity	Deployment Type	Message
Regular	Config	Low	Both	Route to {} {} on gateway {} was deleted
Regular	Config	Low	Both	Server Group Collection {} was created.
Regular	Config	Low	Both	Alias {} on gateway {} was changed
Regular	Config	Low	Both	Teradata Service {} - {} - {} was deleted.
Regular	Config	Low	Both	Dictionary Group {} was updated.
Regular	Config	Low	Both	Sybase Service {} - {} - {} was created.
Regular	Config	Low	Both	SQL Table Group New Group was deleted.
Regular	Config	Low	Both	Web Error Page {} was deleted.
Regular	Config	Low	Both	Informix Service {} - {} - {} was created.
Regular	Config	Low	Both	Server Group {} - {} was deleted.
Regular	Config	Low	Both	System Event Policy "{}" changed
Regular	Config	Low	Both	Service Collection {} was deleted.
Regular	Config	Low	Both	Route to {} {} on gateway {} was created
Regular	Config	Low	Both	SQL Table Group New Group was created.
Regular	Config	Low	Both	IP address {} on gateway {} was deleted
Regular	Config	Low	Both	Alias {} on gateway {} was created
Regular	Config	Low	Both	DB2 Service {} - {} - {} was deleted.
Regular	Config	Low	Both	Sybase Service {} - {} - {} was deleted.
Regular	Config	Low	Both	Protocol Collection {} was deleted.
Regular	Config	Low	Both	SQL Stored Procedures List {} was deleted.
Regular	Config	Low	Both	File-Type Group {} was created.
Regular	Config	Low	Both	Parameter Group {} was created.
Regular	Config	Low	Both	Oracle Service {} - {} - {} was created.
Regular	Config	Low	Both	Privileged Stored Procedures And Operations List {} was deleted.
Regular	Config	Low	Both	Session Identifiers Group {} was created.
Regular	Config	Low	Both	Web Service {} - {} - {} was created.
Regular	Config	Low	Both	Service Collection Container {} was created.
Regular	Config	Low	Both	Action Set "{}" deleted
Regular	Config	Low	Both	Service Collection {} was created.
Regular	Config	Low	Both	SQL Application {} - {} - {} was deleted.
Regular	Config	Low	Both	Data Masking Group {} was updated.
Regular	Config	Low	Both	File-Type Group {} was deleted.
Regular	Config	Low	Both	MsSQL Service {} - {} - {} was deleted.
Regular	Config	Low	Both	Service Collection Container {} was deleted.

Type	Sub System	Severity	Deployment Type	Message
Regular	Config	Low	Both	IP Group {} was deleted.
Regular	Config	Low	Both	Session Identifiers Group {} was deleted.
Regular	Config	Low	Both	System Event Policy "{}" deleted
Regular	Config	Low	Both	Global Port List {} was created.
Regular	Config	Low	Both	URL Prefixes/Directories Group {} was created.
Regular	Config	Low	Both	Alias {} on gateway {} was deleted
Regular	Config	Low	Both	SOAP Element Group {} was deleted.
Regular	Config	Low	Both	Virtual router {} on gateway {} was deleted
Regular	Config	Low	Both	Action Set "{}" created
Regular	Config	Low	Both	IP Group {} was created.
Regular	Config	Low	Both	DB2 Service {} - {} - {} was created.
Regular	Config	Low	Both	MsSQL Service {} - {} - {} was created.
Regular	Config	Low	Both	Web Application {} - {} - {} was deleted.
Regular	Config	Low	Both	Virtual router {} on gateway {} was created
Regular	Config	Low	Both	#set ($a=$paramsHelper.getClassFromString("com.mprv.secsph.configuration.domain.PolicyItem"))#set ($b=$paramsHelper.getClassFromString("com.mprv.secsph.configuration.domain.ServiceItem"))#set ($policy=$event.getObject($a, true))#set ($sg=$event.getObject($b, true))#set ($polName = $itemGUIName.getName($policy, {}, {}, 50))#set ($sgName = $itemGUIName.getName($sg, {}, {}, {}))#mprvArgMessage("sysevent.info.policy.attach" [$polName , $sgName])
Regular	Config	Low	Both	System Event Policy "{}" created
Regular	Config	Low	Both	IP address {} on gateway {} was changed
Regular	Config	Low	Both	SOAP Element Group {} was created.
Regular	Config	Low	Both	Parameter Group {} was deleted.
Regular	Config	Low	Both	Data Masking Group {} was deleted.
Regular	Config	Low	Both	Site {} was deleted.
Regular	Config	Low	Both	SQL Stored Procedures List {} was created.
Regular	Config	Low	Both	Web Error Page {} was created.
Regular	Config	Low	Both	Command Group {} was deleted.
Regular	Config	Low	Both	Susceptible Directory Group {} was deleted.
Regular	Config	Low	Both	Policy "{}" detached from {}
Regular	Config	Low	Both	Informix Service {} - {} - {} was deleted.
Regular	Config	Low	Both	Web Application {} - {} - {} was created.
Regular	Config	Low	Both	Policy "{}" applied to {}

Type	Sub System	Severity	Deployment Type	Message
Regular	Config	Low	Both	Oracle Service {} - {} - {} was deleted.
Regular	Config	Low	Both	SSH Proxy Endpoint {} was created
Regular	Config	Low	Both	Global Port List {} was deleted.
Regular	Config	Low	Both	Route to {} {} on gateway {} was changed
Regular	Config	Low	Both	Data Masking Group {} was created.
Regular	Config	Low	Both	Protocol Collection {} was created.
Regular	Config	Low	Both	SQL Application {} - {} - {} was created.
Regular	Config	Low	Both	SSH Proxy Endpoint {} was deleted
Regular	Config	Low	Both	Teradata Service {} - {} - {} was created.
Regular	Gateway	High	Both	Gateway {} has been removed
Regular	Gateway	High	Both	Gateway {} has been added
Regular	Gateway	Low	Both	Configuration changed for gateway group {}
Regular	Gateway	Low	Both	Configuration changed for gateway group {}
Regular	Gateway	Low	Both	Configuration changed for gateway group {}
Regular	Gateway	Low	Both	Gateway {} was added to group {}
Regular	Gateway	Low	Both	Configuration changed for gateway group {}
Regular	Gateway	Low	Both	Configuration changed for gateway group {}
Regular	Gateway	Low	Both	Configuration changed for gateway group {}
Regular	Gateway	Low	Both	Configuration changed for gateway group {}
Regular	Gateway	Low	Both	Configuration changed for gateway group {}
Regular	Gateway	Low	Both	Configuration changed for gateway group {}
Regular	Gateway	Low	Both	Gateway {} was removed from group {}
Regular	Info	Low	Both	Gateway group {} assigned to server group {} for IP address {}
Regular	Info	Low	Both	Gateway Group {} removed from server group {} for IP address {}
Regular	Info	Medium	Both	Operation mode for Server Group {} changed to {}
Regular	Profile	Low	Both	User group {} in server group SQL application: {} was updated. The sharing time restrictions preferences of the group were changed.
Regular	Profile	Low	Both	Query Group {} (in protect mode) was deleted from the profile of {} . SQL application: {}.
Regular	Profile	Low	Both	{} operation is allowed for table "{}" in the profile of "{}". SQL application: "{}". Changes in the same application will not be reported in the next 5 minutes.
Regular	Profile	Low	Both	Query Group {} was added to the profile of {} in protect mode. SQL application: {}.
Regular	Profile	Low	Both	URL {} (in learn mode) was updated. Web Application: {}.

Type	Sub System	Severity	Deployment Type	Message
				URL was switched from protect mode to learn mode.
Regular	Profile	Low	Both	Profile of SQL application {} was updated.
Regular	Profile	Low	Both	Query Group {} in SQL application: {} was updated.
Regular	Profile	Low	Both	URL {} (in learn mode) was updated. Web Application: {}.
Regular	Profile	Low	Both	Parameter '{}' in URL '{}' was updated. Web Application: {}. The parameter type was updated.
Regular	Profile	Low	Both	Parameter '{}' in URL '{}' was updated. Web Application: {}. The parameter is not a prefix.
Regular	Profile	Low	Both	URL {} was added to the profile in learn mode. Web Application: {}.
Regular	Profile	Low	Both	Folder root directory was deleted from the profile. Web Application: {}.
Regular	Profile	Low	Both	URL Pattern {} (in protect mode) was updated. Web Application: {}. URLs matching the pattern are valid links (not broken).
Regular	Profile	Low	Both	URL Pattern {} (in protect mode) was updated. Web Application: {}.
Regular	Profile	Low	Both	URL {} (in learn mode) was updated. Web Application: {}. The URL was unlocked.
Regular	Profile	Low	Both	URL {} was added to the profile in protect mode. Web Application: {}.
Regular	Profile	Low	Both	Optimization issue "{}" created. Severity: {}.
Regular	Profile	Low	Both	Worm protected directory {} (in learn mode) was deleted. Web Application: {}.
Regular	Profile	Low	Both	User {} was deleted from the profile. SQL application: {}.
Regular	Profile	Low	Both	Cookie {} (in protect mode) in Web Application: {} was updated.
Regular	Profile	Low	Both	Folder {} was updated. Web Application: {}. The folder was unlocked.
Regular	Profile	Low	Both	Query Group {} was added to the profile of {} in protect mode. SQL application: {}.
Regular	Profile	Low	Both	Query Group {} (in protect mode) was deleted from the profile of {} . SQL application: {}.
Regular	Profile	Low	Both	URL {} (in learn mode) was updated. Web Application: {}. The URL is not a valid starting URL.
Regular	Profile	Low	Both	Parameter {} of URL pattern {} was updated. Web Application: {}. The parameter is not a prefix.
Regular	Profile	Low	Both	{} operation is not allowed for table "{}" in the profile of "{}". SQL application: "{}". Changes in the same application will not be reported in the next 5 minutes.
Regular	Profile	Low	Both	SOAP parameter {} in action {} in URL {} was updated. Web Application: {}.

Type	Sub System	Severity	Deployment Type	Message
Regular	Profile	Low	Both	URL {} (in learn mode) was updated. Web Application: {}. The URL is a valid link (not broken).
Regular	Profile	Low	Both	User group {} in server group SQL application: {} was updated.
Regular	Profile	Low	Both	Query Group {} in SQL application: {} was updated. The query group was switched from protect mode to learn mode.
Regular	Profile	Low	Both	SOAP parameter {} in action {} in URL {} was updated. Web Application: {}. The minimal allowed parameter length is {}
Regular	Profile	Low	Both	Query Group {} in SQL application: {} was updated.
Regular	Profile	Low	Both	Worm protected directory {} (in learn mode) was updated. Web Application: {}. Worm protected directory was switched from protect mode to learn mode.
Regular	Profile	Low	Both	Worm protected directory {} was added in protect mode. Web Application: {}.
Regular	Profile	Low	Both	Folder {} was deleted from the profile. Web Application: {}.
Regular	Profile	Low	Both	Cookie {} was added to the profile in learn mode. Web Application: {}.
Regular	Profile	Low	Both	Cookie {} (in protect mode) in Web Application: {} was updated. the cookie will be ignored.
Regular	Profile	Low	Both	Parameter '{}' in URL '{}' was updated. Web Application: {}. {} value is allowed.
Regular	Profile	Low	Both	URL {} (in protect mode) was updated. Web Application: {}. The URL is a valid link (not broken).
Regular	Profile	Low	Both	URL Pattern {} (in protect mode) was updated. Web Application: {}. URLs matching the pattern are not valid starting URLs.
Regular	Profile	Low	Both	URL {} (in learn mode) was updated. Web Application: {}. HTTP access is not allowed to this URL.
Regular	Profile	Low	Both	User group {} in server group SQL application: {} was updated. The sharing databeses preferences of the group were changed.
Regular	Profile	Low	Both	URL Pattern {} (in learn mode) was updated. Web Application: {}. URLs matching the pattern are valid links (not broken).
Regular	Profile	Low	Both	User {} was added to the profile. SQL application: {}.
Regular	Profile	Low	Both	Parameter '{}' in URL '{}' was updated. Web Application: {}. The parameter is unbound (changeable).
Regular	Profile	Low	Both	Worm protected directory {} (in learn mode) was updated. Web Application: {}.
Regular	Profile	Low	Both	URL {} (in protect mode) was updated. Web Application: {}. HTTP access is not allowed to this URL.

Type	Sub System	Severity	Deployment Type	Message
Regular	Profile	Low	Both	Cookie {} (in protect mode) in Web Application: {} was updated. Cookie was switched from learn mode to protect mode.
Regular	Profile	Low	Both	User group {} in server group SQL application: {} was updated. The sharing hostnames preferences of the group were changed.
Regular	Profile	Low	Both	Parameter '{}' in URL '{}' was updated. Web Application: {}. The maximal allowed parameter length is {}
Regular	Profile	Low	Both	Parameter {} of URL pattern {} was updated. Web Application: {}. the parameter is optional.
Regular	Profile	Low	Both	Parameter '{}' in URL '{}' was updated. Web Application: {}.
Regular	Profile	Low	Both	SOAP parameter {} in action {} in URL {} was updated. Web Application: {}. The parameter type was updated.
Regular	Profile	Low	Both	Cookie {} (in learn mode) was deleted from the profile. Web Application: {}.
Regular	Profile	Low	Both	User group {} in server group SQL application: {} was updated. The sharing privileged operations preferences of the group were changed.
Regular	Profile	Low	Both	URL {} (in protect mode) was updated. Web Application: {}. SOAP access is not allowed to this URL.
Regular	Profile	Low	Both	URL {} (in learn mode) was updated. Web Application: {}. SOAP access is not allowed to this URL.
Regular	Profile	Low	Both	HTTP method {} is allowed in URL {} . Web Application: {} .
Regular	Profile	Low	Both	Cookie {} was added to the profile in protect mode. Web Application: {}.
Regular	Profile	Low	Both	Parameter '{}' in URL '{}' was created. Web Application: {}.
Regular	Profile	Low	Both	Query Group {} in SQL application: {} was updated.
Regular	Profile	Low	Both	Folder {} was updated. Web Application: {}.
Regular	Profile	Low	Both	Folder root directory was updated. Web Application: {}.
Regular	Profile	Low	Both	URL Pattern {} (in learn mode) was deleted from the profile. Web Application: {}.
Regular	Profile	Low	Both	User group {} was added to the profile. SQL application: {}.
Regular	Profile	Low	Both	HTTP method {} is allowed in URL {} . Web Application: {} .
Regular	Profile	Low	Both	Parameter {} of URL pattern {} was updated. Web Application: {}.
Regular	Profile	Low	Both	HTTP method {} is not allowed in URL {} . Web Application: {} .
Regular	Profile	Low	Both	URL Pattern {} was added to the profile in protect mode. Web Application: {}.

Type	Sub System	Severity	Deployment Type	Message
Regular	Profile	Low	Both	Query {} was deleted from Query Group {} . SQL application: {}.
Regular	Profile	Low	Both	Prefix parameter created. Application: {}. URL: {}. Prefix: {}.
Regular	Profile	Low	Both	User group {} in server group SQL application: {} was updated. The sharing tables and queries preferences of the group were changed.
Regular	Profile	Low	Both	Worm protected directory {} (in protect mode) was deleted. Web Application: {}.
Regular	Profile	Low	Both	Worm protected directory {} (in protect mode) was updated. Web Application: {}.
Regular	Profile	Low	Both	SOAP parameter {} in action {} in URL {} was updated. Web Application: {}. The maximal allowed parameter length is {}
Regular	Profile	Low	Both	URL Pattern {} (in learn mode) was updated. Web Application: {}.
Regular	Profile	Low	Both	URL {} (in learn mode) was deleted from the profile. Web Application: {}.
Regular	Profile	Low	Both	URL {} (in protect mode) was updated. Web Application: {}.
Regular	Profile	Low	Both	Query Group {} in SQL application: {} was updated. The query group was switched to static protect mode.
Regular	Profile	Low	Both	URL {} (in protect mode) was deleted from the profile. Web Application: {}.
Regular	Profile	Low	Both	User group {} in server group SQL application: {} was updated. The sharing IPs preferences of the group were changed.
Regular	Profile	Low	Both	Table {} was deleted from the profile of {} . SQL application: {}.
Regular	Profile	Low	Both	URL {} (in protect mode) was updated. Web Application: {}. URL was switched from learn mode to protect mode.
Regular	Profile	Low	Both	HTTP method {} is not allowed in URL {} . Web Application: {} .
Regular	Profile	Low	Both	SOAP parameter {} in action {} in URL {} was updated. Web Application: {}. {} value is allowed.
Regular	Profile	Low	Both	User group {} in server group SQL application: {} was updated. The sharing OS users preferences of the group were changed.
Regular	Profile	Low	Both	URL {} (in protect mode) was updated. Web Application: {}. The URL is not a valid starting URL.
Regular	Profile	Low	Both	Parameter {} of URL pattern {} was updated. Web Application: {}. The minimal allowed parameter length is {}
Regular	Profile	Low	Both	Optimization issue "{}" status changed to {}.

Type	Sub System	Severity	Deployment Type	Message
Regular	Profile	Low	Both	SOAP parameter {} in action {} in URL {} was updated. Web Application: {}. the parameter is optional.
Regular	Profile	Low	Both	User group {} was deleted from the profile. SQL application: {}.
Regular	Profile	Low	Both	Query {} was added to Query Group {} . SQL application: {}.
Regular	Profile	Low	Both	Parameter '{}' in URL '{}' was updated. Web Application: {}. The minimal allowed parameter length is {}
Regular	Profile	Low	Both	Parameter '{}' in URL '{}' was updated. Web Application: {}. the parameter is optional.
Regular	Profile	Low	Both	Query Group {} in SQL application: {} was updated. The query group was switched to dynamic protect mode.
Regular	Profile	Low	Both	User group {} in server group SQL application: {} was updated. The sharing applications preferences of the group were changed.
Regular	Profile	Low	Both	Parameter {} of URL pattern {} was updated. Web Application: {}. {} value is allowed.
Regular	Profile	Low	Both	URL Pattern {} (in learn mode) was updated. Web Application: {}. URLs matching the pattern are not valid starting URLs.
Regular	Profile	Low	Both	Profile of SQL application {} was updated. The user list was unlocked.
Regular	Profile	Low	Both	URL Pattern {} (in protect mode) was deleted from the profile. Web Application: {}.
Regular	Profile	Low	Both	Cookie {} (in protect mode) was deleted from the profile. Web Application: {}.
Regular	Profile	Low	Both	URL Pattern {} was added to the profile in learn mode. Web Application: {}.
Regular	Profile	Low	Both	Query Group {} (in learn mode) was deleted from the profile of {} . SQL application: {}.
Regular	Profile	Low	Both	Worm protected directory {} (in protect mode) was updated. Web Application: {}. Worm protected directory was switched from learn to protect mode.
Regular	Profile	Low	Both	Table {} was added to the profile of {} . SQL application: {}.
Regular	Profile	Low	Both	Parameter {} of URL pattern {} was updated. Web Application: {}. The parameter type was updated.
Regular	Profile	Low	Both	Folder root directory was updated. Web Application: {}. The folder was unlocked.
Regular	Profile	Low	Both	Query Group {} was added to the profile of {} in learn mode. SQL application: {}.
Regular	Profile	Low	Both	URL {} (in protect mode) was updated. Web Application: {}. The URL was unlocked.

Type	Sub System	Severity	Deployment Type	Message
Regular	Profile	Low	Both	Parameter {} of URL pattern {} was updated. Web Application: {}. The maximal allowed parameter length is {}
Regular	Security	High	Both	Blocked Source Released {} {}
Regular	Security	High	Both	Blocked Source Released {} {}
Regular	Security	High	Both	Blocked Source Released {} {}
Regular	Security	High	Both	Blocked Source Released {} {}
Regular	User	High	Both	Password setting "#mprvMessage($tmp3)" has been changed to "{}"
Regular	User	High	Both	User {} removed
Regular	User	High	Both	Account enabled for user {}
Regular	User	High	Both	User {} created
Regular	User	High	Both	External user {} created upon first login
Regular	User	High	Both	External user {} removed since it was removed from LDAP
Regular	User	High	Both	External role {} removed
Regular	User	High	Both	Account disabled for user {}
Regular	User	High	Both	External role {} created
Regular	User	High	Both	Role {} removed
Regular	User	High	Both	Role {} created
Regular	User	Low	Both	User info changed for {}
Regular	User	Low	Both	Role {} was removed from external role {}
Regular	User	Low	Both	User info changed for {}
Regular	User	Low	Both	User {} is no longer read only.
Regular	User	Low	Both	User info changed for {}
Regular	User	Low	Both	External user {} name was changed to {}.
Regular	User	Low	Both	User info changed for {}
Regular	User	Low	Both	User {} became read only.
Regular	User	Low	Both	Role {} was added to external role {}
Regular	User	Low	Both	External role {} is no longer read only.
Regular	User	Low	Both	External role {} name was changed to {}.
Regular	User	Low	Both	External role {} was added to external user {}
Regular	User	Low	Both	User info changed for {}
Regular	User	Low	Both	External role {} became read only.
Regular	User	Low	Both	Role {} was removed from user {}
Regular	User	Low	Both	External role {} was removed from external user {}
Regular	User	Low	Both	Role {} was added to user {}

Type	Sub System	Severity	Deployment Type	Message
Regular	User	Low	Both	User info changed for {}
Regular	Security	High	Both	Failed to terminate DB session for user {} on {}
Regular	Config	Low	Both	Lookup Data {} was updated.
Regular	Config	Low	Both	Lookup Data {} was created.
Regular	Config	Low	Both	Settings were activated.
Regular	Config	Low	Both	Settings were discarded.
Remove all file classifications failed	File Classification	Low	MX	Remove all file classifications for {} failed.
Remove all file classifications succeeded	File Classification	Low	MX	Remove all file classifications for {} finished successfully.
Report Received from MX	MX	Low	MoM	The Report "{}" was executed on MX "{}", and has been received.
Rollback Package Complete	Software Update	Low	MX	Component {} was rolled back to version {}.
Rollback Package Failed	Software Update	High	MX	Failed to roll back component {} to version {}.
Rollback Package Start	Software Update	Information	MX	Rolling back component {} to version {}.
Send File to Community Defense	Community Defense	Medium	MX	Failed to send file to Community Defense, Response code: {}
Server status	Info	Low	Both	Server started on IP {}
Server status	Info	Low	Both	Server started
Software update is available - new release	Software Update	High	MX	A new version - {} - has been released. Release Content: {} .For further details refer to the Main > Setup > Software Update window.
Software update is available - recommended	Software Update	High	MX	A new {} patch, version {} , is available for {} : {}. Main Content: {}. For further details refer to the Main > Setup > Software Update window.
SOM managed action set was changed on MX	Config	Low	MX	SOM managed Action Set "{}" was locally changed on MX.
SOM to MX SSL certificate is about to expire	Config	High	Both	SOM To MX SSL Certificate is about to expire.
SSL Key expires today	Config	Medium	MX	SSL Key {}>{}>{}>{} expires today.
SSL Key will expire soon	Config	Low	MX	SSL Key {}>{}>{}>{} will expire in {} days.
Stored Procedure group data was retrieved	Config	Low	MX	Stored Procedure group "{}" was populated with data retrieved from DB.
System configuration	Config	Low	Both	System configuration exported

Type	Sub System	Severity	Deployment Type	Message
exported				
System parameter changed	Config	Low	Both	Parameter #mprvMessage($param_name) value changed
System parameter changed	Config	Low	MX	You have reached the allowed limit
System parameter changed	Config	Low	MX	The number of cookies in Application {} (Service {}, Server Group {}) has reached the allowed limit ({}). Consider adding a session identifier or cookie prefixes.
System parameter changed	Config	Low	MX	The number of unique directories in directory {} located in Application {} (Service {}, Server Group {}) has reached the allowed limit ({}). Consider adding URL patterns or contact support.
System parameter changed	Config	Low	MX	The number of unique URLs in the directory {} located on Application {} (Service {}, Server Group {}) has reached the allowed limit ({}). Please add this directory to the ignored list or contact support.
System parameter changed	Config	Low	MX	The number of unique URLs in the application {} located on (Service {}, Server Group {}) has reached the allowed limit ({}). Consider adding application or contact support.
System parameter changed	Config	Low	MX	The number of unique parameters in the application {} located on (Service {}, Server Group {}) has reached the allowed limit ({}). Consider adding application or contact support.
System parameter changed	Config	Low	MX	The number of unique queries in the application {} located on (Service {}, Server Group {}) has reached the allowed limit ({}). Consider adding application or contact support.
System parameter changed	Config	Low	MX	The URL {} in Application {} (Service {}, Server Group {}) has reached the allowed limit of parameters ({}). Consider creating prefixes or extending the parameter limit at the expert page.
System parameter changed	Config	Low	MX	The URL {} in Application {} (Service {}, Server Group {}) has reached the allowed limit of SOAP actions ({}).
System parameter changed	Config	Low	MX	The URL {} in Application {} (Service {}, Server Group {}) has reached the allowed limit of decided allowed methods ({}).
System parameter changed	Config	Low	MX	The URL {} in Application {} (Service {}, Server Group {}) has reached the allowed limit of observed methods ({}).
System parameter changed	Config	Low	MX	The pattern URL {} in Application {} (Service {}, Server Group {}) has reached the allowed limit of observed methods ({}).
System parameter changed	Config	Low	MX	The URL {} in Application {} (Service {}, Server Group {}) has reached the allowed limit of broken reference referrers ({}).
System parameter changed	Config	Low	MX	The pattern URL {} in Application {} (Service {}, Server Group {}) has reached the allowed limit of parameters ({}). Consider creating prefixes or extending the parameter limit at the expert page.

Type	Sub System	Severity	Deployment Type	Message
System parameter changed	Config	Low	MX	The SOAP action {} in Application {} (Service {}, Server Group {}) has reached the allowed limit of parameters ({}).
System parameter changed	Config	Low	MX	The number of learned hosts in the application has reached the allowed limit ({}).
System parameter changed	Config	Low	MX	The number of database users in Application {} (Service {}, Server Group {}) has reached the allowed limit ({}). Please contact support.
System parameter changed	Config	Low	MX	The number of database user groups in Application {} (Service {}, Server Group {}) has reached the allowed limit ({}). Please contact support.
System parameter changed	Config	Low	MX	The number of Query Groups for user {} in Application {} (Service {}, Server Group {}) has reached the allowed limit ({}). Consider extending the parameter limit at the System Definitions.
System parameter changed	Config	Low	MX	The number of source IP addresses for user {} in Application {} (Service {}, Server Group {}) has reached the allowed limit ({}). Consider allowing any IP for this user.
System parameter changed	Config	Low	MX	The number of source applications for user {} in Application {} (Service {}, Server Group {}) has reached the allowed limit ({}). Consider allowing any application for this user.
System parameter changed	Config	Low	MX	The number of source host names for user {} in Application {} (Service {}, Server Group {}) has reached the allowed limit ({}). Consider allowing any host for this user.
System parameter changed	Config	Low	MX	The number of source operating-system usernames for user {} in Application {} (Service {}, Server Group {}) has reached the allowed limit ({}). Consider allowing any OS username for this user.
System parameter changed	Config	Low	MX	The number of databases for user {} in Application {} (Service {}, Server Group {}) has reached the allowed limit ({}). Consider allowing any database for this user.
System parameter changed	Config	Low	MX	The number of queries in Query Group {} for user {} (Application {}, Service {}, Server Group {}) has reach the allowed limit ({}). Switch this query group to dynamic protect.
System parameter changed	Config	Low	MX	Application {} (Service {}, Server Group {}) has reached the allowed limit of login URLs ({}).
Tag was applied	Config	Low	MX	Tag {} was applied to {} {}
Tag was created	Config	Low	MX	Tag {} was created
Tag was deleted	Config	Low	MX	Tag {} was deleted
Tag was removed	Config	Low	MX	Tag {} was removed from {} {}
Tag was renamed	Config	Low	MX	Tag {} was renamed to {}
Update dictionary succeeded	Config	Low	Both	Update {} "{}" succeeded

Type	Sub System	Severity	Deployment Type	Message
Update dynamic group succeeded	Config	Low	Both	Update {} "{}" succeeded
User logged in	User	High	Both	External user {} logged in from {}.
User logged in	User	High	Both	User {} logged in from {}.
User logged out	User	High	Both	User {} logged out. (session expired)
User logged out	User	High	Both	External user {} logged out.
User logged out	User	High	Both	External user {} logged out. (session expired)
User logged out	User	High	Both	User {} logged out.
User session(s) killed by another user	User	Low	Both	Session(s) of User {} killed by user {}.
User was removed from group	File User Rights	Low	MX	User(or group): {} was revoked from group: {}.
Users Repository Update Failed	Config	High	MX	Users Repository "{}" update has failed.
Users Repository Update From CSV File Failed	Config	High	MX	Users Repository "{}" update from CSV file has failed.
Users Repository Updated	Config	Information	MX	Users Repository "{}" was updated successfully.
Users Repository Updated From CSV File	Config	Information	MX	Users Repository "{}" was successfully updated from CSV file.
Vulnerability: change of due date	Vulnerability	Low	MX	Due Date of vulnerability number {} was changed from {} to {}
Vulnerability: change of owner	Vulnerability	Low	MX	Owner of vulnerability number {} was changed from {} to {}
Vulnerability: change of priority	Vulnerability	Low	MX	Priority of vulnerability number {} was changed from {} to {}
Vulnerability: change of status	Vulnerability	Low	MX	Status of vulnerability item {} was changed from {} to {}
Vulnerability: closed	Vulnerability	Low	MX	Vulnerability was Closed
Vulnerability: comment added	Vulnerability	Low	MX	New comment {} was added to vulnerability number {}.
Vulnerability: export	Vulnerability	Low	MX	Export of Vulnerability was done. Action Set {} was executed.
Vulnerability: mitigation	Vulnerability	Low	MX	Mitigation of vulnerability number {} was executed. Status changed from {} to {}
Vulnerability: new	Vulnerability	Low	MX	New vulnerability was created
Vulnerability: new observation	Vulnerability	Low	MX	New observation
Vulnerability: tracking status changed	Vulnerability	Low	MX	Tracking of vulnerability number {} was done. Old value is {}. New value is {}

Type	Sub System	Severity	Deployment Type	Message
WFD Policy set to DEBUG mode	Config	Information	Both	Fraud Detection Policy "{}" was set to debug mode

APPENDIX H

Filter Criteria and Report Data Fields

SecureSphere filter criteria and report data fields provide expanded capabilities when filtering data and generating reports. This chapter provides a reference for the available criteria and data fields and includes the following:

Filter Criteria and Report Data Fields Overview 575
Dataset: Lookup Attribute Usage Example 575
Alert and Violation Criteria 575
Assessment Criteria 577
System Event Criteria 579
Vulnerability Criteria 579
Task Report Criteria 581
Configuration Report Criteria 583
Risk Management Filter Criteria 586
Discovery Report Match Criteria 588
Discovery Filter Criteria 590
Active Module Criteria 592
Profile Criteria 595
Custom Policies Match Criteria 600
Enrichment Policy Match Criteria 607

Filter Criteria and Report Data Fields Overview

SecureSphere utilizes a similar data element for two different features in SecureSphere. These are as follows:

- **Filter Criteria**: Criteria used to configure advanced filters in locations such as the Alert and Audit window.
- **Report Data Fields**: Fields used to configure the Data Scope in reports, which limit the amount of data that appears in reports based on the configuration of these items.

This appendix provides a comprehensive list of these data elements and a description for each.



Note: Criteria that include predefined values such as applications or users only list the first 100 values found. Therefore when looking to configure a filter and not finding a value, you may need to manually type the value if it is not in the list displayed.

Dataset: Lookup Attribute Usage Example

The Dataset: Lookup Attribute criteria can be used to look up attributes from one Lookup Data Set, that are part of another. Let's say you have a network event that contains the **user** field, and you want to audit all users defined in the Lookup Data set Users you created who also belong to the **Blocked Departments** Lookup Data Set.

- The Users Lookup Dataset has 2 columns: User (key column), and Department.
- The Blocked Departments data set has 1 column: Department.

Configure the criteria as follows:

- F**ield**: User (the Field item is an Event field)
- **Lookup Data Set**: Users
- **Attribute**: Department
- (Search in) **Lookup Data Set**: Blocked Departments

So first, the criteria searches the Users Lookup Data Set for a record matching the value that appears in the field "User."

Once found, it takes the department value in that record and searches for a matching department in the Blocked Departments Lookup Data Set.

The criteria filters out events that don't match this logic.

Alert and Violation Criteria

Alert and Violation criteria are available both as part of the Advanced filter in alert and violation monitoring, and as Available Fields when defining the Data Scope for Alert reports.

Alerts: Report Data Scope and Advanced Filter Criteria

Criteria	Description
Alert Flag	Filters data by flag attached to the alert. Options include Important, Acknowledged, Dismissed, and None.
Alert Number	Filters data by number assigned to the alert.
Alert Type	Filters data by type of alert. Options include Firewall, Signature, Protocol, Profile, Worm, Correlation, Custom
Application	Filters data by all applications configured in SecureSphere and to which the logged in user has permission to view.
Custom Policy	Filters data by custom policy by name that has been configured in SecureSphere.
Event ID	Filters data by event ID that generated the alert.
Gateway	Filters by gateway. Enables you to select one or more gateways defined in the system.
Immediate Action	Filters data by immediate action taken as part of the violation that generated the alert: None, Block.
Lookup Data Set	Filters data by Lookup Data set. Select a Type of dataset to search, determine if the operation should be at least one or exclude all. Then manually enter a Lookup Data group value, or choose a specific Lookup Data set.
Object Type	
Protocol	
Server Group	Filters data by all server groups configured in SecureSphere and to which the logged in user has permission to view.
Service	Filters data by all services configured in SecureSphere and to which the logged in user has permission to view.
Session ID	Filters data by session ID that generated the alert.
Severity	Filters data by severity of the alert. Options include Informative, Low, Medium and High.
Source GeoLocation	Filters by the geographic location that was the source of the traffic. **Apply this rule to unidentified locations**: When enabled, includes traffic of IP addresses that aren't assigned to a geographic location. **Apply this rule to an internal network**: When enabled, includes IP addresses if they belong to internal network **Note:** This criteria is only available if you have an active license for ThreatRadar Reputation Services Additionally, if your license expires, source IP data is no longer updated.
Source IP	Filters data by IP address that was the source of the generated alert.
Source of Activity	Filters by the source of the activity. Options include Remote, Local and Unknown.

Criteria	Description
Source GeoLocation	Filters by the geographic location that was the source of the traffic. **Apply this rule to unidentified locations**: When enabled, includes traffic of IP addresses that aren't assigned to a geographic location. **Apply this rule to an internal network**: When enabled, includes IP addresses if they belong to internal network **Note:** This criteria is only available if you have an active license for ThreatRadar Reputation Services Additionally, if your license expires, source IP data is no longer updated.
Time Frame	Filters data based on a time frame with start and finish date in which to search. When searching alerts, finds aggregated alerts in which their last aggregated event took place within this time frame.
URLs	Filters data by a URL (or part thereof) which may have been the source of the alert.
Username	Filters data by user to whom the alert is associated.
Violations	Filters data by type of violation.

Web Based Alert and Violation Criteria

Criteria	Description
Generated by ThreatRadar	Filters data by whether or not it was generated by ThreatRadar.
Generated by ThreatRadar	Filters data by whether or not it was generated by ThreatRadar.
Criteria	**Description**
ThreatRadar Service	Filters by type of ThreatRadar Reputation Service such as Malicious IPs.

Assessment Criteria

Assessment criteria are available both when defining filters in the Assessment Results pane, under Risk Management > Risk Console, and as Available Fields when defining the Data Scope for Assessment reports.

Assessment Filter and Reports Data Scope

Criteria	Description
Asset	Filters data by a specific Site, Server Group, and Service. Enables you to select these items from dropdown lists.
Baseline Job Run	Filters data by baseline jobs of an assessment comparison.
DB Error Message	Filters by database error message.
DB Scan Status	Filters by the status of a database scan.
DB Type	Filters by type of database.
Database Owner	Filters by database owner, as configured through the database owner global object and applied on the database server.
Date	Filters by the date an assessment was conducted. Format is DD/MM/YYYY. You can alternatively select a date by clicking the scheduling icon.
Date - Last Few Days	Filters data by the number of days preceding the current day in which to search.
Direct Access Alias	Filters by alias name used as part of direct access configuration.
Error Message	Filters by a specific error message or part thereof.
IP	Filters data by IP address of the server being assessed.
IP Range	Filters by a range of IP addresses.
Last Scan in Each Policy	Filters by the results of the last scan per policy. Shows only assessment results of the last scan.
OS Error Message	Filters by OS error message.
OS Scan Result	Filters by the result of the OS scan. Options include Error, NotScanned and Scanned.
Policy	Filters by the policy that generated the results. Select a policy name by which to filter.
Policy Name	Filters by policy name. Enables you to manually type a policy name.
Regulation Section Name	Filters by regulation section name.
Regulation Section No.	Filters by regulation section name number.
Result	Filters data by status of the test. Options include Passed, Failed, Error and Irrelevant.
Scan	Filter by predefined scan name.
Scan Name	Filters by scan name.
Service Name	Filters by the name of the service on which the assessment was run.
Service Type	Filters by type of service. Options include.
Severity	Filters data by test severity. Options include Critical, High, Medium, Low and Info.
Sphere	Filters by the Sphere in which the assessment test was run.

Criteria	Description
Test	Filters by a specific test that is part of an assessment.
Test Category	Filters by the category of assessment test.
Test Name	Filters by name of the assessment test.
Test Parameter	Filters by a specific parameter of the assessment test.
Test Type	Filters by the type of assessment test. Options include Connectivity and Regular.
Vulnerability Type	Filters by vulnerability type.
Vulnerability Type CVE	Filters by vulnerability type as determined by its CVE. For more information on CVEs, see Introduction to Common Vulnerabilities and Exposures on page 152.
Vulnerability Type Description	Filters by the description of the vulnerability as it appears in the Assessment Results window under Description. Enables you to match a complete description (equals), a string contained in the description, and more.

System Event Criteria

System criteria can be used as Available Fields when defining the Data Scope for System Event reports.

System Event Report Data Scope

Criteria	Description
Last Few Days	Filters data by the number of days preceding the current day in which to search.
Message	Filters data by message or part thereof which generated as the result of a system event.
Severity	Filters data by event severity. Options include Informative, Low, Medium and High.
Subsystem	Filters data by subsystem in which the event took place.
Time frame	Filters data based on a time frame. Enables you to configure a range of dates within which to search.
User	Filters data by user that generated the system event when user based (for example, with events in which a user that logged in or out).

Vulnerability Criteria

Vulnerability criteria can be used as Available Fields when defining the Data Scope for both Vulnerability reports and List of Vulnerabilities reports. These criteria are also used for filtering vulnerability data in the Vulnerability Management window, as part of Risk Management.

Vulnerability Report Data Scope

Criteria	Description
Age Category	Filters by the period of time since the vulnerability has been discovered.
Asset	Filters by specific asset (Site/Server Group/Service combination).
Asset IP	Filters by a specific asset's IP address.
Asset IP Range	Filters by a range of asset IP addresses.
Asset Location	Filter's by a specific asset's location.
Closure Date	Filters by the date a vulnerability was closed. Enables you to configure a range of dates within which to search.
Creation Date	Filters by vulnerability creation date. Enables you to configure a range of dates within which to search.
Due Date	Filters by vulnerability due date. Enables you to select a range of dates to search.
First Observation Date	Filter by the date vulnerabilities were first observed.
Last Modified Time	Filters by the last time a vulnerability was modified.
Last Observation Date	Filters by the last time a vulnerability was observed.
Lookup Data Set	Filters by value that appears in Lookup Data set. Select the type of item you want to search (the dataset should have this set as its relevance), and determine whether or not the item should appear in that item. Then manually enter a Lookup Data group value, or choose a specific Lookup Data set in which to search.
Mitigatable	Filters by whether or not the vulnerability is mitigatable.
Mitigation Status	Filters by the status of mitigation. Options include Blocked, Manually, Not Blocked, and Not Mitigated.
Number of Observations	Filters by the number of times this vulnerability has been observed on a specific asset. This includes both multiple jobs finding the same vulnerability or a single job finding the vulnerability more than once.
Number of Failed Tests	Filters by the number of tests that have failed the assessment. Each assessment policy contains a number of tests.
Number of Passed Tests	Filters by the number of tests that have passed the assessment. Each assessment policy contains a number of tests.
Number of Test Ended with Errors	Filters by the number of tests in an assessment policy that ended with errors.
Owner Name	Filters by vulnerability owner name.
Parameter Name	Filters by HTTP parameter name.
Priority	Filters by the priority assigned to the vulnerability. Options include Low, Medium and High.
Risk	Filters by level of Risk assigned by SecureSphere to the vulnerability. Risk level can be found by viewing the General tab of a vulnerability in the Vulnerability Workbench.

Criteria	Description
Risk Level	Filters by the level of risk, options include Low, Medium and High.
Service Type	Filters by service type.
Severity	Filters by severity of vulnerabilities. Options include Info, Low, Medium, High and Critical.
Severity Level	Filters by severity of vulnerabilities. Options include Info, Low, Medium and High.
Source Assessment Test	Filters by a specific Source Assessment Test. Enables you to select one or more assessment tests to filter by.
Source Module Name	Name of the source module, or the mechanism in SecureSphere that discovered the vulnerability.
Source Module Type	SecureSphere module through which the vulnerability was discovered. Options include Assessment Test, Assessment Policy, Web Scanner or other module.
Source Policy	Assessment policy that discovered the vulnerability.
Status	Filters by the vulnerability status. Options include Closed, Deleted, Mitigated and Excluded.
Test Name	Filters by assessment test name.
Test Severity	Severity of the vulnerability.
Tracking Flag	Filters by tracking flag. Options include Accepted, Assigned, New, Rejected.
Type	Filters by the type of assessment test.
User	Filters by a specific user.
Vulnerability Number	Number of the vulnerability as assigned by SecureSphere.

Web Based Vulnerability Criteria

Criteria	Description
Cookie Name	Filters by cookie name.
Header Name	Filters by HTTP header name when filtering web vulnerabilities.
Method Name	Filters by HTTP method name when filtering web vulnerabilities.
Parameter Value	
URL	Filters by a specific URL.

Task Report Criteria

Task report criteria are available when configuring reports based on tasks.

Task Report Data Scope

Criteria	Description
Assigned To Me	Filters by tasks assigned to currently logged in user when enabled.
Assignee Changed Followed Action	Filters by changes to followed actions made by the assignee. Enables you to select an assignee who changed a followed action.
Assignment Date	Filters data by date tasks were assigned. Enables you to configure a range of dates within which to search.
Assignment Date - Last Few Days	Filters by date tasks were assigned, in the last few days.
Closure Date	Filters by date tasks were closed. Enables you to configure a range of dates within which to search.
Closure Date - Last Few Days	Filters by date tasks were closed, in the last few days.
Comment Author	Filters by author of comments assigned to tasks.
Comment Date	Filters data by date comments were added. Enables you to configure a range of dates within which to search.
Comment Date - Last Few Days	Filters by date comments were added, in the last few days.
Comment Text	Filter by comment text, or a part thereof.
Creation Date	Filters by the date tasks were created.
Creation Date - Last Few Days	Filters by task creation date. Enables you to configure a range of dates within which to search.
Due Date	Filters by the date a task is due.
Due for Today	Filters by task that are due today.
Due in	Filters by period of time in which tasks are due. Enables you to select a period of time from the dropdown list.
Last Update Date	Filters by the last date tasks were updated.
Last Update Date - Last Few Days	Filters by the last date tasks were updated. Enables you to configure a range of dates within which to search.
Notify Owner	Filters by whether or not the owner was notified.
Overdue Followed Action	Filters by those tasks with overdue followed actions.
Owned by Me	Filters by tasks owned by the currently logged in user when enabled.
Priority	Filters by task priority. Options include High, Low, Medium and Urgent.
Reviewer	Filters by task reviewer. Enables you to select reviewer by which to filter data.
Status	Filters by task status. Options include Closed, In Progress, New and Rejected.

Criteria	Description
Status Changed Followed Action	Filters by status of a followed action that has been changed.
Subsystem	Filters by subsystem.
Task Assignee	Filters by task assignee. Enables you to select an assignee by which to filter data.
Task Name	Filters by task name.
Task Owner	Filters by task owner. Enables you to select an owner by which to filter data.
Task Type	Filters by task type. Options include Assignment and Review.

Configuration Report Criteria

Configuration Report criteria can be used as Available Fields when defining the Data Scope for Configuration reports. The following are the available types of configuration reports and their criteria:

- Configuration - Sites Report Criteria on page 584
- Configuration Security Policy Report Criteria on page 585

Configuration - Sites Report Criteria

Site Configuration Report Data Scope

Criteria	Description
Application	Filters by application configured on the site. Enables you to select one or more applications configured in SecureSphere.
Application Creation Time	Filters by the date an application was created. Enables you to configure a range of dates within which to search.
Is Server Discovered	Filters by whether or not the server has been discovered as part of a discovery policy.
Is Server Protected	Filters by whether or not the server is actively protected by SecureSphere.
Protecting Gateway Group	Gateway group to which the site is assigned.
Server Group	Filters by server group configured on the site. Enables you to select one or more server groups configured in SecureSphere.
Server Group Creation Time	Filters by the date a server group was created. Enables you to configure a range of dates within which to search.
Server Group Mode	Filters based on the current server group mode. Options include Active, Simulation, and Disabled.
Server Host Name	Filters by server host name.
Server IP	Filters by server IP address.
Server IP Creation Time	Filters by the date a server IP address was created.
Server IP Range	Filters by range of server IP addresses.
Server OS Type	Filters by server OS type. Options include Windows, Linux, Solaris, HP-UX, and AIX.
Service	Filters by service. Enables you to select one or more services configured in SecureSphere.
Service Creation Time	Filters by the date a service was created.
Service Ports	Filters by ports running on the service.
Service SSL Ports	Filters by SSL ports running on the service.
Service Type	Filters by services configured on the site. Enables you to select one or more services groups configured in SecureSphere.
Site	Filters by site. Enables you to select one or more sites configured in SecureSphere.
Site Creation Time	Filters by the date a site was created.

Configuration Security Policy Report Criteria

Configuration Security Policy Report Data Scope

Criteria	Description
ADC Content	Filters by whether or not this policy was defined in ADC content.
ADC Keywords	Filters by ADC Keyword. Options include all available keywords pre-defined by SecureSphere ADC.
Applied by Default	Filters by whether or not this policy is applied by default on a new asset.
Applied to Assets	Filters by a specific asset to which the policy is applied. Enables you to configure a specific Site, Server Group, Service and Application.
Display Web Page Response	Filters by whether or not the security policy displays a web page response.
Policy Creation Date	Filters by the date the security policy was created.
Policy Creation Date - Last Few Days	Filters by the number of days preceding the current day that the security policy was created.
Policy Last Update Date	Filters by the number of days since the security policy was last updated.
Policy Last Update Date - Last Few Days	Filters by the number of days preceding the current day since the security policy was last updated.
Policy Level	Filters by the policy level. Options include Web, Web Service, Web Application, and Network.
Policy Name	Filters by the name of a policy.
Policy Type	Filters by the type of security policy. Enables you to select an existing security policy type.
Rule Action	Filters by action taken as the result of a rule.
Rule Enabled	Filters by whether or not a rule is enabled.
Rule Followed Action	Filters by a specific followed action which has been configured to run as part of a security policy.
Rule Has Exception	Filters by whether or not the rule has one or more exceptions configured.
Rule Name	Filters by rule name.
Rule Severity	Filters by the severity of a rule. Options include No Alert, Informative, Low, Medium, and High.
Vulnerability Mitigation	Filters by whether or not a vulnerability is mitigated.

Risk Management Filter Criteria

Risk Management filter criteria can be used as Available Fields when filtering information in the Risk Management window.

- Vulnerability Management Filter Criteria on page 586
- System Event Criteria on page 579

You can use the Basic and Advanced filters to further refine Vulnerability information that is displayed. The Basic Filter enables you to select the desired search criteria and apply it to display data containing specific attributes. For information on how to work with Advanced filter criteria, see Using the Advanced Filter on page 112.

Vulnerability Management Filter Criteria

For a list of Vulnerability Management Filter Criteria, see Vulnerability Criteria on page 579.

Assessment Results Filter Criteria

Assessment Results filter criteria can be used as Available Fields when wanting to filter information in the Risk Management > Assessment Results window.

Assessment Results Filter Criteria

Criteria	Description
Asset	
Baseline Job Run	Filters data by baseline jobs of an assessment comparison.
Date	Filters by the date an assessment was conducted. Format is DD/MM/YYYY. You can alternatively select a date by clicking the scheduling icon.
Date - Last Few Days	Filters data by the number of days preceding the current day in which to search.
Direct Access Alias	Filters by alias name used as part of direct access configuration.
Error Message	Filters by a specific error message or part thereof.
IP	Filters data by IP address of the server being assessed.
IP Range	Filters by a range of IP addresses.
Last Scan in Each Policy	Filters by the results of the last scan per policy. Shows only assessment results of the last scan.
OS Error Message	Filters by OS error message.
OS Scan Result	Filters by the result of the OS scan. Options include Error, Not Scanned and Scanned.
Policy	Filters by the policy that generated the results. Select a policy name by which to filter.
Policy Name	Filters by policy name. Enables you to manually type a policy name.
Regulation Section Name	Filters by regulation section name.
Regulation Section No.	Filters by regulation section name number.
Result	Filters data by status of the test. Options include Passed, Failed, Error and Irrelevant.
Scan Name	Filters by scan name.
Service Name	Filters by the name of the service on which the assessment was run.
Service Type	Filters by type of service. Options include.
Severity	Filters data by test severity. Options include Critical, High, Medium, Low and Info.
Sphere	Filters by the Sphere in which the assessment test was conducted.
Test Category	Filters by the category of assessment test.
Test Name	Filters by name of the assessment test.
Test Parameter	Filters by a specific parameter of the assessment test.
Test Type	Filters by the type of assessment test. Options include Connectivity and Regular.
Vulnerability Type	Filters by vulnerability type.

Criteria	Description
Vulnerability Type CVE	Filters by vulnerability type as determined by its CVE. For more information on CVEs, see Introduction to Common Vulnerabilities and Exposures on page 152.
Vulnerability Type Description	Filters by the description of the vulnerability as it appears in the Assessment Results window under Description. Enables you to match a complete description (equals), a string contained in the description, and more.

Discovery Report Match Criteria

Active Module Server Discovery criteria can be used as Available Fields when defining the Data Scope for Server Discovery reports.

Criteria are available for the following types of discovery reports:

Service Discovery Execution Report on page 590

Service Discovery Detailed Report

Service Discovery Detailed Reports Match Criteria

Criteria	Description
Asset	Filters data by specific asset (Site/Server Group/Service combination).
Data Status	Filters data by data status (whether or not data was found on service). Options include Found, Not Found and Sensitive.
Database Owner	Filters by the data owner assigned to the file detected.
DB Credentials	Filters data by whether or not database credentials were detected. Options include Found, Not Found and Not Scanned.
Decision	Filters data by decision regarding discovered service according to its status. Options include Existing in Configuration, Pending, Accepted, Auto Accepted, and Reject.
First Discovery Date	Filters data by date services were first discovered.
Host	Filters data by hostname.
IP	Filters data by IP address.
IP Range	Filters by a range of IP addresses.
Last Discovery Date	Filters data by last time services were discovered.
Last Scan Only	Filters data by last service discovery.
Lookup Data Set	Filters by value that appears in Lookup Data set. Select the type of item you want to search (the dataset should have this set as its relevance), and determine whether or not the item should appear in that item. Then manually enter a Lookup Data group value, or choose a specific Lookup Data set in which to search.
OS Name	Filters data by OS Name.
OS Type	Filters by OS Type.
OS Vendor	Filters by OS Vendor.
OS Version	Filters by OS Version.
OS Credentials	Filters by whether or not OS credentials were detected. Options include Found, Not Found and Not Scanned.
Port	Filters by port number.
Port Name	Filters by port name.
Port Version	Filters by port version.
Scan Name	Filter by the name of a scan.
Service Status	Filters by the status of services. Options include Existing and New.
Service Type	Filters data by service type. Enables you to select from all of the available services supported by SecureSphere.

Service Discovery Execution Report

Service Discovery Execution Reports Match Criteria

Criteria	Description
Execution Date	Filters data by date service discovery was conducted. Enables you to configure a range of dates within which to search.
Number of Existing Servers Found	Filters data by number of existing servers found.
Number of Existing Services Found	Filters data by number of existing services found.
Number of New Servers Found	Filters data by number of new servers found.
Number of New Services Found	Filters data by number of new services found.
Number of Servers Found	Filters data by total number of servers found.
Number of Services Accepted	Filters data by number of services accepted.
Number of Services Auto Accepted	Filters data by number of services that were automatically accepted.
Number of Services Found	Filters data by total number of services found.
Number of Services Rejected	Filters data by number of services rejected.
Policy	Filters data by policy name.
Site	Filters data by site.

Discovery Filter Criteria

You can use the Basic and Advanced filters to further refine Discovery information that is displayed. The Basic Filter enables you to select the desired search criteria and apply it to display data containing specific attributes. The following table lists the available discovery filter criteria.

For information on how to work with Advanced filter criteria, see Using the Advanced Filter on page 112.

Discovered Servers Filter Criteria

Criteria	Description
Data Status	Filters data by data status. Options include Not found, Found and Found Sensitive.
DB Owner	Filters by database owner, as configured through the database owner global object and applied on the database server.
DB Credentials	Filters data by whether or not database credentials were found. Options include Found, Not Found, and not Scanned.
Decision	Filters data by the decision on what action to take. Options include Pending, Added by User, Added Automatically, and Rejected by User.
Host	Filters data by host name.
IP	Filters data by IP address.
OS Credentials	Filters data by whether or not Operating System credentials were found. Options include Found, Not Found, and not Scanned.
Port	Filters data by port number
Scan Name	Filters data by discovery scan name. Enables you to select one or more discovery scans on which to base the filter.
Service Status	Filters data by service status. Options include New and Existing.
Service Type	Filters data by service type. Enables you to select one or more service types.

Classified DB Data Filter Criteria

Criteria	Description
Database Name	Filters data by database name.
Database Owner	Filters by database owner, as configured through the database owner global object and applied on the database server.
Decision	Filters data by action taken on discovered data table. Options include Accepted by User, Accepted Automatically, Existing in Configuration, and Pending.
Port	Filters data by port number
Scan Name	Filters data by discovery scan name. Enables you to select one or more discovery scans on which to base the filter.
Schema Name	Filters data by schema name.
Sensitive Data	Filters data based on whether or not it is marked as sensitive.
Service Decision	Filters data by service decision. Options include Pending, Added by User Added Automatically, and Rejected by User.
Service Status	Filters data by service status. Options include New and Existing.
Service Type	Filters data by service type. Enables you to select one or more service types.
Table Group	Filters data by table groups. Enables you to type a table group on which to base the filter.
Table Group Status	Filters data by table group status. Options include New and Existing.

Criteria	Description
Table Name	Filters data by table name.
Table Status	Filters data by database table status. Options include New and Existing.

Active Module Criteria

Active Module criteria can be used as Available Fields when defining the Data Scope for Active Module reports. SecureSphere has a number of Active Module reports, each with its own Available Fields. These include the following:

- Active Module - Change Tracking Details on page 592
- Active Module Pre-configured Objects on page 593
- Active Module Track Value Changes on page 594
- Active Module Track Value Changes Configuration on page 594

Active Module - Change Tracking Details

Active Module Change Tracking Details criteria can be used as Available Fields when defining the Data Scope for Change Tracking Details reports.

Active Module: Report Change Tracking Details Data Scope

Criteria	Description
Change in Past X days	Filters data by changes made in the last X days.
IP Address	Filters data by target IP address.
Modification	Filters data by the type of modification that was detected.
Object	Filters data by the object that has been modified.
Object List	This is the list of objects to which the object belongs. For example, Oracle Configuration Files, Sybase executable files, and so on.
Object Type	Filters data by the type of object that has been modified. Available type is file.
Policy	Filters data by policy name.
Server Group	Filters data by all server groups configured in SecureSphere and to which the logged in user has permission to view.
Site	Filters data by all sites configured in SecureSphere and to which the logged in user has permission to view.
Size	Filters data by size of file.
Time Frame	Filters data based on a time frame. Enables you to configure a range of dates within which to search.
Time of Day	Filters data by time of day (hours), and by days of the week based on the current gateway's local timezone. Data is included for times that are highlighted. You can click a heading to mark an entire day or hour.

Active Module Pre-configured Objects

Active Module Predefined Configuration criteria can be used as Available Fields when defining the Data Scope for Preconfigured Objects reports.

Active Module: Report Change Tracking Preconfigured Objects Data Scope

Criteria	Description
Object List	Filters data by name of the predefined object list.
Object Type	Filters data by the type of object that was modified. Available type is file.
Operating System	Filters data by the type of operating system to which the objects are relevant.
Pattern	Filters data by name of the object or the object pattern.
Report Missing	Filters data by whether or not the object should be reported if missing.

Active Module Track Value Changes

Active Module Track Value Changes criteria can be used as Available Fields when defining the Data Scope for Track Value Changes reports.

Active Module: Report Track Value Changes Data Scope

Criteria	Description
After Value	Filters data by using string entered as the updated value of item that was changed.
Before Value	Filters data by using string entered as the original value of item that was changed.
Column Name	Filters data by column name.
DB Session ID	Filters data by Session ID.
DB Owner User Name	Filters data by schema name where the logged table resides.
Database User	Filters data by Database user.
Hour	Filters data by hour.
Last Few Days	The number of days which you want to report on since today.
Operation	Filters data by database operation.
Percentage Change	Filters data by the percentage of change. Relevant only for numeric values.
Policy	Filters data by policy name.
Row Unique Identifier	Filters data by internal ID of the row.
Serial Number	Filter's data according to the session's serial number.
Server IP	Filters data by server IP address.
Source Application	Filters data by source application.
Source OS Host Name	Filters data by source OS host name.
Source OS User Name	Filters data by source OS user name.
Table Name	Filters data by table name.
Time Frame	Filters data based on a time frame. Enables you to configure a range of dates within which to search.

Active Module Track Value Changes Configuration

Active Module Track Value Changes Configuration criteria can be used as Available Fields when defining the Data Scope for Track Value Changes Configuration reports.

Active Module: Report Track Value Changes Configuration Data Scope

Criteria	Description
Column Name	Filters data by column name.
Column Type	Filters data by column type.
DB User Owner	Filters by database owner, as configured through the database owner global object and applied on the database server.
Policy	Filters data by policy name.
Server IP	Filters data by server IP address.
Table Name	Filters data by table name.

Profile Criteria

Profile criteria can be used as Available Fields when defining the Data Scope for Profile reports. SecureSphere has a number of Profile reports, each with its own available fields. These include the following:

- Web Profile URL Report on page 595
- Web Profile Cookies Report on page 597
- Web Profile - Learned Hosts Report on page 598
- Web Profile - Susceptible Directories Report on page 599
- Web Profile - Web Application User Tracking Report on page 599

Web Profile URL Report

Web Profile URL criteria can be used as Available Fields when defining the Data Scope for Web Profile URL reports.

Report: Web Profile URL - Criteria

Criteria	Description
Application	Filters data by application.
Correlated Query	Filters data by correlated query.
Correlated Query Application	Filters data by correlated query application.
Correlated Query Service	Filters data by correlated query service.
Correlated Query User	Filters data by correlated query user.
File Type	Filters data by file type.
Folder	Folder in which the page is located
Learning State	Filters data by SecureSphere learning (learning, in protection).
Learning State Switch Date	Filters data by last date the URL learning state was changed.
Learning State Switch Date Last Few Days	Filters data by number of days preceding the current day.
Link Status	Filters data by link status. Options include Valid, Broken Link, Broken Reference.
Locked	Filters data by locked status (whether or not the profile is still learning).
Method	Filters data by HTTP method.
Number of Correlated Queries	Filters data by number of correlated queries in the URL.
Number of Parameters	Filters data by number of parameters used by the URL.
Occurrences	Filters data by number of occurrences before this URL was learned.
Parameter Max Length	Filters data by maximum length of a parameter in the URL.
Parameter Min Length	Filters data by minimum length of a parameter in the URL.
Parameter Name	Filters by parameter name.
Parameter Prefix	Filters data by parameter prefix.
Parameter Read Only	Filters data by read only status (if parameter is read only).
Parameter Required	Filters data by required status (if parameter is required).
Referer	Filters data by URL that was referred as the result of a broken link.
Response Time	Filters data by response time.

Criteria	Description
SOAP	Filters data by Soap parameter.
SOAP Action	Filters data by Soap action.
SOAP Parameter	Filters data by Soap parameter.
SOAP Parameter Max Length	Filters data by Soap parameter maximum length.
SOAP Parameter Min Length	Filters data by Soap parameter minimum length.
SOAP Parameter Required	Filters data by Soap parameter required status (if Soap parameter is required).
Server Group	Filters data by all server groups configured in SecureSphere and to which the logged in user has permission to view.
Service	Filters data by all services configured in SecureSphere and to which the logged in user has permission to view.
Site	Filters data by all sites configured in SecureSphere and to which the logged in user has permission to view.
URL	Filters data by URL.
URL Creation Date	Filters data by creation date.
URL Creation Date - Last Few Days	Filters data by the number of days preceding the current day.
URL Update Date	Filters data by date URL was updated.
URL Update Date - Last Few Days	Filters data by the number of days preceding the current day on which an update occurred.

Web Profile Cookies Report

Web Profile Cookie criteria can be used as Available Fields when defining the Data Scope for Web Profile Cookies reports.

Web Profile Cookies - Criteria

Criteria	Description
Application	Filters data by application.
Cookie Creation Date	Filters data by cookie creation date.
Cookie Creation Date - Last Few Days	The number of days preceding the current day in which to search for cookies that have been created.
Cookie Injection	Filters data by whether cookie was injected.
Cookie Name	Filters data by cookie name
Cookie Prefix	Filters data by cookie prefix.
Cookie Update Date	Filters data by date cookie was updated.
Cookie Update Date - Last Few Days	Filters data by last date cookie was updated, in the past few days.
Server Group	Filters data by server group.
Service	Filters data by service.
Site	Filters data by site.
Status	Filters data by cookie status (learn, protect, ignore).

Web Profile - Learned Hosts Report

Web Profile – Learned Hosts criteria can be used as Available Fields when defining the Data Scope for Web Profile – Learned Hosts reports.

Web Profile Learned Hosts - Criteria

Criteria	Description
Application	Filters data by application.
Host Name	Filters data by host name.
Server Group	Filters data by all server groups configured in SecureSphere and to which the logged in user has permission to view.
Service	Filters data by all services configured in SecureSphere and to which the logged in user has permission to view.
Site	Filters data by all sites configured in SecureSphere and to which the logged in user has permission to view.

Web Profile - Susceptible Directories Report

Web Profile – Susceptible Directories criteria can be used as Available Fields when defining the Data Scope for Web Profile – Susceptible Directories reports.

Web Profile Susceptible Directories - Criteria

Criteria	Description
Application	Filters data by application.
Directory	Filters data by directory name.
Directory Creation Date	Filters data by directory creation date.
Directory Creation Date - Last Few Days	Filters data by date the directory was created, in the last few days.
Directory Status	Filters data by status of the directory (learning, protection).
Directory Update Date	Filters data by date directory was last updated.
Directory Update Date - Last Few Days	Filters data by date this directory was last updated, in the last few days.
Server Group	Filters data by all server groups configured in SecureSphere and to which the logged in user has permission to view.
Service	Filters data by all services configured in SecureSphere and to which the logged in user has permission to view.
Site	Filters data by all site configured in SecureSphere and to which the logged in user has permission to view.

Web Profile - Web Application User Tracking Report

Web Profile – Web Application User Tracking criteria can be used as Available Fields when defining the Data Scope for Web Profile – Web Application User Tracking reports.

Web Profile Web Application User Tracking - Criteria

Criteria	Description
Action URL	Filters by action URL, used to login to the web application.
Action URL Field 1	Filters data by the first field of the login / user identification fields. For example, User Name.
Action URL Field 2	Filters data by the second field, if required, of the user identification fields. For example, Password.
Action URL Creation Date	Filters by date action url was created.
Action URL Creation Date - Last Few Days	Filters by date action URL was created, in the last few days.
Action URL Update Date	Filters by last date action URL was updated.
Action URL Update Date - Last Few Days	Filters by last date action URL was updated, in the last few days
Application	Filters data by all applications configured in SecureSphere and to which the logged in user has permission to view.
Decision Rule Type	
Default Decision	Filters data by the default decision. Options include Succeeded Failed, or Can't Tell.
Login Result	Filters by login result. If user login succeeded, failed or can't tell.
Login URL	Filters by login URL used to reach action URL.
Pattern Value	Filters by pattern value defined in the active module.
Redirect URL Value	Filters data by the URL the user is redirected to after attempting to login
Response Code Value	Filters data by the response code returned after the user attempts to login (the redirect URL and response code vary according to the login result.
Server Group	Filters data by all server groups configured in SecureSphere and to which the logged in user has permission to view.
Service	Filters data by all services configured in SecureSphere and to which the logged in user has permission to view.
Site	Filters data by all sites configured in SecureSphere and to which the logged in user has permission to view.
Status	Filters data by status of the User. Options include learning, in protection or manual.

Custom Policies Match Criteria

This section describes the policies match criteria used to define custom policy rules.



Note: When specifying more than one parameter, the ratio between them is OR.

Match Criteria are available for the following types of policies:

Web Custom Policies Match Criteria 601

Web Custom Policies Match Criteria

The following table presents the web custom policy match criteria.

Web Custom Policies Match Criteria

Criteria	Description
Application User	Match on the applicative user name, which is used to connect to the application. This enables you to identify the end user who initiated the database transaction. **Note:** This match criteria is effective only when UUT is defined, see Understanding Universal User Tracking on page 285. When the **Match Unknown Value** parameter is enabled and the value does not exist, there is a match.
Authenticated Session **Authentication Result** **Authentication URL**	All the authentication processes are defined in the profile. The authentication match criteria is used as an interface to the Application User Tracking engine, see Web Application User Tracking on page 287, as follows: • **Authentication Session**: Enables verification of session authentication. **Yes** means that SecureSphere checks that session authentication is done exactly as it is defined in the profile. Default: No. • **Authentication Result**: Match on the login results of the application URL. **Successful** means the user is authenticated according to the rules configured in the profile AUT. **Failed** means the login is failed. **Note:** The Authentication Result match criteria runs on the response from the server. Do not configure the **Blocking** action, as **Blocking** cannot be executed when working with response. • **Authentication URL**: The URLs that are used for the authentication process and that are specified in the Action URL list in the profile. **Yes** means that the URL is from the Action URL list and **No** means that the URL is not included in that list. Default: No.
Data Set: Attribute Lookup	Filters data by an attribute in a Lookup Data Set. This criteria is two phased. It first searches for the event Field selected from the dropdown menu in the key column of the Lookup Data Set, selected from the Lookup Data Set dropdown menu. Then, in the results from the first phase above, and for each result; searches for the value of the column defined by the Attribute field in the User Defined Values table or in the Data Sets configured in the Lookup Data Set table. For an example on how to use this criteria, see Dataset: Lookup Attribute Usage Example on page 575.
Enrichment Data	Filters data by Lookup Data set. Select a Field to search (the dataset should have this set as its relevance), determine if the operation should be at least one or exclude all. Then manually enter a Lookup Data group value, or choose a specific Lookup Data set.

Criteria	Description
Fraud Prevention Results	Match on correlation of parameters as described in the criteria. For more information on using this criteria in a policy, see ThreatRadar Fraud Prevention Services on page 206. **Provider**: Determines the provider to use for detecting web fraud. If Information is Missing: Determines what to do if any part of information from your Web Fraud Prevention service provider is missing. For example, the required tag may not have been received when connecting to the provider, or SecureSphere may be having problems connecting to the provider's servers. Choices include **Apply Policy** and **Do Not Apply Policy**. Operations: If using more than one tag, determines if all tags and their values need to be matched or the policy to be applied, or if only one tag and its values matching is sufficient. Tag Name: Lists the name of a tag used by your Web Fraud Prevention provider. For example, the Tag name might be "Malware Detected." Operation: Determines if the policy should work if the tag name equals, or does not equal the Value. Value: Represents the actual value of the tag. for example your Web Fraud Prevention service provider and may have values like True (meaning that malware has been detected), or False (no malware was detected).
Generic Dictionary Search	Match on specific values in responses. Define the following parameters: • **Search Mode**: Define Exact to search for the exact pattern, or Contains to search for the pattern that includes the pattern that you want. • **Dictionaries**: Select the generic dictionaries that are defined in the Global Objects, see Generic Dictionary Groups on page 361. • **Location**: URL- Request URLs. Parameters - Parameters in the request. Headers - HTTP headers. Response Content - content returned by the web server. Non Normalized URL - the URL row, as it is received in the actual request.

Criteria	Description
HTTP Request	Match on the appearance of a pattern or a substring in an HTTP request header, parameter, or URL, or a combination of these items. Can be used to identify offending characters or patterns you want to blacklist. • **Operation**: Matches Part, Name, Match Operation and Value as selected. Options include Match All, Match Any, or Match None of the items. • Part: Determines the part of the traffic you want to track. Options include: • **Header**: Matches on the header name and value, for example a cookie header. Can use a string or signature as the value. • **Parameter**: Matches on the parameter name and value. Can search the parameter value for a character such as a single quote ('). • **URL**: Matches on a specific URL. Note that in this case there is no part name as there is only one URL in the request. When using this option, it is recommended to add the HTTP Request URL match criteria to focus the search and make it more effective. • **Name**: Name of the specific part (header or parameter). The name option is not available when choosing to use a part of type URL. • **Match Operation**: The means by which to match traffic. Options include: • **Includes**: The part (URL, header or parameter) includes the string specified in the Value field • **Does Not Include**: The part does not include the string specified in the Value field • **Matches Regular Expression**: The part matches the regular expression specified in the Value field • **Does Not Match Regular Expression**: The part does not match the regular expression specified in the Value field **Note:** For more information on Regular Expressions, see Writing Signatures and Regular Expressions on page 473. • **Value**: The actual value you are looking to match. **Note:** Using this criteria blacklists an item under specific circumstances. While using SecureSphere Web Profile blacklists the item under all circumstances. **Note:** This criteria is commonly used by the Web Scanner Integration module.
HTTP Request Accept-Language (Header)	Match on the language that is used by the server in response. The definition appears in the request header field and it restricts the set of natural languages that the sender of the request prefers to see in the response. If the language in the response is different from the languages defined by this criteria, SecureSphere alerts you. For example: • Operations: **Exclude all** • Value: **en** • Value: **en-us** So if the header that is sent by the client includes **de-de** (German for Germany), then SecureSphere alerts you that the preferred language in response should be other than English.
HTTP Request Content-Type (Header)	Filters by Content Type prefix from the request header. This parameter is case sensitive. To set user-defined request headers, use the Headers match criteria.

Criteria	Description
HTTP Request Cookie Name	Match on cookie names found in client requests. The match is done on cookie names only. For match on cookie value, use the Request Cookie match criteria. For each cookie name, specify **Match Type**: whether you want to search for the exact cookie name or just the prefix.
HTTP Request Cookies	As an extension to the HTTP Request Cookie Name match criteria this match criteria allows you to perform match on the cookie value in addition to cookie names, as follows: • **Name**: Type the name of the cookie as it appears on your web site. • **Match Type**: Select if you want to search for the full value or just for the prefix. • **Value**: Type the cookie value.
HTTP Request File Extension	Filters by file type. For each type add a new row in the table and type the file extension, for example, `.gif` **Note:** If Case Sensitive is enabled in the service configuration, then the **Value** that you define for **File Type** is also case sensitive, see Configuring Advanced Web Service Settings on page 424.
HTTP Request Header Name	Match on request HTTP headers, even if there are customized headers. For match on response header, use the Response Header match criteria. Define the **Value** using a full header name or just header prefix. This parameter is case sensitive.
HTTP Request Header Value	Match on header names in requests. SecureSphere differentiates between the header names in requests and responses. You can match on At Least One header name, Or Exclude All.
HTTP Request Host Name (Header)	Match on particular Host Names. Define the **Value** using a full name. This parameter is case sensitive.
HTTP Request Method	Match on the HTTP methods. This match criteria can be customized to use proprietary methods. **Note:** If you want to use proprietary methods, you must update the HTTP Protocol validation policy. The Unknown HTTP Request Method rule in the policy should be customized to include the new proprietary methods.
HTTP Request Parameter Name	Match on HTTP parameter names.
HTTP Request Referer Hostname (Header) **Referer URL Prefix (Headers)** **HTTP Request Content-Type (Headers)** **HTTP Request User-Agent (Header)**	Match on specific, predefined, commonly used request headers. Type the header value. This parameter is case sensitive. To set user-defined request headers, use the Headers match criteria. Filters by User-Agent request header.
HTTP Request URL	Match on URLs using URL prefixes or Exact URLs.
HTTP Response Header Name	Match on header names in responses. SecureSphere differentiates between the header names in requests and responses. You can match on At Least One header name, Or Exclude All.

Criteria	Description
HTTP Response Status Code	Match on the response code that returns from the server. You can type the user-defined response code, Or select a predefined value.
HTTP Session	SecureSphere can track HTTP client-server sessions according to session cookies. In a typical HTTP session, the web server initiates the session by setting a cookie that the client uses and that the server updates as the session progresses. The Session match criteria can match on any of 3 statuses SecureSphere assigns to sessions that go through it, as follows: • **Validated**: The session is validated by SecureSphere according to the session cookie received from the server. A minimum of one verified cookie is enough to consider the session as validated even if the session is based on more than one cookie. • **Invalidated**: The session is invalidated if no session cookie was received from the server. • **No Session**: SecureSphere does not detect any session cookie used by the client.
Lookup Data Set Search	Filters data by Lookup Data set. Select a Field to search (the dataset should have this set as its relevance), determine if the operation should be at least one or exclude all. Then manually enter a Lookup Data group value, or choose a specific Lookup Data set.
Number of Occurrences	Match over time. This match criteria is usually used in combination with other criteria. The values of the **In the context of a single** parameter are used as follows: • Server Group: To define the destination. • Originating Session/Source IP/User: To define the source.
Profiled Referer Host	Match on the learned host names that are included in the Learned Hosts list in the profile, see Web Profile Policy. "Web Profile Policy" on page 275 SecureSphere takes the referer header value from the request and checks if the value appears in the Learned Hosts list in the profile. Using this match criteria you can differentiate between the external referers and internal referers.
Protocols	Match on the web protocol: HTTP/HTTPS. Using this match criteria you can differentiate between HTTP to HTTPS. You can select HTTP, HTTPS, or both.
Proxy IP Addresses	Match on the proxy IP addresses in configurations where clients' Source IP addresses are hidden by the proxy. The Source IP addresses are contained in one of the predefined proxy lists that SecureSphere maintains or are user-defined. Select the group of IP addresses defined in Global Objects, or define the IP address. When the **Match Unknown Value** parameter is enabled and the value does not exist, there is a match.
Sensitive Dictionary Search	Enables you to search for specific values in responses, see Configuring Sensitive Data Protection on page 350. For an example of a custom policy with this match criteria refer to Preventing Data Leakage in Web Responses xxx.
Signatures	Match on the list of Signatures. The signatures used in this rule must be included in a dictionary that is applied to the relevant service.

Criteria	Description
Source GeoLocation	Filters by the geographic location that was the source of the traffic. **Apply this rule to unidentified locations**: When enabled, includes traffic of IP addresses that aren't assigned to a geographic location. **Note:** This criteria is only available if you have an active license for ThreatRadar Reputation Services Additionally, if your license expires, source IP data is no longer updated.
Source IP Addresses	Match on the Source IP addresses when you want to limit the search to a specific IP range, as follows: • Operations: At Least One/Exclude All. • IP groups: The IP groups defined in the Global Objects, see Configuring IP Groups on page 347. • User Defined: Type the Source IP address that you want.
Time of Day	Enables you to define a period of time in which you want to activate your custom rule based on the current gateway's local timezone. You can either include or exclude certain periods time, allowing you to force various rules in different times of the day. Define the period of time that you want by highlighting the squares. The purple squares stand for the period of time in which the rule is enforced.
Violations	Match on all the violations that appear in the policy rules. Using this match criteria, you can define custom rules that correlate between different policy types and different layers. The ability to correlate between the rules from different policies enables you to define more sophisticated and accurate policies. **Note:** To prevent alerts from appearing twice, from the original policy and from the custom policy, it is recommended to define in the custom policies using the Violations match criteria the **Severity** parameter as **No Alert**. for more information on alerts, see Understanding Alerts, Events and Violations on page 80.
Web Page Response Size	Match on the size of web page responses defined in bytes. For example, a page that is longer than the value defined here might indicate an anomaly.
Web Page Response Time	Match on the period of time that is required for web page response. This match criteria provides you with the ability to identify and alert abnormal activity based on web server response time. The time is presented in milliseconds.

Enrichment Policy Match Criteria

SecureSphere enrichment enables you to enrich event data with information manually entered or imported from an external system. There are two types of enrichment policy match criteria:

- Web Service Enrichment Policy Match Criteria on page 607

Web Service Enrichment Policy Match Criteria

This section describes the match criteria used to define enrichment policies.

Web Service Enrichment Match Criteria

Criteria	Description
Application User	Filters by users according to AUT, see Web Application User Tracking on page 287.
Data Set: Attribute Lookup	Filters data by an attribute in a Lookup Data Set. This criteria is two phased. It first searches for the event Field selected from the dropdown menu in the key column of the Lookup Data Set, selected from the Lookup Data Set dropdown menu. Then, in the results from the first phase above, and for each result; searches for the value of the column defined by the Attribute field in the User Defined Values table or in the Data Sets configured in the Lookup Data Set table. For an example on how to use this criteria, see Dataset: Lookup Attribute Usage Example on page 575
Enrichment Data	Filters data by a field configured in enrichment data created by an enrichment rule.
Generic Dictionary Search	Filters by generic dictionaries. Enables you to search numeric values: which converts the database numeric fields to strings for matching with the dictionary group patterns. You can search in Queries or Parsed Queries.
HTTP Request	
HTTP Request Content-Type (Header)	Filters by Content Type prefix from the request header. This parameter is case sensitive. To set user-defined request headers, use the Headers match criteria.
HTTP Request Cookies	Filters by cookie value in addition to cookie names, as follows: • **Name**: Type the name of the cookie as it appears on your web site. • **Match Type**: Select if you want to search for the full value or just for the prefix. • **Value**: Type the cookie value.
HTTP Request File Extension	Filters by file type. For each type add a new row in the table and type the file extension, for example, `.gif` **Note:** If **Case Sensitive** is enabled in the service configuration, then the **Value** that you define for **File Type** is also case sensitive, see Configuring Advanced Web Service Settings on page 424.
HTTP Request Header Name	Filters data by request HTTP headers, even if there are customized headers. Define the **Value** using a full header name or just header prefix. This parameter is case sensitive.
HTTP Request Header Value	Filters by header names in requests. SecureSphere differentiates between the header names in requests and responses. You can match on At Least One header name, Or Exclude All.
HTTP Request Host Name (Header)	Filters by host name. Enables you to type a host name by which to filter.
HTTP Request Method	Filters data by HTTP method. This match criteria can be customized to use proprietary methods. **Note:** If you want to use proprietary methods, you must update the HTTP Protocol validation policy. The Unknown HTTP Request Method rule in the policy should be customized to include the new proprietary methods.
HTTP Request Parameter Name	Filters data by HTTP parameter name.
HTTP Request Referer URL (Header)	Filters by Referer URL Prefix from the request header. This parameter is case sensitive. To set user-defined request headers, use the Headers match criteria.

Criteria	Description
HTTP Request URL	Filters data by the URL from which the query originated. This match criteria is effective only when UUT is defined, see Universal User Tracking. When the **Match Unknown Value** parameter is enabled and the value does not exist, there is a match.
HTTP Request User-Agent (Headers)	Filters by User-Agent request header.
HTTP Session	SecureSphere can track HTTP client-server sessions according to session cookies. In a typical HTTP session, the web server initiates the session by setting a cookie that the client uses and that the server updates as the session progresses. The Session match criteria can match on any of 3 statuses SecureSphere assigns to sessions that pass through it, as follows: • **Validated**: The session is validated if SecureSphere received the set cookie from the server for the session cookie. A minimum of one verified cookie is enough to consider the session as validated even if the session is based on more than one cookie. • **Invalidated**: The session is invalidated if SecureSphere received no set cookie from the server. • No Session: SecureSphere does not detect any session cookie used by the client.
HTTP Version	Filters by HTTP version. Enables you to type an HTTP version by which to filter.
Lookup Data Set Search	Filters data by Lookup Data set. Select a Field to search (the dataset should have this set as its relevance), determine if the operation should be at least one or exclude all. Then manually enter a Lookup Data group value, or choose a specific Lookup Data set.
Profiled Referer Host	Referer to the host.
Protocols	Filters data by web protocol, either HTTP or HTTPS. Using this match criteria you can differentiate between HTTP to HTTPS. You can select HTTP, HTTPS, or both.
Sensitive Dictionary Search	Filters by values in sensitive dictionaries which can be located in specific locations such as URL, header, parameters and more.
Signatures	Filters by signatures included in a dictionary that is applied to the relevant service.
Source GeoLocation	Filters by the geographic location that was the source of the traffic. **Apply this rule to unidentified locations**: When enabled, includes traffic of IP addresses that aren't assigned to a geographic location. **Apply this rule to an internal network**: When enabled, includes IP addresses if they belong to internal network **Note:** This criteria is only available if you have an active license for ThreatRadar Reputation Services Additionally, if your license expires, source IP data is no longer updated.
Source IP Addresses	Filters data by source IP addresses that are either part of a SecureSphere IP group or manually defined. Enables you to select one or more source IP addresses or type a source IP address on which to base the filter.
Time of Day	Filters data by time of day (hours), and by days of the week based on the current gateway's local timezone. Data is included for times that are highlighted. You can click a heading to mark an entire day or hour.

Criteria	Description
Violations	Filters by violations that appear in the policy rules. Using this match criteria, you can define custom rules that correlate between different policy types and different layers. The ability to correlate between the rules from different policies enables you to define more sophisticated and accurate policies.

APPENDIX I

Report Columns

Many SecureSphere reports can be configured with a tabular view which enables you to select columns included in a report.

This appendix lists the available columns that can be used in reports and provides a short description of each. Available columns include the following:

Columns Available in Alert Reports 611
Columns Available in Assessment Reports 614
Columns Available in System Event Reports 616
Columns Available in Vulnerability Reports 616
Columns Available in Task Reports 619
Columns Available in Configuration Reports 620
Columns Available in Active Module Reports 622
Columns Available in Discovery Reports 626
Columns Available in Profile Reports 630

Columns Available in Alert Reports

A number of columns are available when generating Alert reports. They include the following:

- Alert Report Available Columns on page 612: lists the available columns for use with Alert reports.
- Alert Event Available Columns on page 613: lists the available event columns for use with Alert reports.

Alert Report Available Columns

Alert Report Available Columns

Column Name	Description
Additional Applicative User Name	Additional Applicative User Name.
Alert Description	Description of the alert.
Alert End Time	Time the alert ended.
Alert Flag	If the alert has been flagged, flag applied to the alert.
Alert Name	Name of the Alert.
Alert No.	Alert number.
Alert Start Time	
Alert Type	Type of alert.
Generated by ThreatRadar	Whether or not the alert was generated by ThreatRadar Reputation Services.
Immediate Action	What, if any immediate action was taken when alert was generated.
Num. of Events	The number of events that are a part of the alert.
Read	If the alert has been read.
Remedy Incident ID	The ID of a remedy incident opened as a followed action on the alert.
Server Group	Server Group on which the alert took place.
Session ID	Session ID related to the alert.
Severity	Severity of the alert.
Source GeoLocation	The geographic location from which traffic that generated the alert originated. **Note:** Only available with a ThreatRadar Reputation Services license.
Source IP	IP address that generated the alert.
Source of Activity	Source of the activity such as Remote, Local and Unknown.
ThreatRadar Service	The specific ThreatRadar Reputation service that generated the event.
URL	A URL (or part thereof) which may have been the source of the alert.
User Agent	Agent name if the alert originated in an agent.
User Name (Applicative)	User name of the application to whom the alert is associated.

Alert Event Available Columns

From Alert Event Available Columns

Column Name	Description
Affected Rows	The number of rows that were affected by the violation.
Application	Application on which the violation took place.
Bind Variables	Name and value of bind variables.
Correlated HTTP Type	The HTTP type of the correlated event in a correlation alert. Options include URL/SoapAction.
Correlated SOAP Action	The SOAP action of the correlated event in a correlation alert.
Correlated URL Path	The URL path of the correlated event in a correlation alert.
Correlated Web Client IP	The Web Client IP from the correlated event in a correlation alert.
Correlated Web Event ID	The web event ID of the correlated event in a correlation alert.
Correlated Web Session ID	The web session ID for the correlated event in a correlation alert.
Destination IP	Destination IP address listed in the data packet that generated the violation.
Destination Port	Destination port number listed in the data packet that generated the violation.
Event ID	ID that was generated for the event by the SecureSphere gateway.
Event Time	Time the event took place.
Gateway	SecureSphere gateway responsible for monitoring the file.
Last Close	Last time the file was closed.
Parsed Query	Parsed query that generated the alert.
Proxy IP	IP address of the Proxy when in reverse proxy mode.
Query Group	Query group that was violated.
Query Group Type	Type of query group that was violated.
Raw Query	Raw query that generated the violation.
Service	Service on which the violation took place.
Signature	Signature that triggered the violation.
Source App	Application that caused the violation.
Source Host	Source IP address that caused the violation.
Source OS User	OS User that caused the violation.

Column Name	Description
Source Port	Port on which the event originated.
User Company	
User Email	
User Full Name	Full name of the user with whom alert is associated. For example, Jane Smith.
User Manager	
User Title	
Username	User name with whom alert is associated.
Violated Item	Item that was violated, for example, parameter, cookie, or URL.
Violated Description	Description of the item that was generated.

Columns Available in Assessment Reports

A number of columns are available when generating Assessment reports. The following table lists the available columns.



Note: While Tabular options are available for Assessment Reports, they are not available for Assessment Results Reports.

Assessment Report Available Columns

Column Name	Description
Error Message	Error message generated by the assessment.
Location Port	Port, as defined in SecureSphere Direct Access information.
Location User	Username, as defined in SecureSphere Direct Access information.
Manual Remediation Suggestion	Remediation suggestion manually configured in SecureSphere.
Next Run	Time and Date of next scheduled Job run.
OS Error String	OS Connectivity error. Relevant if the OS Scan Status is Error.
OS Scan Status	OS Connectivity status (Scanned, Not Scanned, Error).
Policy	Policy that generated the Assessment.
Policy Auto Closing Vulnerability	If the policy is configured to automatically close discovered vulnerabilities.
Policy Description	Assessment policy description.
Policy Vulnerability Type	
Regulation Section Name	Name of the test inside the regulation.
Regulation Section No.	Section number of the test in the regulation.
Result	Type of assessment result. Whether the assessment Passed, Failed, Error or Irrelevant.
Scan Name	Name of the scan that conducted the assessment.
Scan Run Date	Date the assessment was run.
Server Group	Name of the server group on which the assessment was run.
Service Name	Name of the service on which the assessment was run.
Severity	Severity of the vulnerability.
Site	Name of the site on which the assessment was run.
Target Server IP	IP address of server being assessed.
Test Category	Category to which the test belongs.
Test Description	Description of the test.
Test Name	Name of the assessment test.
Test Type	Type of test.
Test Type (OS/DB)	Type of server on which the assessment is being conducted.

Column Name	Description
Test Vulnerability Severity Level	Severity level of the vulnerability type. Number between 0-10.
Test Vulnerability Type CVE	CVE number of the vulnerability type.
Test Vulnerability Type Description	Vulnerability Description
Test Vulnerability Type Severity	Enumerated value (Info, Low, Medium, High, Critical).
Vulnerability Type	

Columns Available in System Event Reports

A number of columns are available when generating System Event reports. They include the following:

System Event Report Available Columns

Column Name	Description
Create Time	Time system event occurred.
Message	Message which was generated as the result of a system event.
Severity	Severity of the event: Informative, Low, Medium and High.
Subsystem	Subsystem to which the system event was reported.
User	User that generated the system event when user based.

Columns Available in Vulnerability Reports

A number of columns are available when generating Vulnerability reports. They include the following:



Note: While Tabular options are available for Vulnerability Reports, they are not available for List of Vulnerabilities Results Reports.

Vulnerability Report Global Available Columns

Column Name	Description
# Details	Number of details part of the vulnerability. For example, a vulnerability might be "Access privileges are granted to the PUBLIC role". The details would include a list of all vulnerabilities granted to the PUBLIC role.
# Observations	Filters by the number of times this vulnerability has been observed on a specific asset. This includes both multiple jobs finding the same vulnerability or a single job finding the vulnerability more than once.
Age Category	The period of time since the vulnerability has been discovered.
CVE	CVE associated with the vulnerability.
Closure Date	Date the vulnerability was closed.
Creation Date	Date the vulnerability was created.Enables you to configure a range of dates within which to search.
Database Owner	Database owner, as configured through the database owner global object and applied to the database server.
Description	N/A
Due Date	Date the vulnerability is configured to be managed.
Full Description	Lists a detailed description of the vulnerability including the IP addresses, and what the vulnerability entails.
Host	Hostname on which the vulnerability was discovered.
Informix Server Name	Name of the Informix server on which the vulnerability was discovered.
Location	Information about the source of a vulnerability. For example, Username, port, etc.
Manual Remediation Suggestion	Recommended fix for the vulnerability.
Mitigatable	Whether or not the vulnerability can be mitigated by SecureSphere
Mitigating User	Name of the user that mitigated the vulnerability.
Mitigation Comment	Comment added upon mitigation of the vulnerability.
Mitigating Status	Status of vulnerability mitigation.
Mitigation Status Change Date	Date the mitigation status was last changed.
Observation Age	The time period between the first time and the last time the vulnerability was discovered.
Occurrences	Number of occurrences of this vulnerability.
Owner	Owner to whom the vulnerability is assigned.

Column Name	Description
Policy Level - # Errors	Number of errors encountered in the policy when configured as per policy under Vulnerabilities Creation Method.
Policy Level - # Failed	Number of tests that failed during a policy run.
Policy Level - # Passed	Number of tests that passed during a policy run.
Port	Database / Service port on which the vulnerability was discovered.
Priority	Priority assigned to the vulnerability.
Regulation Section Name	Name of the regulation to which the vulnerability belongs.
Regulation Section No.	Section of the regulation to which the vulnerability belongs.
Risk	Risk level as rated from0 to 10.
Risk Level	Risk level enum (Low, Medium, High).
SID	Security Identifier associated with the Database.
Server Group	Server group on which the vulnerability was discovered.
Service	Service on which the vulnerability was discovered.
Service Type	Server type on which the vulnerability was discovered.
Severity (1-10)	Numeric severity assigned to the vulnerability.
Severity (H/M/L)	Category based severity assigned to the vulnerability (Low, Medium, High).
Short Description	Short description of the vulnerability as provided by Imperva ADC.
Site	Site on which the vulnerability was discovered.
Source	Source SecureSphere module that detected the vulnerability.
Source IP	IP address on which the vulnerability was discovered.
Source Policy	The policy which created the vulnerability.
Status	Status of the vulnerability. Possible statuses include Open, Closed, Dismissed, or Excluded.
Tracking Flag	Tracking flag assigned to the vulnerability (Accepted, Assigned, New, Rejected).
Type	Name of the vulnerability type.
Update Time	Last time the vulnerability was updated.
User Name	Username that caused the vulnerability.

Column Name	Description
Vulnerability ID	ID of the vulnerability discovered.

Vulnerability Report Vulnerability Generic Available Columns

Column Name	Description
Vulnerability Attribute 1 - 5	

Vulnerability Report Details Generic Available Columns

Column Name	Description
Details Attribute 1 - 5	

Columns Available in Task Reports

A number of columns are available when generating Task reports. They include the following:

Task Reports Available Columns

Column Name	Description
Assignee Changed Followed Action	User to whom the task was assigned when the followed action was changed.
Assignment Date	Date the task was assigned.
Closure Date	Date the task was closed.
Comment Author	User that added a comment.
Comment Date	Date comment was added.
Comment Text	Actual text of the comment.
Creation Date	Date the task was created.
Current Status	Current status of the task.
Due Date	Date the task is due.
Due in	Period of time in which the task is due.
Last Update Date	Last time a property of the task was updated.
Notify Owner	Owner who is notified if changes or updates are made.
Overdue Followed Action	Filters data by those tasks with overdue followed actions.
Priority	Priority assigned to the task.
Review List	List of reviewers assigned to the task.
Status Changed Followed Action	Whether or not a followed action has been changed.
Subsystem	The task's subsystem.
Task Assignee	User to whom the task is assigned.
Task Name	Name of the task as configured in its properties.
Task Owner	User that created the task.
Task Type	Type of task. Either Assignment or Review.

Columns Available in Configuration Reports

A number of columns are available when generating configuration reports. These columns are report dependent.



Note: If SecureSphere is in delayed activation mode, these reports may include items that have been configured, but not yet activated (saved in the Management Server, but not yet implemented in the SecureSphere gateway). For further information, see Activating Settings on page 75.

Available column report types include the following:

Configuration Reports - Sites 621
Configuration Reports - Security Policies 622

Configuration Reports - Sites

A number of columns are available when generating Site configuration reports. They include the following:

Configuration Sites Report Available Columns

Column Name	Description
Application	Name of application configured on the site.
Application Create Time	Date and Time the applications were created in SecureSphere.
Host Name	Name of application host.
Is Protected	Whether or not traffic across the application is protected by SecureSphere.
Is Server Discovered	Whether or not the application was discovered as part of a service discovery scan.
Protecting Gateway Group	Gateway group to which the application belongs.
Server Group	Server group to which the application belongs.
Server Group Creation Time	Date and time the server group to which the application belongs was created.
Server Group Mode	Server Group operation mode.
Server IP	IP address of the server hosting the application.
Server IP Creation Time	Date and server's IP address was created.
Server OS Type	Host operating system of the application.
Service	Name of services configured on the site.
Service Creation Time	Date the services were created.
Service Ports	TCP/UDP ports utilized by the service.
Service SSL Ports	SSL ports utilized by the service.
Service Type	Type of service configured on the site.
Site	Site name to which the service belongs.
Site Creation Time	Date and time the site to which the service belongs was created.

Configuration Reports - Security Policies

A number of columns are available when generating Configuration reports about Security Policies. They include the following:

Configuration Security Policies Report Available Columns

Column Name	Description
ADC Content	Whether or not this policy was defined in ADC content.
ADC Keywords	Keywords pre-defined by SecureSphere ADC.
Applied to Assets	Lists the assets to which the policy is applied.
Applied by Default	Whether or not the policy is applied by default to an asset.
Creation Time	Date and time the policy was created.
Display Web Page Response	Whether or not a violation of the policy results in the displaying of a web page response.
Last Update Time	Last date and time the policy was updated.
Match Criteria	Match criteria used to filter the policy. Only relevant to custom policies. If used on a policy that is not custom, column will be empty.
Policy Level	Type of policy that is applied. For example, service, application, network, etc.
Policy Name	Name of the policy.
Policy Type	Type of policy.
Rule Action	Actions configured if rules in the policy are violated.
Rule Enabled	Rules that are enabled in the policy.
Rule Exception	Lists all the exceptions configured for the rule.
Rule Followed Action	Followed actions configured to run as a result of the policy being violated.
Rule Has Exception	Whether or not an exception was configured for the rule.
Rule Name	Name of configured rules.
Rule Severity	Severity of configured rules.
SOM Rule Followed Action	
Vulnerability Mitigation	Whether or not the policy was configured as a mitigation policy.

Columns Available in Active Module Reports

A number of columns are available when generating Active Module reports. These columns are report dependent. Available Active Module report types include the following:

- Active Module - Change Tracking Details on page 592

- Active Module Pre-configured Objects on page 593
- Active Module Track Value Changes on page 594
- Active Module Track Value Changes Configuration on page 594

Active Module - Change Tracking Details Report Columns

A number of columns are available when generating Change Tracking Details reports. They include the following:

Active Module: Change Tracking Details Report Available Columns

Column Name	Description
IP Address	(None and Plain Value) IP address of the server on which the detected change occurred.
Job	(None and Plain Value) ID of the active assessment job
Modification	(None and Plain Value) Type of modification. New, Modified, Deleted or Missing.
OS Mod. Date	(None and Plain Value) File modification date as taken from the operating system.
OS Permissions	(None and Plain Value) File permissions as taken from the operating system.
Object	(None and Plain Value) Object that was modified.
Object List	(None and Plain Value) Name of predefined list to which the object belongs.
Object Type	(None and Plain Value) Type of object that was modified. Type of object can be either File or Registry Key.
Policy	(None and Plain Value) Name of the policy that detected the modification.
Server Group	(None and Plain Value) Server group to which modified object belongs.
Site	(None and Plain Value) Site to which modified object belongs.
Size	(None and Plain Value) Size of the modified object.
Time of Day	(None and Plain Value) Time of day modification took place.
Timestamp	(None and Plain Value) Actual timestamp of modification.

Active Module Preconfigured Objects Report Columns

A number of columns are available when generating Preconfigured Objects reports. They include the following:

Active Module: Change Tracking Preconfigured Objects Report Available Columns

Column Name	Description
Object List	(None and Plain Value) Name of the predefined object list.
Object Type	(None and Plain Value) Type of object that was modified.
Operating System	(None and Plain Value) Operating System to which the objects are relevant.
Pattern	(None and Plain Value) Name of the object or the object pattern.
Report Missing	(None and Plain Value) Whether or not the object should be reported if missing.

Active Module Track Value Changes Report Columns

A number of columns are available when generating Track Value Changes reports. They include the following:

Active Module: Track Value Changes Report Available Columns

Column Name	Description
After Value	(None, Plain, Maximum and Sum) Updated value of item that was changed.
Before Value	(None, Plain, Maximum and Sum) Original value of item that was changed.
DB Session ID	(None, Plain) Session in which the modification took place.
Column Name	(Plain and Group By) Name of the column containing the updated value.
DB Session ID	(Plain and Group By) ID of the database session in which the change took place.
DB User Owner Name	(Group By, Plain) Schema name of the database that is logged as having made the change.
Database User	(Group by, Plain) Database user that made the modification.
Hour	(Group by and Plain) Hour that the modification was made.
Percentage Type	(Group by, Plain, and Maximum) Type of percentage.
Policy	(Group by, Plain) Type of policy.
Row Unique Identifier	(Group by, Plain) Internal ID of the row that was changed.
Serial Number	(Group by, Plain) Session's Serial Number.
Server IP	(Group by, Plain) IP address of the server on which the modified data is located.
Source Application	(Group by, Plain) Application used to modify data.
Source OS Host Name	(Group by, Plain) OS hostname from which the data was modified.
Source OS User Name	(Group by, Plain) OS username from which the data was modified.
Table Name	(Group by, Plain) Table in which modified data is located.
Transaction Time Stamp	(Group by, Plain) Time the transaction took place.

Active Module Track Value Changes Configuration Report Columns

A number of columns are available when generating Track Value Changes Configuration reports. They include the following:

Active Module: Track Value Changes Configuration Report Available Columns

Column Name	Description
Column Name	(None, Plain) Column being monitored.
Column Type	(None, Plain) Type of column being monitored.
DB User Owner Name	(None, Plain) The schema name where the monitored table resides.
Policy	(None, Plain) Name of policy that is tracking changes.
Server IP	(None, Plain) IP address of server that is being monitored.
Table Name	(None, Plain) Name of the table being monitored.

Columns Available in Discovery Reports

A number of columns are available when generating Discovery reports. These columns are report dependent. Available discovery report types include the following:

- Service Discovery Detailed Report on page 589
- Service Discovery Execution Report on page 590

Service Discovery Detailed Report Columns

Service Discovery Detailed Report Columns

Column Name	Description
DB Credential	DB Credentials used during discovery.
Data Findings	Data Findings
Decision	Decision on the status of a discovered service. Can be pending, accepted, accepted automatically or rejected.
Decision Changed Date	Date the service status was changed.
Discovery Date	Date the service was discovered.
Execution Date	Execution date. Date service discovery was conducted.
First Discovery Date	First date the service was discovered.
Host	Hostname on which the service was discovered.
ID	Discovery ID.
IP	IP address on which the service was discovered.
Informix Server Name	Name of the informix server used in credentials verification.
Last Discovery Date	Last time the service was discovered.
OS Credentials	OS credentials used to access the service.
OS Extra Information	Extra information discovered regarding the service OS.
OS Name	Name of the Operating System.
OS Type	Type of operating system discovered.
OS Vendor	Operating System vendor name. For example, Microsoft or Linux.
OS Version	Version of the discovered Operating System.
Port	Port on which the service is running.
SID	Security Identifier used for credentials verification.
Scan	Name of the service discovery scan.
Server Auth. Method	Authentication method used to connect with the server.
Server Group	Server group to which the service belongs.
Server User Name	User name used to access the server.
Service	Service

Column Name	Description
Service Extra Information	Extra information discovered regarding the service.
Service Name	Name of the service.
Service Status	Status of the service.
Service Type	Type of service.
Service Version	Version of the service running.
Site	Site to which the service belongs.
Username	Username used for credentials verification.

Service Discovery Execution Summary Report Columns

Service Discovery Execution Summary Report Columns

Column Name	Description
Execution Date	Filters data by policy name.
Number of Existing Servers Found	Lists the number of servers found during discovery.
Number of Existing Services Found	Lists the number of servers already existing in SecureSphere found during discovery. A separate column is available for each type of Server supported by SecureSphere including **HTTP**.
Number of New Servers Found	Lists the total number of services found during discovery.
Number of New Services Found	Lists the number of new services found during discovery. A separate column is available for each type of Server supported by SecureSphere including **HTTP**.
Number of Servers Found	Lists the total number of servers found during discovery.
Number of Services Accepted	Lists the number of services accepted. Lists the number of services accepted. A separate column is available for each type of Server supported by SecureSphere.
Number of Services Accepted	Lists the number of services accepted by SecureSphere. A separate column is available for each type of Server supported by SecureSphere including **HTTP**.
Number of Services Automatically Accepted	Lists the number of services automatically accepted by SecureSphere.
Number of Automatically Services Accepted	Lists the number of services automatically accepted by SecureSphere. A separate column is available for each type of Server supported by SecureSphere including **HTTP**.
Number of Services Found	Lists the total number of services found during discovery.
Number of Services Pending	Lists the total number of services pending review. Lists the number of services accepted. A separate column is available for each type of Server supported by SecureSphere.
Number of Services Pending	Lists the number of services pending review. A separate column is available for each type of Server supported by SecureSphere including **HTTP**.
Number of Services Rejected	Filters data by number of services that were automatically accepted. A separate column is available for each type of Server supported by SecureSphere.
Number of Services Rejected	Lists the number of services rejected. A separate column is available for each type of Server supported by SecureSphere including **HTTP**
Scan	Name of the service discovery scan.

Columns Available in Profile Reports

A number of columns are available when generating Profile reports. These columns are report dependent. Available Profile report types include the following:

- Web Profile URL Report on page 595
- Web Profile Cookies Report on page 597
- Web Profile - Learned Hosts Report on page 598
- Web Profile - Susceptible Directories Report on page 599
- Web Profile - Web Application User Tracking Report on page 599

Web Profile URL Report Columns

A number of columns are available when generating URL reports. The following table lists the columns available for generating URL reports.

Report: Web Profile URL - Column Names

Criteria	Description
Application	Application the URL belongs to.
Folder	Folder in which the file in the URL is located.
Learning State	SecureSphere learning state of the URL (learning, in protection).
Learning State Switch Date	Last date the URL learning state was changed.
Link Status	Status of the link.
Locked	Whether or not the profile is locked and no longer learning.
Method	HTTP method used by the URL.
Number of Correlated Queries	Number of correlated queries in the URL.
Number of Parameters	Number of parameters used by the URL.
Occurrences	Number of occurrences before this URL was learned.
Parameter Additional Allowed Characters	Number of additional characters allowed in URL.
Parameter Max Length	Maximum length of a parameter in the URL.
Parameter Min Length	Minimum length of a parameter in the URL.
Parameter Name	Name of parameter in the URL.
Parameter Prefix	If the parameter is a prefix.
Parameter Read Only	If the parameter is read only.
Parameter Required	If the parameter is required.
Parameter Type	The type of parameter. For example, Numeric, Foreign Language Characters (UTF-8), Latin Characters.
Referer URL	URL that was referred as the result of a broken link.
Response Time	Response time of the URL.
Soap	Whether or not the URL uses Soap.
Soap Action	Action taken if URL is Soap based.
Soap Parameter Max Length	Maximum length of Soap parameter in URL.
Soap Parameter Min Length	Minimum length of Soap parameter in URL.

Criteria	Description
Soap Parameter Name	Name of Soap Parameter.
Soap Parameter Required	Whether or not Soap parameter is required in URL.
Soap Parameter Type	Type of Soap parameter.
SQL Correlated Query	The query that is run by the URL. For example, the database activity we have correlated back to this URL.
SQL Correlated Query Application	The database application that generated the query.
SQL Correlated Query Service	The database service that generated the query.
SQL Correlated Query User	The database user that runs the query.
Server Group	Server group the URL belongs to.
Service	Service the URL belongs to.
Site	Site the URL belongs to.
URL	The full path of the URL.
URL Occurrence	The number of URLs profiled.
Creation Date	Date the URL was created.
Times tamp	Timestamp of URL creation.

Web Profile Cookies Report Columns

A number of columns are available when generating Cookie reports. The following table lists the columns available for Cookie reports.

Web Profile Cookies - Columns

Criteria	Description
Application	Application the cookie belongs to.
Injection	Whether or not the cookie was injected.
Occurrences	Number of cookies profiled.
Prefix	Whether or not the cookie has a prefix.
Server Group	Server group to which the cookie belongs.
Service	Service to which the cookie belongs.
Site	Site to which the cookie belongs.
Status	Status of cookie profile (learn, protect, ignore).
Creation Date	Date the cookie was created.
Time Stamp	Timestamp of the cookie.

Web Profile - Learned Hosts Report Columns

A number of columns are available when generating Learned Hosts reports. The following table lists the columns available for Learned Hosts reports.

Web Profile Learned Hosts - Columns

Criteria	Description
Application	Application related to the learned host
Host	Domains or IP that are recognized as belonging to the application.
Occurrences	Number of hosts learned.
Server Group	Server group to which the host belongs.
Service	Service to which the host belongs.
Site	Site to which the host belongs.

Web Profile - Susceptible Directories Report Columns

A number of columns are available when generating Susceptible Directories reports. The following table lists the columns available for Susceptible Directories reports.

Web Profile Susceptible Directories - Columns

Criteria	Description
Application	Application the directory belongs to.
Directory	Name of the directory
Directory Status	Status of the directory (learning, protection).
Occurrences	Number of times the directory was experienced until it was profiled.
Server Group	Server group to which the directory belongs.
Service	Service to which the directory belongs.
Site	Site to which the directory belongs.
Directory Creation Date	Date the directory was created.
Timestamp	Timestamp of the directory.

Web Profile - Web Application User Tracking Report Columns

A number of columns are available when generating Web Application User Tracking reports. The following table lists the columns available for Web Application User Tracking reports.

Web Profile Web Application User Tracking - Columns

Criteria	Description
Action URL	URL in which users login to the web application.
Action URL Creation Time	Date the user was created.
Action URL Update Time	Timestamp of the user.
Application	Application the user belongs to.
Default Decision	Whether succeeded failed, or can't tell.
Field 1	The first field of the login / user identification fields. For example, User Name.
Field 2	The second field, if required, of the user identification fields. For example, Password.
Login Result	If the user login succeeded, failed or can't tell.
Login URL	The URL from which the user reached action URL.
Occurrences	Number of action URLs (used when grouping the report by application).
Pattern URL	A pattern used to identify login success.
Redirect URL Value	URL user is redirected to after login attempt.
Response Code Value	The action type by which a login is successful including URL Redirect, Response Code or Pattern Returned.
Server Group	Server group to which the user belongs.
Service	Service to which the user belongs.
Site	Site to which the user belongs.
Status	Status of the User, in Learning, in protection or manual.
Type	The type of action url: redirect, response or pattern.

APPENDIX J

Pre-Defined Reports

This Appendix describes the various types pre-defined reports available in SecureSphere. SecureSphere provides you with a number of reports that have been pre-defined which can be used as is on-the-fly, or be further customized to suit your needs. These reports fulfill a number of needs such as assisting in providing specific information to meet regulatory requirements.

This appendix reviews the following:

For further information about configuring and using reports, see Reporting on page 28.

Alert Reports 637
Active Module Reports 641
System Events Reports 645

Alert Reports

Alert Reports

Name	Description	Additional Keywords	ISO 27001	SOX	PCI	HIPAA	Basel II	Privacy	GLBA
PCI - Alerts on violations by administrative user	Alerts on all violations caused by administrative users. Supports PCI DSS 10.2.4.				X				
PCI - Alerts on violations of privileged commands	Alerts on violations of privileged commands. Support PCI DSS 10.2.4.				X				
PeopleSoft - Report Access to PeopleSoft Schema	Report suspected access to PeopleSoft schema.	PeopleSoft							
PeopleSoft - Report Querying of Tables by Non-Default Users	Report suspected querying of tables by non default users.	PeopleSoft	7.1.3				X		
PeopleSoft - Report Querying with Unknown Application	Report suspected querying of tables with Unknown Applications.	PeopleSoft	7.1.3				X		
PeopleSoft - Report Querying of Tables from Unknown IPs	Report suspected querying of tables from unknown IPs.	PeopleSoft	7.1.3				X		
PeopleSoft - Report Changes by Non-Default Users	Report suspected changes of tables by non-default users.	PeopleSoft	10.1.2				X		
PeopleSoft - Report Changes by Non-Default Applications	Report suspected changes of tables by non-default applications.	PeopleSoft	10.1.2				X		
PeopleSoft - Report Changes from Unknown IPs	Report suspected changes of tables from unknown IPs.	PeopleSoft	10.1.2				X		
PeopleSoft - Report Changes to Login Tables	Report suspected changes to user login tables.	PeopleSoft	10.1.2				X		
PeopleSoft - Report Changes to User Admin Tables	Report suspected changes to user administration tables.	PeopleSoft	10.1.2				X		
PeopleSoft - Report Query of Payment Card Info Columns	Report suspected querying of payment cards information columns.	PeopleSoft			X				
PeopleSoft - Report Query of Identification Info Columns	Report suspected querying of personal identification information columns.	PeopleSoft						X	

Name	Description	Additional Keywords	ISO 27001	SOX	PCI	HIPAA	Basel II	Privacy	GLBA
PeopleSoft - Report Query of Personal Info Columns	Report suspected querying of personal information columns.	PeopleSoft						X	
PeopleSoft - Report DB Management by Non-Default Users	Report suspected database management by non-default applications.	PeopleSoft	7.1.3				X		
PeopleSoft - Report DB Management by Non-Default Applications	Report suspected database management by non default users.	PeopleSoft	7.1.3				X		
PeopleSoft - Report DB Management from Unknown IPs	Report suspected database management from unknown IPs	PeopleSoft	7.1.3				X		
Weekly Data Leakage Summary Report	Weekly Data Leakage Summary Report - Alerts with CC or SSN signature.		12.2.4						
Daily Summary Blocked Connections	Displays alerts information on blocked connections over the last 24 hours.								
Weekly Summary Top 10 DB Violations	Weekly Summary Top 10 DB Violations.		13.1.1				X		
Weekly Summary Top 10 WAF Violations	Weekly Summary Top 10 WAF Violations.		13.1.1				X		
Weekly Summary Top 10 All Violations	Weekly Summary Top 10 All Violations.		13.1.1				X		
Daily Top 10 DB Violations	Daily Top 10 DB Violations.		13.1.1				X		
Daily Top 10 WAF Violations	Daily Top 10 WAF Violations.		13.1.1				X		
Daily Top 10 All Violations	Daily Top 10 All Violations.		13.1.1				X		
Daily Report for Outbound U.S Social Security Number Visibility - Data Leakage	Report of last 24 hours for outbound U.S Social Security Number visibility - data leakage.		12.2.4,12,5,4					X	X
Daily Report for Outbound Credit Card Numbers Visibility - Data Leakage	Display Report of last 24 hours for outbound Credit Card Numbers visibility - data leakage.		12.2.4,12,5,4					X	
SAP - Report Suspected Activity in User Administration Tables	Report suspected user administration tables activity.	SAP	13.1.1				X		

Name	Description	Additional Keywords	ISO 27001	SOX	PCI	HIPAA	Basel II	Privacy	GLBA
SAP - Report Suspected Activity in Data Dictionary Tables	Report suspected SAP data dictionary tables activity.	SAP	13.1.1				X		
SAP - Report Suspected Activity in Confidential Financial Data and Vendor Tables	Report suspected SAP confidential financial data and vendor tables activity.	SAP	13.1.1				X		
SAP - Report Suspected Activity in User Master Data Tables	Report suspected SAP user master data tables activity.	SAP	13.1.1				X		
SAP - Report Suspected Activity in Accounting Documents Tables	Report suspected SAP accounting documents tables activity.	SAP	13.1.1				X		
SAP - Report Suspected Activity in Personal Info Tables	Report suspected SAP personal info tables activity.	SAP	13.1.1				X		
SAP - Report Querying of Tables by Non-Default User	Report suspected querying of SAP tables by non-default user.	SAP	7.1.3				X		
SAP - Report Suspected modifications in Tables by Non-Default User	Report suspected changing of SAP tables by non-default user.	SAP	13.1.1				X		
SAP - Report Querying of Tables with a Non-Default Application	Report suspected querying of SAP tables with a non-default application.	SAP	7.1.3				X		
SAP - Report Suspected Access to Tables with a Non-Default Application	Report suspected access to SAP tables with a non-default application.	SAP	13.1.1				X		
SAP - Report Suspected modification of Metadata	Report suspected modifications of SAP Metadata.	SAP	13.1.1				X		
SAP PCI - Alerts on violation to a card number	Alerts on violations to tables belonging to payment cards: card number information. Supports PCI DSS 10.2.4.	SAP			X				
SAP PCI - Alerts on violation to a cardholder	Alerts on violations to tables belonging to payment cards: cardholder name and expiration date information. Supports PCI DSS 10.2.4.	SAP			X				

Name	Description	Additional Keywords	ISO 27001	SOX	PCI	HIPAA	Basel II	Privacy	GLBA
SOX - Unauthorized data changes	Display all changes to financial data that violate the normal access policy.	SOX	7.1.3				X		
SOX - Application data changes by administrator	Display changes to financial data committed by administrators.	SOX	7.1.3				X		
SOX - Manual data changes	Display changes to financial data committed by free query applications.	SOX	7.1.3				X		
SOX - Unauthorized direct changes to data dictionary	Display attempts to change structure of database by directly manipulating data dictionary tables.	SOX	7.1.3				X		

Active Module Reports

Active Module Reports

Name	Description	Additional Keywords	ISO 27001	SOX	PCI	HIPAA	Basel II	Privacy	GLBA
Track Value Changes - Last 3 Days Report	Display all value changes that were committed over the last 72 hours.		10.1.2				X		
Track Value Changes - Outside of working hours	Display changes that were committed outside of working hours on the last 72 hours.		10.1.2				X		
Track Value Changes - Manual Changes	Display changes that were invoked by a free query application over the last 72 hours.		10.1.2				X		
Track Value Changes - Manual Changes, outside of working hours	Display changes that were invoked by a free query application outside of working hours. S		10.1.2				X		
Track Value Changes - Major Changes	Display major value changes that were committed over the last 72 hours.		10.1.2				X		
Track Value Changes - Configuration Report	Display the track value changes configuration: server ip, ownername, table and tracked colnames.		10.1.2				X		
Track Value Changes - All Deleted Rows	Display all value changes that were committed over the last 72 hours - Filter delete Operations.		10.1.2				X		
Track Value Changes - All Update Rows	Display all value changes that were committed over the last 72 hours - Filter update Operations.		10.1.2				X		
Track Value Changes - All New Rows	Display all value changes that were committed over the last 72 hours - Filter insert Operations.		10.1.2				X		
Track Value Changes - Last 24 Hours Report	Display all value changes that were committed over the last 24 hours.		10.1.2				X		
Change Tracking - Objects - Daily Report	Display all objects that were modified in the past day								

Name	Description	Additional Keywords	ISO 27001	SOX	PCI	HIPAA	Basel II	Privacy	GLBA
Change Tracking - Daily Change Report	Display all changes for the past day								
Change Tracking - Objects - Weekly Report	Display all objects that were modified in the past week								
Change Tracking - Weekly Change Report	Display all changes for the past week								
Change Tracking - New Objects - Daily Report	Display all objects that were created in the past day								
Change Tracking - New Objects - Weekly Report	Display all objects that were created in the past week								
Change Tracking - Missing Objects - Monthly Report	Display all objects that were reported missing in the past month								
Change Tracking - Modified Objects - Daily Report	Display all objects that were modified in the past day								
Change Tracking - Modified Objects - Weekly Report	Display all objects that were modified in the past week								
Change Tracking - Weekly Change Report - Oracle	Display all changes for the past week of Oracle related objects								
Change Tracking - Weekly Change Report - MS SQL Server	Display all changes for the past week of SQL Server related objects								
Change Tracking - Weekly Change Report - Sybase	Display all changes for the past week of Sybase ASE related objects								
Change Tracking - Weekly Change Report - DB2	Display all changes for the past week of DB2 related objects								
Change Tracking - SecureSphere Agent Files - Weekly	Display all changes to SecureSphere Remote Agent files in the past week								
Change Tracking - Default Objects	List all default objects								
Sensitive Data Discovery - All data	Display all discovered sensitive data.		7.1.1				X		

Name	Description	Additional Keywords	ISO 27001	SOX	PCI	HIPAA	Basel II	Privacy	GLBA
Sensitive Data Discovery - All data by last discovery date	Display all discovered sensitive data along with their discovery date.		7.1.1				X		
Sensitive Data Discovery - Name based	Display all discovered sensitive data that were detected by the name based method								
Sensitive Data Discovery - Content based	Display all discovered sensitive data that were detected by the content based method								
Sensitive Data Discovery - Name and Content based	Display all discovered sensitive data that were detected by both name and content method								
Sensitive Data Discovery - Credentials	Display all discovered sensitive data that belongs to the category Credentials								
Sensitive Data Discovery - Financials	Display all discovered sensitive data that belongs to the category Financials. Supports GLBA and privacy regulations.							X	X
Sensitive Data Discovery - SOX	Display all discovered sensitive data that belongs to the category SOX	SOX							
Sensitive Data Discovery - Personal	Display all discovered sensitive data that belongs to the category Personal. Supports GLBA and privacy regulations.							X	X
Sensitive Data Discovery - Discovered during last week	Display all discovered sensitive data that were discovered during the last week								
Sensitive Data Discovery - Disappeared during last week	Display all discovered sensitive data that were disappeared during the last week								
Sensitive Data Discovery - Summary Charts	Display charts that summaries the detected data								
Sensitive Data Discovery - Tables Summary	Display all discovered tables contains sensitive data								

Name	Description	Additional Keywords	ISO 27001	SOX	PCI	HIPAA	Basel II	Privacy	GLBA
Vulnerability Database - All data	Display all reported vulnerabilities								
Vulnerability Database - All data by first/last discovery date	Display all reported vulnerabilities with their first and last discovery date								
Vulnerability Database - All data by mitigation status	Display all reported vulnerabilities with their mitigation status								
Vulnerability Database - All data by vulnerability location	Display all reported vulnerabilities with the vulnerable resource								
Vulnerability Database - All data by URL and Method	Display all reported vulnerabilities with their URL and method								
Auto Server Discovery - All Data by IP	Display all discovered servers by their IP.		7.1.1				X		
Auto Server Discovery - All Data by Server Type	Display all discovered servers by server type.		7.1.1				X		
Auto Server Discovery - Web Servers	Display all web servers discovered.		7.1.1				X		
Auto Server Discovery - Database Servers	Display all database servers discovered.		7.1.1				X		
Auto Server Discovery - Oracle Servers	Display all Oracle servers discovered.		7.1.1				X		
Auto Server Discovery - MSSQL Servers	Display all MSSQL servers discovered.		7.1.1				X		
Auto Server Discovery - DB2 Servers	Display all DB2 servers discovered.		7.1.1				X		
Auto Server Discovery - Sybase Servers	Display all Sybase servers discovered.		7.1.1				X		
Auto Server Discovery - Discovered last week	Display all servers discovered during the last week.		7.1.1				X		
Auto Server Discovery - Disappeared last week	Display all servers that have not been discovered during the last week, but did before.		7.1.1				X		
Auto Server Discovery - Discovered During Last Scan	Display all servers discovered for the first time during the last scan		7.1.1				X		

Name	Description	Additional Keywords	ISO 27001	SOX	PCI	HIPAA	Basel II	Privacy	GLBA
Auto Server Discovery - Discovered Automatically	Display all servers discovered automatically by the module.		7.1.1				X		

System Events Reports

System Events Reports

Name	Description	Additional Keywords	ISO 27001	SOX	PCI	HIPAA	Basel II	Privacy	GLBA
All Login Events to SecureSphere	Display Login events to SecureSphere								
Failed Login Attempts to SecureSphere	Display all Failed Login Attempts to SecureSphere								
Security alerts in SecureSphere	Display all security alerts in SecureSphere.		13.1.1				X		
SecureSphere Privilege Changes	Report all changes to SecureSphere user privileges.		7.1.3				X		
SecureSphere New Users	Report SecureSphere New Users.		7.1.3				X		

APPENDIX K

Alert Aggregation Descriptions

This Appendix provides a description for the various types of aggregations used to group alerts. An alert aggregations is a group of related violations that have been aggregated into a single alert to assist in identifying attacks.

For a list of alert aggregation descriptions, see Alert Aggregation Descriptions on page 646

For further information regarding aggregated alerts, see Understanding Aggregated Alerts on page 94.

Alert Aggregation Descriptions 646

Alert Aggregation Descriptions

The following table lists the various alert types, their aggregation type, a description of the aggregation and key fields used.

Alert Aggregation Descriptions

Alert Name	Aggregation Type	Aggregation Description	Key Fields
Abnormally Long Header Line	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
Abnormally Long Header Line	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Abnormally Long Request	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Abnormally Long Request	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Access to a black-listed table	full	All alerts of this type with the same source IP, OS username, same destination Server Group, destination IP and the same black list table are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	server-group-id, content-type
Access to a black-listed table	partial	All alerts of this type with the same source IP, OS username, destination Server Group and destination IP are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	server-group-id
Access to a black-listed table	partial	All alerts of this type with the same destination Server Group, destination IP and same black list table are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	server-group-id, server-ip, sql-os-user
Attempt to Execute Privileged Operation	full	All attempts to execute Privileged Operations alerts with the same destination Server Group, username and unauthorized Privileged Operation are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	application-id, source-ip, session, web-user
Attempt to Execute Privileged Operation	partial	All attempts to execute Privileged Operations alerts with the same destination Server Group and the same username are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different unauthorized Privileged Operations.	server-group-id, source-ip
Bad IP Option Length	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have	service-id, source-ip

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	
Bad IP Option Length	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
Bad IP Option Padding	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
Bad IP Option Padding	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
Bad Source IP Address	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
Bad Source IP Address	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
Bad TCP Option Length	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
Bad TCP Option Length	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
Bad TCP Option Padding	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates	service-id, source-ip

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		violation alerts generated on the same socket. Only the first alert is presented below.	
Bad TCP Option Padding	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
Blocked HTTP User	full	Alerts invoked as a followed action of the same rule, with the same source Web User and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	application-id, cookie-name
Cookie Injection	full	All cookie-related profile violation alerts with the same source IP or source session, the same destination server group, and the same cookie name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	response-code, source-ip, session, server-group-id
Cookie Injection	partial	All cookie-related profile violation alerts with the same destination server group, and the same cookie name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same cookie name. This could indicate a false positive (i.e. the cookie is not traceable) or a widespread attack.	server-ip, table, server-group-id, source-ip, sql-os-user
Cookie Tampering	full	All cookie-related profile violation alerts with the same source IP or source session, the same destination server group, and the same cookie name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	response-code, source-ip, session, server-group-id
Cookie Tampering	partial	All cookie-related profile violation alerts with the same destination server group, and the same cookie name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same cookie name. This could indicate a false positive (i.e. the cookie is not traceable) or a widespread attack.	server-ip, table, server-group-id, source-ip, sql-os-user
Cross-site scripting	full	All Cross Site Scripting violations with the same session ID, source-IP, user, destination application, URL and Parameter are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	url, source-ip, method, host, parameter-name, session, server-group-id
Cross-site scripting	partial	All Cross Site Scripting violations with the same destination application, URL and parameter are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	application-id, parameter-name, web-user, session, source-ip, url

Alert Name	Aggregation Type	Aggregation Description	Key Fields
Cross-site scripting	partial	All Cross Site Scripting violations with the same session ID, source-IP, user and destination application are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	url, source-ip, parameter-name , session, web-user, application-id
Custom Violation	full	All "Custom Policy Violation" alerts with the same source IP, destination Server Group and policy rule are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates similar custom policy alerts generated by the same source.	cookie-name, application-id, source-ip, web-user, session
Custom Violation	full	All "Custom Policy Violation" alerts with the same source IP, destination Server Group and policy rule are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates similar custom policy alerts generated by the same source.	cookie-name, application-id, source-ip, web-user, session
Custom Violation	partial	All "Custom Policy Violation" alerts with the same destination Server Group and policy rule are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates similar custom policy alerts generated by different sources.	sql-os-user, source-ip, server-ip, server-group-id
Custom Violation	partial	All "Custom Policy Violation" alerts with the same destination Server Group and policy rule are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates similar custom policy alerts generated by different sources.	sql-os-user, source-ip, server-ip, server-group-id
Double URL Encoding	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Double URL Encoding	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Excessive Attempts of Database Login	full	All alerts of this type with the same source IP, OS username, destination Server Group and destination IP are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	server-group-id, url, host, method, read-only-param eter
Excessive Attempts of Database Login	partial	All alerts of this type with the same destination Server Group and destination IP are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule	parameter-name , host, method, url, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		aggregates profile violation alerts that involve different queries.	
Excessive Login Attempts in Mid Database Session	full	All alerts of this type with the same source IP, OS username, destination Server Group and destination IP are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	server-group-id, url, host, method, read-only-parameter
Excessive Login Attempts in Mid Database Session	partial	All alerts of this type with the same destination Server Group and destination IP are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	parameter-name, host, method, url, server-group-id
Extremely Long Parameter	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Extremely Long Parameter	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Forceful Browsing	full	All URL-related correlation alerts with the same source IP or source session, the same destination Server Group, and the same HTTP host name, URL, and method are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id, rule-id
Forceful Browsing	partial	All URL-related correlation alerts with the same destination server group, and the same source IP or source session are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by the same source but on different URLs. This could indicate a false positive or a widespread attack.	parameter-name, url, application-id
Forceful Browsing	partial	All URL-related correlation alerts with the same destination Server Group, and the same HTTP host name, URL, and method are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same URL. This could indicate a false positive (i.e. the URL exists but is not part of the profile) or a wide spread attack.	cookie-name, source-ip, server-group-id, session
Fragmented Packet	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates	url, source-ip, method, host, soap-action, session, server-group-id,

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		violation alerts generated on the same socket. Only the first alert is presented below.	soap-parameter
Fragmented Packet	partial	All alerts of this type with the same destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated by different sources. Only the first alert is presented below.	server-group-id, server-ip, table
HTTP access attempt - blocked session	full	Alerts invoked as a followed action of the same rule, with the same Session ID and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	url, server-group-id, method, soap-action, host
HTTP Signature Violation	full	All Signature alerts with the same source IP, destination Service, session ID, protocol and attack type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id, host, source-ip, url
HTTP Signature Violation	partial	All Signature alerts with the same source IP, destination Service, protocol and session ID are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates Signature alerts with different signatures all coming from the same source (e.g. scanner).	server-port, protocol, server-group-id
HTTP Signature Violation	partial	All Signature alerts with the same destination Service, attack type and protocol are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates similar Signature alerts generated by different sources. This type of aggregation could indicate a false positive or a widespread worm.	source-ip, server-group-id
ICMP - LAND Attack	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
ICMP - LAND Attack	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
Illegal Byte Code Character in Header Name	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
Illegal Byte Code Character in Header Name	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Illegal Byte Code Character in Method	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Illegal Byte Code Character in Method	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Illegal Byte Code Character in Parameter Name	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Illegal Byte Code Character in Parameter Name	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Illegal Byte Code Character in Parameter Value	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Illegal Byte Code Character in Parameter Value	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Illegal Byte Code Character in Query String	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Illegal Byte Code Character in Query String	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
Illegal Byte Code Character in URL	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Illegal Byte Code Character in URL	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Illegal Chunk Size	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Illegal Chunk Size	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Illegal Content Length	full	All Illegal Content Length violation alerts with the same host or url or source IP, the same destination Server Group and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	session, source-ip, server-group-id
Illegal Content Length	partial	All "Illegal Content Length" violation alerts with the same destination Server Group and the same source IP are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates alerts generated by the same source but on different URLs and using different content types.	source-application, server-group-id, user
Illegal Content Length	partial	All "Illegal Content Length" violation alerts with the same destination Server Group, the same host and the same url are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates alerts generated by the same source but on different URLs and using different content types.	sql-os-user, source-ip, server-group-id
Illegal Content Type	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Illegal Content Type	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
Illegal Host Name	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Illegal Host Name	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Illegal HTTP Version	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Illegal HTTP Version	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Illegal Parameter Encoding	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Illegal Parameter Encoding	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Illegal Response Code	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Illegal Response Code	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Illegal TCP Option	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is	service-id, source-ip

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		presented below.	
Illegal TCP Option	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
Illegal URL Path Encoding	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Illegal URL Path Encoding	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Incorrect cookie signature	partial	All incorrect cookie signature violations with the same session ID, source-IP, user and destination application are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id, url, host, method
Incorrect cookie signature	partial	All incorrect cookie signature violations with the same application and cookie name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	query, user, server-group-id
Invalid IP Flags	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
Invalid IP Flags	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
IP - Bad Fragment	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip

Alert Name	Aggregation Type	Aggregation Description	Key Fields
IP - Bad Fragment	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
IP - Bad Total Length	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
IP - Bad Total Length	partial	All Firewall alerts with the same destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, source-ip, server-group-id
IP - Micro-Fragment	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
IP - Micro-Fragment	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
IP - Too Small TTL	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
IP - Too Small TTL	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
Malformed HTTP Header Line	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Malformed HTTP Header Line	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different	session, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		sources.	
Malformed SOAP Message	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Malformed SOAP Message	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Malformed URL	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Malformed URL	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Missing cookie signature	full	All missing cookie signature violations with the same session ID, source-IP, user, destination application and cookie name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	source-port, server-group-id, server-ip, server-port, source-ip
Missing cookie signature	partial	All missing cookie signature violations with the same session ID, source-IP, user and destination application are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	application-id, url, parameter-name
Missing cookie signature	partial	All missing cookie signature violations with the same application and cookie name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	source-ip, server-ip, server-group-id, source-port, server-port
Modified TCP Retransmission Data (client to server only)	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
Modified TCP Retransmission Data (client to	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have	server-group-id, header

Alert Name	Aggregation Type	Aggregation Description	Key Fields
server only)		passed since the alert was first generated. Only the first alert is presented below.	
Monitored Database User	full	Alerts invoked as a followed action of the same rule, with the same source Database User and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	response-code, server-group-id
Monitored HTTP Session	full	Alerts invoked as a followed action of the same rule, with the same Session ID and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	url, server-group-id, method, soap-action, host
Monitored HTTP User	full	Alerts invoked as a followed action of the same rule, with the same source Web User and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	application-id, cookie-name
Monitored IP	full	Alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id, session
Non-SOAP Access to a SOAP Only URL	full	All Unauthorized Non-SOAP Access violation alerts with the same destination Server Group, HTTP host name, URL, method, session ID and source IP are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates identical alerts.	source-ip, server-group-id, method, host, url, session
Non-SOAP Access to a SOAP Only URL	partial	All Unauthorized Non-SOAP Access violation alerts with the same destination Server Group, and the same HTTP host name, and URL are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same URL.	server-group-id, source-ip
NULL Character in Header Name	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
NULL Character in Header Name	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
NULL Character in Method	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have	user, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		passed since the alert was first generated.	
NULL Character in Method	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
NULL Character in Parameter Name	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
NULL Character in Parameter Name	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
NULL Character in Parameter Value	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
NULL Character in Parameter Value	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
NULL Character in Query String	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
NULL Character in Query String	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
NULL Character in Url	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
NULL Character in Url	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an	session, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	
Parameter Read Only Violation	full	All parameter-related profile violation alerts with the same source IP or source session, the same destination Server Group, and the same HTTP host name, URL, method and parameter name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id, web-user
Parameter Read Only Violation	partial	All parameter-related profile violation alerts with the same destination Server Group, and the same HTTP host name, URL, method and parameter name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same parameter name. This could indicate a false positive (i.e. the parameter's actual behavior is different than the profiled behavior) or a wide spread attack.	service-id, protocol, session, attack-id, source-ip
Parameter Type Violation	full	All parameter-related profile violation alerts with the same source IP or source session, the same destination Server Group, and the same HTTP host name, URL, method and parameter name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id, sql-os-host
Parameter Type Violation	partial	All parameter-related profile violation alerts with the same destination Server Group, and the same HTTP host name, URL, method and parameter name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same parameter name. This could indicate a false positive (i.e. the parameter's actual behavior is different than the profiled behavior) or a wide spread attack.	session, web-user, application-id, source-ip
Parameter Value Length Violation	full	All parameter-related profile violation alerts with the same source IP or source session, the same destination Server Group, and the same HTTP host name, URL, method and parameter name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id, sql-os-host
Parameter Value Length Violation	partial	All parameter-related profile violation alerts with the same destination Server Group, and the same HTTP host name, URL, method and parameter name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same parameter name. This could indicate a false positive (i.e. the parameter's actual behavior is different than the profiled behavior) or a wide spread attack.	session, web-user, application-id, source-ip
PCI Compliance - Card Track Data Detection	full	All profile violation alerts with the same destination Server Group, the same username and query are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over	session, source-ip, protocol,

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		an hour, or if 12 hours have passed since the alert was first generated.	server-group-id
PCI Compliance - Card Track Data Detection	partial	All profile violation alerts with the same destination Server Group, the same username are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	web-user, session, application-id, source-ip
Redundant HTTP Headers	full	All "Redundant HTTP Headers" violation alerts with the same destination Server Group, same source IP, same URL, and the same redundant headers are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	source-ip, application-id, session, web-user
Redundant HTTP Headers	partial	All "Redundant HTTP Headers" violation alerts with the same destination Server Group, and the same redundant header are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated by different sources with the same redundant header. This could indicate a false positive (i.e. the header can be redundant) or a wide spread attack.	server-ip, server-group-id, server-port, source-port, source-ip
Redundant HTTP Headers	partial	All "Redundant HTTP Headers" violation alerts with the same destination Server Group, and the same source IP are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on different URLS and with different redundant headers.	host, server-group-id, url
Redundant UTF-8 Encoding	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Redundant UTF-8 Encoding	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Required Parameter Not Found	full	All parameter-related profile violation alerts with the same source IP or source session, the same destination Server Group, and the same HTTP host name, URL, method and parameter name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id, sql-os-host
Required Parameter Not Found	partial	All parameter-related profile violation alerts with the same destination Server Group, and the same HTTP host name, URL, method and parameter name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	session, web-user, application-id, source-ip

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		This rule aggregates profile violation alerts generated by different sources but on the same parameter name. This could indicate a false positive (i.e. the parameter's actual behavior is different than the profiled behavior) or a wide spread attack.	
Required XML Element Not Found	full	All SOAP element-related profile violation alerts with the same source IP or source session, the same destination Server Group, and the same HTTP host name, URL, method and SOAP element name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	attack-id, source-ip, server-group-id, session, protocol
Required XML Element Not Found	partial	All SOAP element-related profile violation alerts with the same destination Server Group, and the same HTTP host name, URL, method and SOAP element name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same SOAP element name. This could indicate a false positive (i.e. the SOAP element's actual behavior is different than the profiled behavior) or a wide spread attack.	cookie-name, source-ip, application-id, session, web-user
Reuse of Expired Session's Cookie	full	All cookie-related profile violation alerts with the same source IP or source session, the same destination server group, and the same cookie name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	response-code, source-ip, session, server-group-id
Reuse of Expired Session's Cookie	partial	All cookie-related profile violation alerts with the same destination server group, and the same cookie name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same cookie name. This could indicate a false positive (i.e. the cookie is not traceable) or a widespread attack.	server-ip, table, server-group-id, source-ip, sql-os-user
Session Hijacking	full	All alerts of this type with the same source session and the same destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	protocol, source-ip, server-group-id
SOAP Access to a Non-SOAP URL	full	All Unauthorized SOAP Action violation alerts with the same destination Server Group, HTTP host name, URL, method, SOAP action, session ID and source IP are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates identical alerts.	server-group-id, server-ip, source-ip, sql-os-user
SOAP Access to a Non-SOAP URL	partial	All Unauthorized SOAP Action violation alerts with the same destination Server Group, and the same HTTP host name, URL, and SOAP action are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same URL and SOAP action. This could indicate a false positive (i.e. the SOAP action is allowed exists but is not part of the	session, method, source-ip, url, host, read-only-parameter, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		profile) or a wide spread attack.	
SOAP Element Value Length Violation	full	All SOAP element-related profile violation alerts with the same source IP or source session, the same destination Server Group, and the same HTTP host name, URL, method and SOAP element name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	attack-id, source-ip, server-group-id, session, protocol
SOAP Element Value Length Violation	partial	All SOAP element-related profile violation alerts with the same destination Server Group, and the same HTTP host name, URL, method and SOAP element name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same SOAP element name. This could indicate a false positive (i.e. the SOAP element's actual behavior is different than the profiled behavior) or a wide spread attack.	cookie-name, source-ip, application-id, session, web-user
SOAP Element Value Type Violation	full	All SOAP element-related profile violation alerts with the same source IP or source session, the same destination Server Group, and the same HTTP host name, URL, method and SOAP element name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	attack-id, source-ip, server-group-id, session, protocol
SOAP Element Value Type Violation	partial	All SOAP element-related profile violation alerts with the same destination Server Group, and the same HTTP host name, URL, method and SOAP element name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same SOAP element name. This could indicate a false positive (i.e. the SOAP element's actual behavior is different than the profiled behavior) or a wide spread attack.	cookie-name, source-ip, application-id, session, web-user
SQL Database access attempt - blocked user	full	Alerts invoked as a followed action of the same rule, with the same source Database User and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	response-code, server-group-id
SQL Database/HTTP access attempt - blocked IP address	full	Alerts with the same source IP and destination Application are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	url, host, server-group-id, method, source-ip, session
SQL injection	full	All SQL Injection violations with the same session ID, source-IP, user, destination application, URL and Parameter are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	method, host, soap-action, url, server-group-id, soap-parameter

Alert Name	Aggregation Type	Aggregation Description	Key Fields
SQL injection	partial	All SQL Injection violations with the same destination application, URL and parameter are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id
SQL injection	partial	All SQL Injection violations with the same session ID, source-IP, user and destination application are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	url, source-ip, method, host, soap-action, session, server-group-id
SQL protocol - invalid authentication	full	All SQL protocol violation alerts type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	session, source-ip, server-group-id
SQL protocol - invalid authentication	partial	SQL protocol alerts with the same source IP, destination Server Group, destination port and protocol are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	source-ip, user, query, server-group-id
SQL protocol - invalid flags	full	All SQL protocol violation alerts type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	session, source-ip, server-group-id
SQL protocol - invalid flags	partial	SQL protocol alerts with the same source IP, destination Server Group, destination port and protocol are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	source-ip, user, query, server-group-id
SQL protocol - invalid header	full	All SQL protocol violation alerts type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	session, source-ip, server-group-id
SQL protocol - invalid header	partial	SQL protocol alerts with the same source IP, destination Server Group, destination port and protocol are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	source-ip, user, query, server-group-id
SQL protocol - invalid header size	full	All SQL protocol violation alerts type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates	session, source-ip, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		violation alerts generated on the same socket. Only the first alert is presented below.	
SQL protocol - invalid header size	partial	SQL protocol alerts with the same source IP, destination Server Group, destination port and protocol are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	source-ip, user, query, server-group-id
SQL protocol - invalid length	full	All SQL protocol violation alerts type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	session, source-ip, server-group-id
SQL protocol - invalid length	partial	SQL protocol alerts with the same source IP, destination Server Group, destination port and protocol are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	source-ip, user, query, server-group-id
SQL protocol - invalid login	full	All SQL protocol violation alerts type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	session, source-ip, server-group-id
SQL protocol - invalid login	partial	SQL protocol alerts with the same source IP, destination Server Group, destination port and protocol are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	source-ip, user, query, server-group-id
SQL protocol - invalid message type	full	All SQL protocol violation alerts type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	session, source-ip, server-group-id
SQL protocol - invalid message type	partial	SQL protocol alerts with the same source IP, destination Server Group, destination port and protocol are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	source-ip, user, query, server-group-id
SQL protocol - invalid statement	full	All SQL protocol violation alerts type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates	session, source-ip, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		violation alerts generated on the same socket. Only the first alert is presented below.	
SQL protocol - invalid statement	partial	SQL protocol alerts with the same source IP, destination Server Group, destination port and protocol are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	source-ip, user, query, server-group-id
SQL protocol - unauthorized operation	full	All SQL protocol violation alerts type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	session, source-ip, server-group-id
SQL protocol - unauthorized operation	partial	SQL protocol alerts with the same source IP, destination Server Group, destination port and protocol are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	source-ip, user, query, server-group-id
Sql Signature Violation	full	All Signature alerts with the same source IP, destination Service, session ID, protocol and attack type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id, host, source-ip, url
Sql Signature Violation	partial	All Signature alerts with the same source IP, destination Service, protocol and session ID are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates Signature alerts with different signatures all coming from the same source (e.g. scanner).	server-port, protocol, server-group-id
Sql Signature Violation	partial	All Signature alerts with the same destination Service, attack type and protocol are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates similar Signature alerts generated by different sources. This type of aggregation could indicate a false positive or a widespread worm.	source-ip, server-group-id
SSL Untraceable Connection	full	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	host, server-group-id, method, url
SSL Untraceable Connection	partial	All SSL Untraceable alerts with the same destination Server Group and reason are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id, server-ip, sql-os-user, source-ip

Alert Name	Aggregation Type	Aggregation Description	Key Fields
Stream Signature Violation	full	All Signature alerts with the same source IP, destination Server Group, session ID, protocol and attack type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	url, method, server-group-id, source-ip, session, host
Stream Signature Violation	partial	All Signature alerts with the same destination Server Group, protocol and attack type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates similar Signature alerts generated by different sources. This type of aggregation could indicate a false positive or a widespread worm.	server-group-id, source-ip
Stream Signature Violation	partial	All Signature alerts with the same source IP, destination Server Group, session ID and protocol are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates Signature alerts with different signatures all coming from the same source (e.g. scanner).	protocol, session, service-id, source-ip
TCP - Bad State	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
TCP - Bad State	partial	All Firewall alerts with the same destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, source-ip, server-group-id
TCP - LAND Attack	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
TCP - LAND Attack	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
TCP - Micro-Fragment	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip

Alert Name	Aggregation Type	Aggregation Description	Key Fields
TCP - Micro-Fragment	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
TCP - Non Zero Reserved Flags	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
TCP - Non Zero Reserved Flags	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
TCP - SYN with Data	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
TCP - SYN with Data	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
TCP - SYN with FIN	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
TCP - SYN with FIN	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
TCP - SYN with Non Zero ACK Sequence	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip

Alert Name	Aggregation Type	Aggregation Description	Key Fields
TCP - SYN with Non Zero ACK Sequence	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
TCP - SYN with RST (not SYN ACK)	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
TCP - SYN with RST (not SYN ACK)	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
TCP - TCP Unexpected SYN	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
TCP - TCP Unexpected SYN	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
Time of Day Violation	full	All profile violation alerts with the same destination Server Group, the same username and query are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	session, source-ip, protocol, server-group-id
Time of Day Violation	partial	All profile violation alerts with the same destination Server Group, the same username are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	web-user, session, application-id, source-ip
Too Many Cookies in a Request	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Too Many Cookies in a Request	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different	session, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		sources.	
Too Many Headers per Request	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Too Many Headers per Request	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Too Many Headers per Response	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Too Many Headers per Response	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Too Many of the Same Response Code	full	All response code web correlation violation alerts with the same source IP, source session, destination Server Group and response code are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	source-ip, rule-id, server-group-id
Too Many of the Same Response Code	partial	All response code web correlation violation alerts with the same destination Server Group, and the same response code are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the response code.	session, server-group-id, header, source-ip, url
Too Many URL Parameters	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Too Many URL Parameters	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
UDP - Micro-Fragment	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
UDP - Micro-Fragment	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
Unauthorized Access to Service	full	All Firewall alerts with the same source IP, destination Server Group, destination port and protocol are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, query, source-application, server-group-id
Unauthorized Access to Service	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates Firewall alerts with different destination ports (for example a port scan).	protocol, attack-id, service-id
Unauthorized Access to Service	partial	All Firewall alerts with the same destination Server Group and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates Firewall alerts generated by the different sources but all to the same destination port.	server-group-id
Unauthorized Database Schema	full	All alerts of this type with the same source IP, OS username, destination Server Group and destination IP are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	server-group-id, server-ip
Unauthorized Database Schema	partial	All alerts of this type with the same source IP, OS username and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	server-group-id, attack-id, protocol
Unauthorized Database User	full	All profile violation alerts with the same destination Server Group, the same username and query are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	session, source-ip, protocol, server-group-id
Unauthorized Database User	partial	All profile violation alerts with the same destination Server Group, the same username are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different	web-user, session, application-id, source-ip

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		queries.	
Unauthorized Host	full	All alerts of this type with the same OS host name and the same destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	host, url, server-group-id, method
Unauthorized Method for Known URL	full	All URL-related profile violation alerts with the same source IP or source session, the same destination Server Group, and the same HTTP host name, URL, and method are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	application-id, cookie-name
Unauthorized Method for Known URL	partial	All URL-related profile violation alerts with the same destination Server Group, and the same HTTP host name, URL, and method are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same URL. This could indicate a false positive (i.e. the URL exists but is not part of the profile) or a wide spread attack.	server-group-id, session, source-ip
Unauthorized Method for Known URL	partial	All URL-related profile violation alerts with the same destination server group, and the same source IP or source session are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by the same source but on different URLs. This could indicate a false positive or a widespread attack.	server-group-id, user
Unauthorized OS User	full	All alerts of this type with the same OS username, the same destination Server Group and the same destination IP are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries	source-ip, server-group-id, server-port, protocol
Unauthorized Query	full	All profile violation alerts with the same destination Server Group, the same username and query are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	session, source-ip, protocol, server-group-id
Unauthorized Query	partial	All profile violation alerts with the same destination Server Group, the same username are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	web-user, session, application-id, source-ip
Unauthorized Query Group	full	All profile violation alerts with the same destination Server Group, the same username and query are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over	session, source-ip, protocol,

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		an hour, or if 12 hours have passed since the alert was first generated.	server-group-id
Unauthorized Query Group	partial	All profile violation alerts with the same destination Server Group, the same username are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	web-user, session, application-id, source-ip
Unauthorized Request Content Type	full	All "Unauthorized Content Type" violation alerts with the same destination Server Group, same source IP, same URL and the same content type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates identical alerts.	server-group-id, user
Unauthorized Request Content Type	partial	All "Unauthorized Content Type" violation alerts with the same destination Server Group, and the same source IP are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates alerts generated by the same source but on different URLs and using different content types.	session, server-group-id, source-ip
Unauthorized Request Content Type	partial	All "Unauthorized Content Type" violation alerts with the same destination Server Group, and the same content Type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates alerts generated by different sources on different URLs but with the same content-type. This could indicate a false positive (i.e. the content-type is required) or a wide spread attack.	reason, server-group-id
Unauthorized Sensitive Query	full	All profile violation alerts with the same destination Server Group, the same username and query are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	session, source-ip, protocol, server-group-id
Unauthorized Sensitive Query	partial	All profile violation alerts with the same destination Server Group, the same username are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	web-user, session, application-id, source-ip
Unauthorized Sensitive Query Group	full	All profile violation alerts with the same destination Server Group, the same username and query are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	session, source-ip, protocol, server-group-id
Unauthorized Sensitive Query Group	partial	All profile violation alerts with the same destination Server Group, the same username are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different	web-user, session, application-id, source-ip

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		queries.	
Unauthorized Sensitive Table	full	All profile violation alerts with the same destination Server Group, the same username and query are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	session, source-ip, protocol, server-group-id
Unauthorized Sensitive Table	partial	All profile violation alerts with the same destination Server Group, the same username are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	web-user, session, application-id, source-ip
Unauthorized SOAP Action	full	All Unauthorized SOAP Action violation alerts with the same destination Server Group, HTTP host name, URL, method, SOAP action, session ID and source IP are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates identical alerts.	server-group-id, server-ip, source-ip, sql-os-user
Unauthorized SOAP Action	partial	All Unauthorized SOAP Action violation alerts with the same destination Server Group, and the same HTTP host name, URL, and SOAP action are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same URL and SOAP action. This could indicate a false positive (i.e. the SOAP action is allowed exists but is not part of the profile) or a wide spread attack.	session, method, source-ip, url, host, read-only-parameter, server-group-id
Unauthorized Source Application	full	All Unauthorized Source Application violation alerts with the same destination Server Group, username, query and source application are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	host, url, server-group-id, source-ip, method
Unauthorized Source Application	partial	All Unauthorized Source Application violation alerts with the same destination Server Group, the same username, and the same source application are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	host, method, url, server-group-id
Unauthorized Source IP Address	full	All profile violation alerts with the same source IP, the same destination Server Group, the same username and query are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	source-ip, server-group-id, protocol, server-port
Unauthorized Source IP Address	partial	All profile violation alerts with the same source IP, the same destination Server Group, the same username are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that	application-id, source-ip

Alert Name	Aggregation Type	Aggregation Description	Key Fields
		involve different queries.	
Unauthorized Table/Operation Access	full	All profile violation alerts with the same destination Server Group, the same username and query are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	session, source-ip, protocol, server-group-id
Unauthorized Table/Operation Access	partial	All profile violation alerts with the same destination Server Group, the same username are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts that involve different queries.	web-user, session, application-id, source-ip
Unauthorized URL Access	full	All URL-related profile violation alerts with the same source IP or source session, the same destination Server Group, and the same HTTP host name, URL, and method are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	application-id, cookie-name
Unauthorized URL Access	partial	All URL-related profile violation alerts with the same destination server group, and the same source IP or source session are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by the same source but on different URLs. This could indicate a false positive or a widespread attack.	server-group-id, user
Unauthorized URL Access	partial	All URL-related profile violation alerts with the same destination Server Group, and the same HTTP host name, URL, and method are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same URL. This could indicate a false positive (i.e. the URL exists but is not part of the profile) or a wide spread attack.	server-group-id, session, source-ip
Unknown HTTP Request Method	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
Unknown HTTP Request Method	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id

Alert Name	Aggregation Type	Aggregation Description	Key Fields
Unknown IP Option	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip
Unknown IP Option	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
Unknown Parameter	full	All parameter-related profile violation alerts with the same source IP or source session, the same destination Server Group, and the same HTTP host name, URL, method and parameter name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id, sql-os-host
Unknown Parameter	partial	All parameter-related profile violation alerts with the same destination Server Group, and the same HTTP host name, URL, method and parameter name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same parameter name. This could indicate a false positive (i.e. the parameter's actual behavior is different than the profiled behavior) or a wide spread attack.	session, web-user, application-id, source-ip
Unknown SOAP Element	full	All SOAP element-related profile violation alerts with the same source IP or source session, the same destination Server Group, and the same HTTP host name, URL, method and SOAP element name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	attack-id, source-ip, server-group-id, session, protocol
Unknown SOAP Element	partial	All SOAP element-related profile violation alerts with the same destination Server Group, and the same HTTP host name, URL, method and SOAP element name are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates profile violation alerts generated by different sources but on the same SOAP element name. This could indicate a false positive (i.e. the SOAP element's actual behavior is different than the profiled behavior) or a wide spread attack.	cookie-name, source-ip, application-id, session, web-user
Unknown TCP Option	full	All alerts of this type with the same source IP, source port, destination Server Group, destination IP, and destination port are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates violation alerts generated on the same socket. Only the first alert is presented below.	service-id, source-ip

Alert Name	Aggregation Type	Aggregation Description	Key Fields
Unknown TCP Option	partial	All Firewall alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. Only the first alert is presented below.	server-group-id, header
Untraceable Database User	full	Alerts invoked with the same destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id, user, operation
URL is Above Root Directory	full	All protocol violation alerts with the same source IP or source session, the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	user, server-group-id
URL is Above Root Directory	partial	All protocol violation alerts with the same destination Server Group, and the same violation type are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates protocol violation alerts generated by different sources.	session, server-group-id
Web Worm	full	All Web Worm alerts with the same source IP, destination Server Group, HTTP host name, URL and HTTP method are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated.	server-group-id, session, source-ip
Web Worm	partial	All Web Worm alerts with the same destination Server Group and destination host name, URL and method are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates Worm alerts generated by different sources but all the same URL. This could indicate a false positive (i.e. the URL exists but is not part of the profile) or a wide spread worm.	server-group-id
Web Worm	partial	All Web Worm alerts with the same source IP and destination Server Group are aggregated into a single alert. The aggregation is stopped if no matching alert is generated for over an hour, or if 12 hours have passed since the alert was first generated. This rule aggregates Worm alerts with different URLs all coming from the same source (e.g. URL guessing).	url, session, server-group-id, content-type, source-ip

APPENDIX L

Troubleshooting Hashed Users

The following table lists reasons that SecureSphere displays user names as hashed users.

Reasons for Hashed User

Reason	Description
Unknown reason **SSL open mode** **SSL session reuse** **Unknown server SSL certificate** **Unsupported SSL cipher** **SSL temporary key** **Unsupported SSL legacy version** **SSL error**	SSL failure
No SSL data	No data was received from SSL, although an SSL process returned with no errors.
PDU mid packet	MSSQL PDU started in the middle of an SSL packet. This situation is not supported.
Unable to decrypt Kerberos	SecureSphere could not decrypt Kerberos authentication. The Kerberos key probably was missing or wrong.
Protocol error during SSL	Tabular Data Stream (TDS) error during SSL parsing.
Open mode during SSL	SSL is not supported for TCP sessions that were not detected by SecureSphere from the beginning of the session or when TCP gaps are detected.

APPENDIX M

Glossary

Action Interface Types

An action interface defines the type of action and the part of its configuration which is user configurable. Action interfaces can only be created by Administrators. The administrator defines the part of the set which is then no longer configurable by the user. The user can define the remaining parameters. For example, the administrator sets up an email and its contents, and the user fills in the actual email address of the recipients.

Each action interface is based on a predefined action type, which defines what the action performs. One of these types is OsCommand, allowing great flexibility for performing a variety of actions.

Adaptive Learning

SecureSphere has the ability to build the Dynamic Profile based on the process of seeing network traffic and learning the valid and legal use of the application. The process of seeing traffic and learning the legitimate use of the application is called "adaptive learning". See also "Dynamic Profiling".

ADC

Imperva's Application Defense Center (ADC) comprises security experts, conducting ongoing research based on multiple data sources. These include:

- Internal research of database implementations to discover security vulnerabilities (for example, several vulnerabilities discovered by the ADC have been acknowledged and patched by Oracle
- Industry discovered security vulnerabilities
- Patches released by vendors
- Security best practices developed for databases including consultation with industry standard bodies

ADC Content Update

The Application Defense Center (ADC), part of the MX management server, allows you to update and view signatures with their attributes and documentation, review pre-configured dictionaries, and define new dictionaries. Additionally, the ADC allows you to configure and update policies, assessment tests, table groups, and so on. The ability to Import objects enables continuous content updates to SecureSphere, supplied by the ADC. ADC packages include:

- Security Policies

- Assessment tests
- Additional predefined reports

ADC Insight Services

Imperva's ADC offers a range of services to ensure the most up-to-date protection and adherence to best practices. These include:

- Signature Updates
- Alert Profiles
- Audit Profiles
- Reports - generated using the reporting framework available to all users, these reports cover specific themes, such as: regulation related reports (SOX, HIPAA, PCI, and so on), or enterprise applications (Oracle EBS, and so on).

Alert

Collecting multiple events that have common characteristics (such as violation type, user and so on) into a single Alert. See also "Alert", Event", and Violation".

The MX Management server aggregates detailed security alert information from multiple gateways and presents them in a single consolidated view. Alerts are organized according to the type of violation (firewall, signature, profile, correlated, and so on) and contain detailed forensic information ranging from IP address to session ID. Advanced sorting and filtering technologies accelerate forensic investigation efforts.

Alert Aggregation

The SecureSphere system prevents alert storms (thousands of different warnings about what are essentially multiple occurrences of the same thing) by aggregating similar alerts into a single alert.

Alert Reports

Reports based on alerts as the data source. See also "Report", "Alert".

Anomaly Detection

Behavior-based anomaly detection compares a profile of all allowed application behavior to actual traffic. Any deviation from the profile is flagged as a potential attack. It is commonly referred to as a positive security model because it seeks only to identify all "known good" behaviors and assumes that everything else is bad. Behavior anomaly detection has the potential to detect attacks of all kinds – including "unknown" attacks on custom code. See also "Dynamic Profiling".

Appliance

The hardware and operating system that comprise a manageable device.

Application Awareness

The ability to analyze network traffic in the application context. For example, the Dynamic Profile is part of SecureSphere's application awareness. See also "Dynamic Profile".

Application Behavioral Layer

See also "Application Layer".

Application Defense Center

See ADC.

Application Group

A group of applications, which can facilitate the management of profiles and policies. You can work with application groups as either logical representations of the protected domain, or physical representation of sites.

Applications from different sites or server groups using different services can be grouped together to enable management according to the organizational structure
Administrative actions can be performed on these groups, for example:

- Managing the application profile and structure

Applying Security Policies

- Monitoring and reporting

Application Level

Analysis performed after the SecureSphere Gateway obtains the information that connects the network traffic to the web application or database application. This analysis will be relevant to the application, for example is the transaction permitted according to the application's dynamic profile. See also "Application", "DB Application", "Web Application", "DB Profile", "Web Profile".

Application Protocol

See also "Protocol".

Application Security

Security functionality aimed to protect attacks that are focused on application and protocol elements. See also "Protocol", "Application", "Web Application", "DB Application", "Dynamic Profile".

Application Signature

Signatures enforced on the application level. See also "Signatures", "Application Level".

APU

See also "Automatic Profile Update".

APU rule

A rule that indicates to SecureSphere when to apply APU i.e. when to update the profile.

Assessment

See also "DB Assessment".

Assessment Report

A SecureSphere report relating to the result of an assessment. See also "Report", "Assessment".

Assessment Test

A test run on the database server aimed to check if it is configured properly.

Attack

An attempt to subvert or bypass a system's security. Attacks may be passive or active. Active attacks attempt to alter or destroy data. Passive attacks try to intercept or read data without changing it. See also: brute-force attack, Denial of Service, hijacking, password attacks, password sniffing.

Attack Signatures

A signature identifying a specific attack. See also "Signature".

Attributes

Attributes are characteristics assigned to all files and directories. Attributes include: read-only, archive, hidden, or system.

Audit

See also "Auditing".

Audit analysis

The process of analyzing the SecureSphere audit - can be done by doing reports on audit or by looking at the audit information using audit view. See also "Audit View", "Auditing".

Audit Archive

The various SecureSphere gateways reserve a large portion of their disk space for storing the auditing data. However, due to the potentially enormous quantities of auditing data and the long periods of time over which they need to be stored, care needs to be taken to offload the auditing data from SecureSphere's local disks periodically according to specific needs (Archiving needs may differ from one organization to the next based on auditing requirements).

SecureSphere will write new event logs while overwriting the oldest ones stored in the local audit storage if the pre-allocated storage area is exceeded. The archiving capabilities extend your ability to store this data for long-term reference on media outside the SecureSphere system.

The archive process copies data from the SecureSphere system to a specified storage site, in the form of CSV files, which can be then consumed by other event correlation systems (that can consume CSV files).

Audit Policy

A policy the defines what database traffic should be audited. This policy as a match criteria that is based on correlation. See also "Match Criteria", "Policy", "Correlation".

Audit View

Indexing of DB Audit according to fields specified by the SecureSphere User. Applying this indexing on the audit data enables applying queries on the data that use these fields in queries. For example if a view is defined on the data that contains the "Source IP" field, it will be possible to see all audit data related to a specific source IP. See also "Audit Data", Audit Policy".

Auditing

See also "Database Auditing".

Automatic Profile Update

The ability of SecureSphere to detect that an application has changed and update the profile automatically based on this detection.

Background Scanning

Background scanning is a feature in some anti-virus software to automatically scan files and documents as they are created, opened, closed, or executed.

Bayesian Filter

A Bayesian filter is a program that uses Bayesian logic (also called Bayesian analysis) to evaluate the header and content of an incoming email message to determine the probability that it constitutes spam.

Black List

A list of objects that are marked as forbidden/negative/dangerous.

Bridge Mode

A deployment mode of the SecureSphere gateway where it is a transparent bridge on the network. This means that the appliance can be added to the network or removed from the network with no change to the networks configuration and definitions.

Brute Force

Brute force attacks use exhaustive trial and error methods in order to find legitimate authentication credentials.

Brute-force Attack

A brute-force attack is an attack in which each possible key or password is attempted until the correct one is found. See also: attack.

Cookie

A part of the HTTP protocol which is personal information stored in a Web user's computer.

Cookie Poisoning Attack

Cookie poisoning attacks involve the modification of the contents of a cookie in order to bypass security mechanisms. Using cookie poisoning attacks, attackers can gain unauthorized information about another user and steal his identity. See also "Cookie".

Correlated Attack

An attack built of more than one elements.

Correlation

Applying a match criteria that consists of several divers match elements. For example - "Target Table is employees and time of day is between 10am to 17pm".

Correlation Policy

A policy containing a correlation rule. See also "Policy", "Correlation Rule".

Correlation Rule

A rule that applies some action based on a correlation. See also "Correlation".

programs used to commit crimes on the Internet including identity theft and fraud. See also: malware.

Cross-Site Scripting

Cross-site scripting ('XSS' or 'CSS') is an attack that takes advantage of a Web site vulnerability in which the site displays content that includes un-sanitized user-provided data. For example, an attacker might place a hyperlink with an embedded malicious script into an online discussion forum. That purpose of the malicious script is to attack other forum users who happen to select the hyperlink. For example it could copy user cookies and then send those cookies to the attacker.

Data Definition Language

Data Definition Language AKA DDL - a term in the database world that refers to SQL statements that can be used either interactively or within programming language source code to define databases and their components.

Data Enrichment

The process of assigning customized attributes to events taking place in the network, for example extracting an SQL literal from a query or a field from HTTP traffic, then attaching this information to an event.

Data Monitoring Gateway

See SecureSphere Data Monitoring Gateway.

Data Scope

The data type and specification to be used for a report. An example of a data scope is "alerts from the last 3 days".

Data Security Gateway

See SecureSphere Data Security Gateway.

Database Agent

See SecureSphere Database Agent.

Database Assessment

See also "DB Assessment".

Database Auditing

Database auditing allows you to save database queries for audit purposes. For each database server group you can define an unlimited number of audit rules. The audit process runs in parallel with the mechanism that generates alerts, meaning that it sees all the queries, not just those generating alerts.
Bear in mind that the amount of information stored for auditing could be immense, which could affect system performance if prudent filtering is not employed. See also "Audit Policy".

Database Profile

The dynamic profile representing access of a DB application. See also "Dynamic Profiling", "DB application". The database profile contains all the tables of a DB application, and the way these tables are accessed - including allowed queries and users for these tables, as well as allowed operations on these tables.

DB Application

This term is specific for SecureSphere. It describes a collection of one or more pairs of database and schema. This collection describes a part of the database in charge of a specific function that the SecureSphere User would like to secure using SecureSphere.

DB Assessment

Part of the SecureSphere functionality. DB assessment is done by connecting to the target database, and performing a collection of assessment tests on it in order to evaluate if the DB server is configured in a secure way. The result of a DB assessment is a report indicating the result of each assessment test that was performed on the DB server. See also "Assessment Test", "Direct Access Information", "DB Server".

DB Audit

See also "auditing".

DB Profile

Acronym for database profile. See also "database profile".

DB Server

A server that runs a database, and the database is accessed from the network by users and applications.

DB Service

A service that describes communications to DB applications. See also "DB Application".

Default Policy

Every new object is assigned the relevant default policies. For example, the default firewall and network signatures are automatically applied to a new server group.

Deployment Guide

The two deployment guides (WAF and DxG) provide information to assist the SecureSphere administrator in the configuration and management of the initial stages of a SecureSphere deployment. See also "SecureSphere user", "SecureSphere system".

Deployment Topology

The network configuration into which the SecureSphere gateway is deployed This term usually refers to "Inline mode" and "Sniffing Mode". See also "SecureSphere System", "SecureSphere gateway", "Inline mode", "Sniffing mode".

Derived Rights

Rights that are granted to a user through the granting of roles or permissions.

Dictionary

A filter on the signature database, for example, a dictionary that includes all highly accurate, medium severity signatures for IIS 5 and 6. SecureSphere includes a set of predefined dictionaries.

Direct Access Information

Information that enables logging in to a server. This information contains user and password of the server. The information is used in order to log in to the remote server and perform an assessment on it. See also "Assessment".

DmG

Acronym for "Data Monitoring Gateway". See also "Data Monitoring Gateway".

DsG

Acronym for "Data Security Gateway". See also "Data Security Gateway".

DxG

DxG denotes DsG and DmG.

Dynamic Profiling™

Dynamic Profiling automatically examines live application and database traffic to create a comprehensive model (profile) of the structure and dynamics of the application and database. Valid application and database changes are automatically recognized and incorporated into the profile over time.

Dynamic Protect Mode

Once SecureSphere finished building the Dynamic Profile, it starts protecting the application based on this profile. The protection is made by passing the profile to "Protect Mode".

Event

A kind of record issued by the SecureSphere gateway describing an incident that has security implications such as viruses or unauthorized access.

Fail Close(d)

A mode of the SecureSphere gateway where if the gateway does not function (for example because of a power outage) the machine will not pass any traffic so that the protected domain will not be exposed to attacks.

Fail Open

A mode of the SecureSphere gateway where if the gateway does not function (for example because of a power outage) the machine keeps passing traffic so that the connectivity to the protected domain is not harmed.

False Negative

A failure to identify a violation.

False Positive

An erroneous identification of an event as a violation when it is in fact not a violation.

Filtered Dictionary

A signature dictionary that has a filtering condition applied on it. For example a dictionary of signatures that are related to web with a filter of only the signatures that are relevant for the web server of type IIS.

Followed Action

An action taken by SecureSphere after a specified event is detected.

Gateway Group

An defined collection of SecureSphere gateway object that is defined in the MX Server as a group. SecureSphere gateways are grouped when all of the members of the group protect the same server group.

Gateway Management Screen

A screen in the SecureSphere administration interface on the MX server where SecureSPhere gateways can be added or removed. In this screen the status of each SecureSphere gateway is displayed (status includes uptime, CPU consumption and so on). See also "MX Server", "SecureSphere Gateway", "SecureSphere administration interface".

Global Object

The SecureSphere environment is hierarchical. Most objects in the system can be grouped, and then referenced as global objects from policies, configuration, and so on.

Global Object Examples: Table groups, Stored procedures, Signatures and dictionaries.

The granularity supplied by SecureSphere requires that many of the system's components be approachable from anywhere in the system, i.e., global.

GW group

See also "Gateway group".

Host

An individual computer, sometimes used to mean a computer on which specific software is physically located or which provides a specific service.

Implementation Life Cycle

The product implementation life cycle is a series of ongoing refinements divided into four phases (identify risks, set policies & controls, monitor and protect, measure). The iterations through the cycle are driven by the changing environment and reflected in SecureSphere's dynamic learning capabilities, as well as various feedback that the system provides to the administrators and auditors.

When the system has acquired sufficient data, the learned behaviors are automatically enforced, and events are generated for any occurrence that does not follow the learned behavior. Periodically, users need to verify that the learned behaviors are in fact the desired ones. Users can view these events and refine the actions to be taken for them.

Inline Mode

One of the deployment modes of SecureSphere gateway. In this mode the SecureSphere acts as a firewall device on the network. It servers as one of the devices processing traffic on its way to the data center. This mode is usually used when it is desired that the SecureSphere gateway will block malicious or unauthorized network access to the data center.

Kick and Disable

After applying "kick user", disabling future logins of the same user. See also "Kick User".

Kick User

Terminate a session of a SecureSphere user who is using the administrative interface of the MX server. See also "SecureSphere User", "Administrative Interface", "MX Server".

Known Attacks

The term "known attack" is frequently used to refer to attacks that target previously known vulnerabilities in commercial or open source application software such as IIS, Apache, Oracle, and so on Hundreds of vulnerabilities in such software are found and made public (www.cert.org, and so on) each year. Hackers use this information to construct attacks. Examples of such attacks include Code Red, Nimda, and Spida.

License Summary

The page in the SecureSphere administrative interface that indicates the status of the SecureSphere software license.

Match Criteria

A collection of rules and conditions with relationship between them that once matched enable a specific action. An example of a match criteria is: Source IP is 1.1.1.1, and the target URL is www.imperva.com.

Monitoring

The SecureSphere Monitoring tab collects and displays the system's recorded activity, including the following information:

- Dashboard: Constantly updated, easy to understand snapshot views of the gateways, server groups, system's load, connections per second, latest alerts, attack reports, and system events.
- Alerts: Detailed information on every alert.
- System Events: The SecureSphere Management Server log comprises information on each change to the product configuration, and important system events.
- DB Audit – on-the-fly view
- View archived audit data

Monitoring Audit Events

Audit results can be viewed using the SecureSphere® GUI. The audit rule configuration lets you define up to 5 different audit views, each comprising between 1 and 4 aggregation keys (selected from: source IP address, database username, source OS username/hostname, destination table name, source applications, database operation, and exception occurred).
The SecureSphere system provides a data management mechanism for effective management and analysis of the audit results (potentially immense quantities of data). For greatest efficiency, events are stored sequentially in a flat file as they occur, and indexed to an index file according to the preconfigured aggregation keys. Audit views are then generated based on a combination of the flat file and index file.

MX Appliance

The SecureSphere appliance that has the MX server in it. See also "MX Server", "SecureSphere Appliance".

MX Management Server

See also "MX Server".

MX Server

This is the components in the SecureSphere system that performs the management function. This includes:

1. Dispatching the security policy to the SecureSphere gateways
2. Getting the status of the gateway appliances
3. Getting alerts and system events from the gateways
4. When the SecureSphere administrator wishes to configure the SecureSphere system, changes policies and so on, he should communicate with the MX server in order to do so. Once the system is configured the configuration changes are dispatched to the SecureSphere gateways by the MX server.
5. When the SecureSphere user wishes to view security alerts or system events he should communicate with the MX server, where the alerts and system events from all of the gateways will be displayed

Network Security Layer

The functional part of the SecureSphere Security Engine that analyzes traffic on the network level and issues violation on network related attacks, such as malformed IP packets. This is the initial analysis of the traffic before the service level analysis and the application level analysis. See also "Network Level", "Service Level", "Application Level".

Parameter Tampering

Attack that is based on HTTP parameters manipulation. Parameter tampering is a simple attack targeting the application business logic. This attack takes advantage of the fact that many programmers rely on hidden or fixed fields (such as a hidden tag in a form or a parameter in a URL) as the only security measure for certain operations. Attackers can easily modify these parameters to bypass the security mechanisms that rely on them.

PCI

A compliance standard created maintained be credit card companies such as VISA and that mandates securing card holder data (i.e. credit card information) in merchants' web sites/data centers. See also "Compliance".

Peer-to-peer (P2P) networking

Peer-to-peer (P2P) networking is a distributed system of file sharing where any PC on the network can see any other PC on the network. Users access each others' hard drives to download files. This type of file sharing is valuable, but it brings up copyright issues for music, movies, and other shared media files. Users are also vulnerable to viruses, Trojans, and spyware hiding in files. See also: Trojan horse, spyware.

Permissions Tab

The screen in the SecureSphere administration interface where the operations each SecureSphere User may perform are defined. See also "SecureSphere administration interface", "SecureSphere User".

Policy

A set of one or more rules related to a functional area in the system. A policy may contain a match criteria. A policy may contain instructions on how to handle one or more violations - examples of such instructions would be if an alert should be issued and if a user should be blocked by the SecureSphere gateway. See also "Violation", "Match Criteria".

Predefined Correlation Policies

A collection of correlation policies that are built in to SecureSphere and provide out-of-the-box protections against certain attacks.

Privileged Operation

Privileged operations are special highly sensitive operations on a database such as database creation and deletion, user creation and so on They can be defined as any operations other than basic data manipulation (i.e., update, select, insert, and delete), either directly on the database, or via regular stored procedures.

Profile Components

The entities that comprise the Dynamic profile. For example for DB profile the profile components will be tables, operations, users and queries. See also "Dynamic Profile", "DB Profile".

Profile Violation

A violation issued as a result of traffic that contradicts the information in the dynamic profile. See also "Dynamic Profile", "Violation".

Protected Network

Server groups (SGs) comprise all the servers that the SecureSphere® system protects. The server group is a basic reference unit, corresponding to a server or a cluster of servers (i.e., the system does not protect individual IP addresses). A server group is defined as a layer 3 (IP) entity.

Protocol

A standard defining how a client and a server communicate with each other. A protocol defines a set of messages passed between them, states and message order.

Protocol Violation

A violation issued by the SecureSphere Gateway as a result of incorrect protocol between the client and the server. See also "Protocol", "SecureSphere Gateway", "Violation".

Protocol Violation Policies

SecureSphere® protocol compliance checks ensure that HTTP and SQL protocols meet RFC and expected usage requirements. By ensuring that the HTTP and SQL protocols meets guidelines, protocol compliance prevents attacks on both known and unknown vulnerabilities in commercial web and DB server implementations.
Imperva has conducted comprehensive research and collected a group of protocol violations that usually indicate attack attempts. You can enable or disable each of these violations for each group of protected servers.
Protocol violation policies are relevant to both HTTP (web servers) and SQL databases.

Raw Data

The data as extracted from the network traffic.

Redirect

A redirect is an action used by some viruses to point a command to a different location. Often this different location is the address of the virus and not the original file or application.

Regulatory Compliance

See also "Compliance".

Remote Agent

In SecureSphere this usually refers to the DB Agent. See also "DB Agent".

Rename

A rename is an action by which a user or program assigns a new name to a file. Viruses may rename program files and take the name of the file so that running the program inadvertently runs the virus. Anti-virus programs may rename infected files, making them unusable until they are manually cleaned or deleted.

Reverse Proxy

A web proxy that protects one or more web servers. All HTTP connections are opened to the reverse proxy. Once the reverse proxy validates the connection from the client it opens a separate connection to the server and serves the client connection.

Reverse Proxy Mode

A deployment mode of the SecureSphere gateway where it is a reverse proxy. See also "Reverse Proxy"

Rogue Program

A rogue program is a term the media uses to denote any program intended to damage programs or data, or to breach a system's security. It includes Trojan horse programs, logic bombs, and viruses.

Scuba

Scuba is a name of a tool by Imperva that does assessment on the security of databases.

SecureSphere

SecureSphere is the name of the product line dealt with in this glossary. This product contains and delivers the overall security solutions for security and compliance of data-centers both in the database and in the web functional areas. The SecureSphere product line includes the SecureSphere WAF for web and the SecureSphere DxG for database security and compliance.

SecureSphere Administration Interface

A central component in the MX server that allows the SecureSphere users to connect to the MX Server and perform the user management function of SecureSphere. This communication is done technically using web - the SecureSphere user uses a web browser, and the MX server has a web server that gets the requests from the SecureSphere user and performs the appropriate management functions. See also "SecureSphere User", "MX Server".

SecureSphere Appliance

The appliance that runs the SecureSphere software. The SecureSphere appliance is the hardware unit purchased by the Imperva customer in order to use the SecureSphere product line.

SecureSphere Data Monitoring Gateway

This is one of the products in the SecureSphere product line. It has similar function to the DsG, only that it does not block traffic and is generally used to monitor access to the database and not to block unauthorized access. This product includes the SecureSphere WAF capability. The product is deployed in Sniffing Mode (See also "DsG", "Sniffing Mode").

SecureSphere Data Security Gateway

This is one of the products in the SecureSphere product line. This product secures databases by inspecting the database transactions, blocking attacks and malicious traffic, blocking and alerting on unauthorized access to the database and includes various reporting capabilities used for compliance. This product includes the SecureSphere WAF capability. The product is deployed in Inline Mode (See also "DsG", "Inline Mode").

SecureSphere Database Agent

A component that is part of the SecureSphere System. The SecureSphere Database Agent is a software component that is installed on the customer's database server, monitors database traffic and sends it to the SecureSphere gateway for further analysis. It is used when there is access to the database server that cannot be monitors directly by the SecureSphere gateway (for example when the DBA logs into the database server on an encrypted channel that is not exposed to the SecureSphere gateway).

SecureSphere DB Agent

See also "SecureSphere Database Agent".

SecureSphere Gateway

This is the component in the SecureSphere system that inspects and analyzes network traffic to the database servers and web servers. It detects malicious traffic and blocks it, and also reports to the management server (see MX server) on both attacks and legitimate traffic for further analysis and compliance.

SecureSphere Management Server

See MX Server.

SecureSphere Security Engine

The SecureSphere security engine is the sum of the overall capabilities of the SecureSphere gateways to analyze traffic and apply various security functions on it, such as blocking, alerting, and auditing (See also "Audit", "SecureSphere Gateway").

SecureSphere Signature Database

A large collection of signatures that is managed and developed by Imperva's ADC. It is used to identify alert and block on viruses, worms and other attacks (See also "ADC", "Worm", "Virus", "Signature Policy".

SecureSphere System

The collection of components that make up SecureSphere. The SecureSphere system is made up of a management server (see MX Server) and one or more SecureSphere gateways (see SecureSphere gateway).

SecureSphere User

The person manages the SecureSphere product when deployed in the customer environment. This person does the management function by handling the SecureSphere system. The SecureSphere Users are divided to "Administrator Users" who are privileged users that have full access to all of the management functions, and users who are not "Administrator Users". These users have only partial privileges and perform a more limited/specific management function.

Security Policy

A security policy is a SecureSphere policy aimed to define security conditions and define SecureSphere's reaction to these conditions (such as blocking a user). See also "Policy".

Server Group

A collection of servers that run identical web applications or DB applications on similar services. See also "Web Application", "DB Application", "Service".

Service

A service is an object that is part of the SecureSphere MX Server. It describes how the application communicates to the world and thus gives the SecureSphere gateway the information required in order to track the traffic that goes to the application and understand it so that the Security Engine can be applied on it. Example of information in the Service is encryption information, language oriented information, ports. See also "Port", "Application", "Security Engine".

Service Level

Analysis that is done once the SecureSphere Gateway analyses the protocol entity related to the network traffic. An example of service level enforcement is the validity of protocol messages. See also "Protocol".

Session

See also "Web Session"

See also "SecureSphere Signature Database"

Signature Detection

Detection of attacks based on signature match. See also "Signature", "ADC"

Signature Policy

A policy indicating what needs to be done when a signature violation is issued by the SecureSphere gateway. See also "Violation", "Signature", "SecureSphere gateway"

Signatures

Signatures are text strings that match known server vulnerabilities and attack patterns. The SecureSphere® system maintains a list of over 2500 signatures based on the Snort database and Imperva's Application Defense Center (ADC). The Imperva signature database also comprises signatures hand-crafted by the ADC to detect sophisticated application-level attacks. See also "Snort", "ADC", Regular Expression".

Signing Policy

The set of instructions indicating if specific information that is exported by SecureSphere should be digitally signed, and with which key.

Simulation mode

A mode of the SecureSphere gateway where attacks are detected and alerts are issued, but traffic is never dropped. This mode is used in order to verify that SecureSphere does not generate false positives.

Site

An object on the MX Server that represents a geographic entity comprising any number of server groups (the virtual entities protected by SecureSphere®). Every server group belongs to exactly one site (a default site exists after installation). This object is very useful for administration purposes.
See also "MX Server", "Server Group"

Site Hierarchy

SecureSphere objects that describe the structure of the protected domain. They describe the server groups, services and applications that are in the protected domain and the relations between them. See also "Protected Domain", "Web Application", "DB Application".

Sniffer

A sniffer is a software program that monitors network traffic. Hackers use sniffers to capture data transmitted over a network.

Sniffing

Sniffing happens when traffic that traverses between 2 network devices is copied to a third device for analysis purposes, The device performing the sniffing is passive i.e. it does not take an active part in the communication.

Sniffing Mode

One of the deployment modes of SecureSphere gateway. In this mode the SecureSphere gateway acts as a sniffing device on the network. The gateway analyzes the sniffed traffic and thus performs a monitoring function (See also "Sniffing", "DmG", "SecureSphere gateway").

Snort

A public database of signatures (not maintained by Imperva). See also "Signatures", "Imperva"
Snort is a free software network intrusion detection and prevention system capable of performing packet logging and real-time traffic analysis, on IP networks. Snort was written by Martin Roesch but is now owned and developed by Sourcefire, of which Roesch is the founder and current CTO.

SOAP

Simple Object Access Protocol. A protocol that describes a model for packing XML enquiries and responses.

SOAP Action

SecureSphere® automatically identifies URLs comprising XML and SOAP content. For each SOAP/XML URL, SecureSphere presents all the SOAP actions that have been learned for this URL. If the URL does not include any SOAP actions but does contain XML, SecureSphere shows this as the Default SOAP action. When the URL is switched to protect mode, SecureSphere invokes the Unauthorized SOAP Action violation whenever someone attempts to access the URL with an unauthorized SOAP action. See also "SOAP".

SOX (Sarbanes-Oxley)

A compliance standard - an act passed by US Congress to protect investors from the possibility of fraudulent accounting activities by corporations. According to SOX both management and the external auditor are responsible for performing assessments and having controls for the appropriate function of the corporation See also "Compliance".

SQL Connection User Tracking

The SQL Connection User Tracking allows SecureSphere to extract the application user from SQL queries by analyzing the queries. See also "UUT".

SQL Injection

SQL injection is an attack that is used to take advantage of non-validated input vulnerabilities to pass SQL commands through a Web application for execution by a backend database. Attackers take advantage of the fact that programmers often chain together SQL commands with user-provided parameters, and can therefore embed SQL commands inside these parameters. The result is that the attacker can execute arbitrary SQL queries and/or commands on the backend database server through the Web application.

Stored Procedures

Stored procedures are scripts or code stored on the database that when called (externally by their names), perform operations on the data stored on the database. See also "DB Server".

String

A string is a consecutive series of letters, numbers, and other characters. "afsH(*&@~" is a string; so is "The Mad Hatter." Anti-virus applications often use specific strings, called virus signatures, to detect viruses. See also: signature.

Susceptible Directory

Susceptible Directories are directories on a web server that are prone to attacks by web worms.

System Boot Record

See: Boot record.

System Event

One of the message types that is issued by SecureSphere. It has an informative nature and deals with the internal state of the system. Examples of system events are: changes to objects in the SecureSphere system that are made by administrators as part of the configuration process. Another example is messages dealing with the CPU consumption of the SecureSphere appliances.

System Objects

Objects that are created as an integral part of a database. For example, database schemas, tables, columns, or keys.

Table and Operations List

The tables and operations list is simply a learned list of all the tables and the operations performed on them. This list is part of the DB Profile. See also "DB Profile".

Timestamp

The timestamp is the time of creation or last modification recorded on a file or another object. Users can usually find the timestamp in the Properties section of a file.

TOM

TOM stands for Top of Memory. It is a design limitation at the 640kb mark on most PCs. Often the boot record does not completely reach top of memory, thus leaving empty space. Boot-sector infectors often try to conceal themselves by hiding around the top of memory. Checking the top of memory value for changes can help detect a virus, though there are also non-viral reasons this value changes.

Transparent Inspection

Imperva's Transparent Inspection technology delivers multi-gigabit performance, sub-millisecond latency, and options for high availability that meet the requirements of the most demanding application and database environment. Transparent Inspection enables SecureSphere to inspect network traffic at the highest level of security while having a minimal effect on performance and it is very easy to deploy.

Triggered Event

A triggered event is an action built into a virus that is set off by a specific condition. Examples include a message displayed on a specific date or reformatting a hard drive after the 10th execution of a program.

Unauthorized Database User

Access to the database by a user that is not included in the DB Profile. See also "DB Profile".

Unauthorized SOAP Action

A SOAP Action that is not included in the web profile. See also "Web Profile".

Has the following form: \\server\resource-pathname\subfolder\filename.

Unidirectional Tunnel

The SecureSphere DB agent captures all the database traffic throughout the monitoring process, and dispatches every packet to a chosen SecureSphere gateway via a unidirectional tunnel. The agent only uses this tunnel to send these audit records to a remote SecureSphere database gateway - not to listen to traffic from the gateway. The gateway inspects these records just like regular traffic passing through it. However, due to the remote nature of the DB agent (i.e., the traffic is not monitored in real time), SecureSphere does not block/terminate suspicious activity (using TCP reset) detected in the traffic sent by the agent.

Universal User Tracking

This is the SecureSphere's ability to detect which user runs each web and database transaction. This is a centric capability of SecureSphere that is comprised of a collection of technologies for web user detection and database user detection. See also "SQL Connection User Tracking", "web application user tracking".

Unknown Attacks

The term "unknown attack", when used in the context of application security, generally refers to attacks that target previously undocumented vulnerabilities in custom developed enterprise Web application code. Based on penetration testing of over 250 private and public sector penetration tests, Imperva's Application Defense Center has concluded that 92% of Web applications are vulnerable to targeted attacks on internally developed code.

URL Rewrite Group

URL Rewrite Container - The collection of URL rewrite rules to be applied on a web service (Reverse Proxy mode only).

User Enrichment

The process of enriching event data with information about users from an outside source, for example Active Directory.

User Mapping

The process of mapping between Active Directory users and database users.

User Tracking

See also "Web Application UT".

UUT

Acronym for Universal User Tracking - see Universal User Tracking.

Violation

In SecureSphere a violation is a situation where the analysis of the network traffic indicates that the traffic is not right/illegal/not as would be expected under normal conditions.

WAF

Acronym for Web Application FireWall. See also "Web Application FireWall"

Warm Boot

Warm booting is restarting a computer without first turning off the power. Using CTL+ALT+DEL or the reset button on many computers can warm boot a machine. See also: cold boot, reset.

Web Application

A web application is an application that runs on a web server and is accessed from the internet using the HTTP protocol (web). Several web applications can run on a single web server. Each application is identified by its own specific HTTP header called "host" header. In SecureSphere a web application is identified by the HTTP "host" header on a specific web server.

Web Application Firewall

The WAF protects web applications (malicious activity detection & prevention, overall application monitoring, reporting). Offers unified protection (application logic, worms and platform attacks, network attacks). Uses automated security policy (dynamic profiling models application structure and dynamics, adapts when application changes). See also "FireWall".

Web Application UT

Web Application User Tracking is a mechanism which automatically and transparently acquires/discovers the identity of an application user as they login. This mechanism relies on the fact that most web applications use web forms to collect the user identity that initializes the authentication process. This can be accessed via a single login URL, or more than one URL may be available per application.

The SecureSphere application user tracking mechanism identifies these forms and extracts the user identity (username), based on the responses sent back from the user by means of action URLs (in most cases, multiple action URLs relates to every login URL).

SecureSphere also extracts the results (success or failure) of the login (authentication) process by tracking the response from the application server. SecureSphere analyzes three indication types to figure whether the login process was successful or not:

- Redirect URLs, in which case different URLs are used for success or failure
- Response code, explicitly indicating success or failure
- Other page types used as a response to the user, in which case, SecureSphere searches for a text pattern which indicates success or failure of the login process.

More than one type of a response may come back from the server, each of which may include multiple indications. To cope with this, SecureSphere uses a sequential rule base to make a decision about every response. Every rule in this rule base refers to one indication, which could appear in the response. Once a match to a rule is found, SecureSphere determines if the login attempt failed or succeeded based on this rule.

SecureSphere then uses the user identity and the login process results to enrich its enforcement mechanism and its alerting and auditing system.

Web Profile

The dynamic profile representing access of a web application. See also "Dynamic Profiling", "web application". The web profile contains all the URLs of a web application, and the way these URLs are accessed, including parameters and HTTP methods.

XSS

See also "Cross-Site Scripting"

SSL Strip

SSL Strip is an attack that basically does not involve the target application. Here is how it works:

- A victim uses a browser to access an application by manually typing the application's address in the browser's address bar (e.g. www.mybank.com)
- The browser's default behavior is to use HTTP protocol to carry this initial request (and in fact most users who bother would themselves type (http://www.mybank.com).
- The attacker has already launched a DNS related attack that would make the victim's computer send the request to an attacker controlled server rather than to the real server.
- Since the request is an HTTP request and not an SSL request, the attacker can launch a man-in-the-middle attack without making the browser aware of it (if the connection was an SSL connection a browser warning would have popped up).
- From that moment on the attacker has complete control over the session and can either use this for merely credential theft (much like very simple phishing attacks) or control the entire session between the victim and the target application.
- It has nothing to do with signatures, the victim does not necessarily interact with the target application during the attack, nor does the attacker.

Index

A

About Event Types • 434, 571

About Rule.dns • 549

Absolute Modifiers • 496, 497

Access Patterns • 260, 280, 281

Accurate Protection • 23

Action Interface Types • 349

Action Set Main Tasks • 347, 361

Action Sets and Followed Actions Overview • 346

Action URL Details Pane Parameters • 306, 307, 308

Activating Settings • 34, 51, 53, 55, 61, 63, 65, 71, 74, 75, 76, 77, 149, 192, 194, 211, 213, 223, 224, 264, 272, 274, 275, 276, 277, 278, 279, 280, 290, 293, 296, 298, 305, 342, 344, 345, 349, 366, 368, 372, 376, 377, 388, 389, 390, 391, 394, 397, 400, 403, 406, 407, 413, 414, 415, 418, 436, 441, 443, 444, 445, 446, 447, 450, 451, 454, 455, 460, 461, 462, 505, 645

Active Module - Change Tracking Details • 616, 647

Active Module - Change Tracking Details Report Columns • 648

Active Module Criteria • 616

Active Module Pre-configured Objects • 616, 617, 648

Active Module Preconfigured Objects Report Columns • 648

Active Module Reports • 666

Active Module Track Value Changes • 616, 618, 648

Active Module Track Value Changes Configuration • 616, 618, 648

Active Module Track Value Changes Configuration Report Columns • 650

Active Module Track Value Changes Report Columns • 649

ADC • 27

Adding / Removing Columns from the Alerts and Violations Tables • 95, 97, 107

Adding a Comment to a Vulnerability • 176

Adding a New Web Application • 64, 71

Adding a Signature to a Dictionary • 329, 332

Adding Comments to Security Policies • 196

Adding Data Pattern Definitions and Validation Method to Dictionary Groups • 370, 371, 381

Adding IP Addresses and Assigning to a Gateway Group • 52, 53, 57

Adding New Security Policies • 74, 189, 192, 290, 481

Adding Parameters to a Data Masking List • 375

Adding Protected IP Addresses Individually • 54, 55

Adding Session Identifiers Used for Session Tracking • 67, 77

Adding SSL Keys • 61, 66, 75, 484, 487

Adding Tokens to a Session Tracking List • 450, 451

Adding URL Prefixes/Directories Groups to Web Applications • 444

Advanced Configuration Overview • 337

Advanced HTTP Protocol Validation Policy • 206, 207, 208

Advanced SecureSphere Configuration • 45, 336

Alert Aggregation Descriptions • 100, 671

Alert and Event Notification Capacity • 89

Alert and Violation Criteria • 232, 599

Alert Event Available Columns • 636, 638

Alert Management Workflow • 95

Alert Report Available Columns • 636, 637

Alert Reports • 662

All Services • 126

Alternatives • 500

Analyzing Aggregated Alerts • 99, 100, 108

Analyzing Alerts in the Details Pane • 93, 98
Analyzing Discovered Servers • 33, 38, 41
Analyzing Discovered Vulnerabilities • 162
Analyzing General Vulnerability Details • 169
Analyzing Risk • 145, 147, 154
Analyzing Risk Details • 155, 158, 160
Analyzing Risk Workflow • 155
Analyzing Violations in the Details Pane • 98, 106, 108
Analyzing Vulnerability Details • 164, 168
Anchors • 500
Anonymous Proxy Service • 121, 128
Anonymous Proxy Summary • 129
Anti Scraping Policy • 182, 219, 221
Anti-Automation Policies • 219
Application Structure Modeling • 407
Applying a Dictionary to a Policy • 329, 333
Applying a URL Rewrite Rule Group to a Web Service • 454, 455
Applying Malware Detection Policies • 463, 471, 477
Applying Policies • 189, 191, 193, 194, 290, 337, 395, 449, 481
Applying Policies to the Application Tree • 342
Applying Recommended Malware Detection Policies • 477, 478, 480
Applying Security Policies to the Objects in the Applications Tree • 339
Applying Security Policies to the Objects in the Sites Tree • 149, 338
ArcSight Placeholders and Syntax • 415, 559
Assessment • 27
Assessment Criteria • 601
Assessment Results Filter Criteria • 610
Assigning a Vulnerability an Owner • 164, 171, 174
Assigning Followed Actions • 347, 348, 360, 410, 414, 421, 423
Assigning Roles to External Roles • 425, 431
Associating a Plugin with a Service and Configuring the Plugin • 503, 504
Associating a Session Tracking List to a Web Service • 450, 451
Associating a Web Error Page Policy to a Web Service • 445, 446
Associating a Web Service with a Database Service • 313
Associating an SQL User Tracking Group with a Database Service • 317, 322
Associating Client Certificate Validation • 459, 461, 484
Associating Web Service Data Objects • 439, 440, 441
Attaching a Data Masking List to a Service • 370, 375, 376, 381
Authentication and Authorization Configuration for Active Directory • 425, 426
Authentication Parameters in POST/XML • 310
Automatically Adjusting the Web Profile • 282, 287
Available Licenses for SecureSphere Web Security Products • 23
Avoiding Infinite Loops • 453

B

Basic Configuration Guidelines for Gateways, Management Servers and Agents • 46
Basic Configuration Overview • 45
Basic Filter Criteria • 252
Basic SecureSphere Configuration • 44
Basic Server Group Configuration • 45, 51, 56
Basic Server Group Configuration Tasks • 52
Basic Site Configuration • 45, 50
Basic Web Service Configuration Tasks • 60, 71, 365
Basic Web Services Configuration • 45, 52, 60
Before Running a Change Tracking - External Objects Policy • 404
Bot Mitigation Policy • 182, 219, 220
Brackets • 499
Broken Links and References • 260, 265
Browser Support • 45

C

CEF Fields • 559, 561
Certificate-Based User Authentication • 303
Change Tracking - External Objects • 403, 404
Change Tracking - External Objects - Policy Configuration • 404, 405
Change Tracking - External Objects Policy Permissions • 404
Changing Passwords • 412, 413
Changing the Malware Detection Feed User Password • 465, 467
Changing the Threat Level of a Malware Type • 475, 476
Changing Vulnerability Status • 171, 173
Character classes • 500
CIDR Notation • 48, 368
Cloning/Creating New Policies • 192, 223
Columns Available in Active Module Reports • 647
Columns Available in Alert Reports • 232, 636
Columns Available in Assessment Reports • 639
Columns Available in Configuration Reports • 645
Columns Available in Discovery Reports • 651
Columns Available in Profile Reports • 238, 655
Columns Available in System Event Reports • 233, 641
Columns Available in Task Reports • 644
Columns Available in Vulnerability Reports • 641
Community Defense Back Feed • 121, 133, 135
Community Defense License • 121, 134, 141
Completing Assignment Tasks • 411, 412
Completing Review Tasks • 410, 411
Conducting a Search in the Risk Explorer • 158, 160
Conducting a Web Scan and Preparing Results • 147, 148, 149
Configuration - Sites Report Criteria • 607, 608
Configuration Report Criteria • 607
Configuration Reports - Security Policies • 647
Configuration Reports - Sites • 646
Configuration Security Policy Report Criteria • 607, 609
Configuring a Kernel (Non-Transparent) Reverse Proxy • 483
Configuring a Lookup Data Set • 383, 386, 387, 388, 391
Configuring a Service Discovery Scan • 30, 32, 33, 36
Configuring a Transparent Reverse Proxy • 486
Configuring Action Interface Parameters • 250, 346, 347, 348, 349, 351, 360, 410, 414, 458
Configuring Action Sets and Followed Actions • 77, 149, 345
Configuring Active Directory Integration for Data Enrichment • 378, 379, 425
Configuring Advanced Web Service Settings • 68, 337, 438, 447, 629, 633
Configuring an SQL User Tracking Group • 317, 319, 322
Configuring and Applying Archive Settings • 343, 344, 406
Configuring Certificate Authorities • 364, 459
Configuring Certificate Authority Groups • 364, 459, 460, 462
Configuring Certificate Revocation List (CRL) • 364, 459, 460
Configuring Client Authentication Rules • 364, 459, 461, 462, 485
Configuring Credentials • 35
Configuring Data Analysis View • 242, 246
Configuring Data Enrichment from a Lookup Data Set • 392, 395, 396
Configuring Data Enrichment from Event SQL • 396, 399

Configuring Data Enrichment Policies • 392, 394
Configuring Data Enrichment Policy Rules • 395, 396, 399
Configuring Data Enrichment Web Policy Rules • 395
Configuring Data Scope • 242, 244
Configuring Default Domain Credentials • 378, 379
Configuring Direct User Tracking • 323
Configuring FireEye MPS to Pass Information about Malware-infected Hosts to SecureSphere • 469
Configuring FireEye to Enable SecureSphere Malware Detection Support • 464, 468
Configuring Forwarded Connections • 67, 455, 457
Configuring Gateway Graphs • 90, 91
Configuring Gateway HTTP Proxies • 364, 449
Configuring General Report Details • 230, 242, 243, 252
Configuring Host/URL to Application Mapping • 61, 62, 64, 70, 448
Configuring Integration with Active Directory for User Management • 379, 414, 425
Configuring Integration with ArcSight • 414
Configuring Integration with BMC Remedy Service Desk • 414, 415
Configuring Integration with Third Party Management Systems • 116, 414
Configuring IP Based Restrictions • 441, 442
Configuring IP Groups • 34, 199, 213, 363, 365, 366, 442, 458, 631
Configuring Lookup Data Set Keywords and Relevance • 389, 390, 398
Configuring Lookup Data Set Permissions • 389, 391
Configuring Malware Detection Support • 463
Configuring NTLM Support • 69, 304
Configuring OCSP Support • 73, 225, 438, 447
Configuring Operation Mode • 52, 53, 57
Configuring Parameter Groups • 67, 364, 439
Configuring Parameter Type Configuration Objects • 364, 439, 440
Configuring Permissions • 242, 249
Configuring Permissions for the Malware Detection Feed User • 465, 466
Configuring Plugins • 502, 503
Configuring Policy or Report with Followed Action for Remedy • 416, 418
Configuring Preferences • 337, 412
Configuring Reports • 230, 241, 242
Configuring Reports for Distribution by e-mail • 241, 250
Configuring Reports With Followed Actions • 244, 250, 360
Configuring Reports with Tasks • 250, 251
Configuring Restrictions • 438, 441
Configuring Scheduling • 33, 345, 380, 389, 436
Configuring SecureSphere for Integration with BMC Remedy Service Desk • 416, 419
Configuring SecureSphere to Send Audit Records to enVision • 423
Configuring SecureSphere to Send Security Alerts to enVision • 421
Configuring SecureSphere to Send System Events to enVision • 421, 423
Configuring SecureSphere to Work with QualysGuard • 148, 153
Configuring SecureSphere to Work with WhiteHat Sentinel • 148, 151
Configuring Sensitive Data Protection • 67, 224, 364, 369, 371, 498, 630
Configuring Service Credential Options • 33, 35
Configuring Session Tracking • 77, 304, 313, 314, 337, 450
Configuring SOAP Elements • 364, 439, 440
Configuring Source Restrictions • 442, 443
Configuring SQL Connection User Tracking • 317, 318
Configuring Static File Lists • 364, 439, 440

Configuring Susceptible Directories • 222, 260, 271, 364, 449
Configuring System Events to Interface with 3rd Party Tools • 116
Configuring Tabular Options • 242, 245
Configuring the Default Domain Name • 378
Configuring the Default Error Page • 61, 62, 66, 445
Configuring the Malware Detection Feed User • 464
Configuring the Malware Detection IP Group • 464, 467
Configuring URL Prefixes / Directory Based Restrictions • 64, 73, 364, 441, 443, 448
Configuring URL Prefixes/Directories Group • 443, 444
Configuring URL Rewrite and URL Redirection • 453
Configuring User Details • 412
Configuring User Preferences • 412, 413
Configuring Vulnerability Due Date • 164, 171, 175
Configuring Web Application User Tracking • 303
Configuring Web Error Page Policies • 66, 438, 444, 445, 446
Configuring Web Error Pages • 445, 446
Configuring Web Service Restrictions • 57, 67, 337, 438, 439, 443
Configuring Webworm Protection • 443, 449, 458
Configuring Windows Domains • 363, 377
Content Hashing in Data Enrichment Policies • 402
Cookie Encryption • 208, 209
Cookie Injection • 269, 292
Cookie Prefixes • 269, 270
Cookie Protection in Web Profiles • 208, 238, 254, 255, 260, 266, 271
Cookie Signing • 208, 209
Cookie Signing Validation Policy • 182, 208
Cookie Status • 268, 292
Cookie Tampering • 269
Copyright Notice • 2
Correlated Attack Validation (CAV) Used in Correlation Policies • 183, 211, 216, 223
Correlated Queries Tab • 258, 261, 264, 274, 313
Creating a Lookup Data Set • 383, 385, 392, 394, 395, 396
Creating a Lookup Data Set from a CSV File • 386
Creating a Lookup Data Set from a Template • 386, 387
Creating a Lookup Data Set from an External System • 386, 387
Creating a Multiparty Review Workflow • 103, 104, 118
Creating a New Change Tracking - External Objects Policy • 405
Creating a New Data Type • 377
Creating a New Dictionary • 329
Creating a New Server Group • 52, 53, 62
Creating a New Site • 51, 53, 193
Creating a Report • 231, 233, 234, 235, 236, 237, 238, 239, 240, 241, 252
Creating a Request Query • 152, 153
Creating a Scan • 32, 33
Creating a System Events Policy • 434, 571
Creating a URL Pattern • 272, 278
Creating a URL Pattern from an Existing URL • 272, 279
Creating a URL Rewrite Group • 454, 455
Creating a URL Rewrite Rule • 451, 454
Creating a Web API Key on WhiteHat Sentinel • 151, 152
Creating a Web Application User Tracking Decision Rule • 303, 307, 311
Creating a Web Scanner Integration Policy Using Vulnerabilities from WhiteHat Sentinel • 150, 152, 153
Creating an Enrichment Policy • 392, 394, 402
Creating an External Role • 425, 430, 432

Creating an IP Group Individually • 367, 368
Creating and Configuring a Web Scanner Integration Policy • 148, 149, 150, 153, 360
Creating and Managing Reports • 240
Creating Assignment Tasks • 407, 409, 410
Creating Custom Action Sets • 251, 346, 347, 348, 349, 351, 410, 414, 416
Creating New Dictionary Groups • 369, 370, 371, 374, 381
Creating New Global Objects • 365, 366, 375, 376, 439, 440, 442, 443, 445, 446, 449, 450
Creating New Web Services • 61, 62
Creating Review Tasks • 407, 409, 410
Creating the Malware Detection Feed User • 465
Custom Database Plugins • 516
Custom Plugin • 510
Custom Plugins • 266, 503
Custom Plugins Reference • 502, 511
Custom Policies Match Criteria • 186, 383, 624
Custom Security Events • 561
Custom Web Plugins • 257, 511
Customizing How SecureSphere Handles Web Traffic • 438
Customizing Service Discovery Advanced Options • 33, 34
CVSS Risk Scoring • 154

D

Dashboard Management Workflow • 89
Dataset
    Lookup Attribute Usage Example • 599, 626, 633
DB Profiling and UUT • 300, 301
Defining Archive Settings • 343, 344
Defining Cloud Communication Parameters • 135, 136
Defining Custom Malware Detection Policies • 477, 481
Defining External Systems • 426
Defining Web to Database User Tracking Settings • 313, 314
Deleting a URL • 272, 278
Deleting a Web Application User Tracking Decision Rule • 303, 310
Deleting Alerts from the Alerts Table • 101
Deleting All URLs and Folders in a Profile • 274, 280
Deleting an SQL User Tracking Group - Database Service Association • 322
Deleting Filtered Alerts from the Alerts Table • 101, 117
Detecting Optimization Issues • 282
Direct User Tracking • 301, 322
Direct User Tracking Architecture • 323
Directly Assigning Followed Actions from the Security Policy Window • 185, 360
Directly Configuring Policies from the Security Policies Window • 184, 189, 360
Disabling and Enabling ThreatRadar Reputation Services • 125
Disabling Automatic Web Profiling • 190, 193, 194, 290, 293, 323
Discovering and Classifying Network Assets • 30
Discovery • 27
Discovery Filter Criteria • 31, 614
Discovery Report Match Criteria • 612
Displaying Information About a Profile's URLs • 254, 270
Displaying Local IP Addresses Identified as Infected by Malware • 471, 472
Downloading Recommended Malware Detection Policies • 478
Drilling Down and Navigating in the Risk Explorer • 158, 161
Dynamic Profiling • 27

E

Editing a Change Tracking - External Objects Policy • 407
Editing a Signature • 325, 328
Editing a URL Parameter Value Type • 271, 274, 292, 440, 498

Editing a URL's Methods • 262, 271, 274
Editing a Web Profile APU Policy Rule • 298
Editing Discovered Servers • 41, 42
Editing Policies from the Details Pane • 101
Enabling and Disabling a Signature • 325, 328
Enabling and Disabling Cookie Signing and Encryption • 208, 209
Enabling Application Group Settings • 342
Enabling Nested XML Parsing • 264
Enabling Response Page in Violations • 112, 185, 188, 192, 193, 194
End User License and Services Agreement • 3
Enriching Data • 109, 383, 392
Enrichment Policy Match Criteria • 395, 631
Escaping • 500
Event Types • 571
Examining Risk with Risk Views • 155, 158
Examples of Syslog Events for ArcSight • 355, 357, 559, 560
Exporting Alerts • 102
Exporting and Importing Individual Objects • 45
Extended Fields • 560, 561, 565
External Roles and Users Daily Updates • 433
External System - LDAP Authentication and Authorization • 425, 427
External Systems - LDAP • 427, 432

F

Filter Criteria and Report Data Fields • 191, 598
Filter Criteria and Report Data Fields Overview • 120, 245, 599
Filtering Alerts and Violations • 91, 92, 106, 117
Filtering Alerts in the Dashboard • 90
Filtering Data in the Risk Explorer • 159
Filtering Policies • 185, 190, 191, 194, 361
Filtering Reports • 230, 252
Firewall Policy • 182, 197
Firewall Security Events • 561
Flagging Alerts • 95, 102
Form-Based User Authentication • 302

G

General Basic Configuration Tasks • 74
Generating a PDF Listing Vulnerabilities • 176
Generic Dictionary Groups • 363, 371, 380, 627
Geographic Alert Map • 93
Glossary • 706
Grace Period • 268, 269

H

Hashed Users in Database Profiling and UUT • 301, 302
How Security Policies Work with Community Defense • 135
hpl_db_source • 516
hpl_dq2q • 517
hpl_p2p • 511, 515
hpl_p2u • 511, 514
hpl_u2p and hpl_u2pp - URL to Parameters • 498, 511
hpl_u2pp • 512
hpl_web_separator • 511, 516
HTTP Methods and Parameters • 261, 273, 292
HTTP Protocol Signature Policies • 182, 184, 210
HTTP Protocol Validation Policy • 182, 202

I

Identifying Advanced Configuration Requirements • 337
Identifying False Positives • 95, 99, 100
Ignoring Optimization Issues • 282, 287
Importing an IP Group from a CSV File • 367
Importing Credentials • 36, 37
Importing Data from a CSV File • 372, 374
Importing Protected IP Addresses from a CSV File • 53, 54
Importing Recommended Malware Detection Policies • 478, 479
Integrating Web Scanners Task Overview • 148

Integrating with RSA enVision • 414, 420
Introducing Active Modules • 403
Introducing Community Defense • 121, 133
Introducing SecureSphere Web Application Security • 22
Introducing Security Policies • 82, 179, 333
Introducing ThreatRadar Reputation Services • 121, 122
Introduction to Common Vulnerabilities and Exposures • 164, 603, 612
Introduction to SecureSphere Discovery and Classification • 30
Introduction to SecureSphere Risk Management • 143
Intrusion Prevention Used in Signature Policies • 183, 210, 223
IP Forensics Service • 121, 131
IPv4 / IPv6 Dual Stack • 48
IPV6 Address Formats • 47
IPv6 Support • 46
IPv6 Support in Active Modules • 49
IPv6 Support in External Systems • 46, 49
IPv6 Support in Hostnames • 48
IPv6 Support in IP Groups • 49
IPv6 Support in Lookup Data Sets • 49
IPv6 Support in SecureSphere MX and Gateway • 48
IPv6 Support in Server Groups • 49
IPv6 Support in ThreatRadar Reputation Services • 49
IPv6 Support in WAF • 48

K

Kernel (Non-Transparent) Reverse Proxy • 462, 482

L

Learn Individual Queries • 301
Learned URLs • 254, 260, 271
Learning Action URLs • 303
Learning Mode • 255, 256, 260
Legacy Policies • 195, 202, 210
Licensing Information • 126
List of Collection Placeholders • 520, 545
List of Rule.Dn Associated IDs • 549
List of SecureSphere Events • 571
List of Standard Placeholders • 519, 522
Locking a URL Directory • 272, 276
Locking a Web Profile • 272, 277
Locking Application Groups • 342
Logging Into SecureSphere in the Authorization Mode • 425, 432

M

Macros for use with ArcSight Placeholders • 559
Major Discovery and Classification Tasks • 31
Malicious IPs Service • 121, 130
Malicious IPs Summary • 130
Malware Detection Support • 462
Managing Alerts • 91
Managing and Mitigating Vulnerabilities • 172
Managing Archive and Key Settings • 343, 344, 345
Managing Blocked Sources • 111, 188
Managing Columns in Lookup Data Sets • 389
Managing Discovered Servers • 38
Managing Favorite Reports • 230, 243, 252
Managing Keys • 343
Managing System Events • 113, 158, 434
Managing the Anonymous Proxy Service • 129
Managing the Malicious IPs Service • 131
Managing the Phishing URLs Service • 127
Managing the Remote File Inclusion Service • 142
Managing the SQL Injection Service • 142
Managing the TOR IPs Service • 132
Managing Violation Data Sharing • 135, 137
Managing Violations • 105, 126, 127, 129, 130, 131, 132, 140
Managing Vulnerabilities • 145, 147, 151, 162
Managing Web Applications • 70, 257
Manually Accepting and Rejecting Discovered Servers • 41, 43

Manually Adding a Learned Host • 272, 280
Manually Adding a URL • 256, 272, 277
Manually Creating a Lookup Data Set • 385, 386
Mapping Host or URL to a Web Application • 64
Marking a Vulnerability as Mitigated • 174, 195
Masking Data in Elements • 369, 375
Modifying Advanced Settings for Malware Detection Support • 463, 466, 471, 473
Modifying Policies in the Security Window • 186, 189, 210, 211, 212, 223, 290
Modifying Security Policies • 190, 191, 193, 194, 347, 360
Modifying Security Policies in the Monitoring Window • 186, 189, 194
Modifying Service Name and Port • 61, 66, 74
Modifying the Character Set • 61, 66, 75
Modifying the Malware Name White List • 474
Modifying the Malware Target White List • 474
Modifying Threat Levels of Malware Types • 472, 473, 475
Monitoring • 28, 79
Monitoring Architecture • 83, 84, 85
Monitoring Checklist • 85, 95
Monitoring Gateway Status • 90
Monitoring Models in SecureSphere • 80
Multi Layer Protection • 24
Multi Part Signatures • 496, 497
MXs Managed by SOM • 488

N

Nested XML Parsing • 68, 263
Network Protocol Validation • 182, 199
Network Security Policies • 197

O

OCSP Protocol Validation Policies • 183, 219, 225, 448
Opening Violations from the Alerts Table • 101
Order of Plugin Execution • 503, 512
Overall Graphs • 260, 280, 281
Overview • 22, 311
Overview of Change Tracking - External Objects • 404
Overview of Load Balancer and Forwarded Connections • 455
Overview of SecureSphere Web Profiles • 254

P

Parameter Modification Plugin • 505, 509
Parameter Modification Plugin Example • 509
Parameter to URL Plugin • 505, 507
Parameter to URL Plugin Example - I • 507
Parameter to URL Plugin Example - II • 508
Parameter to URL Plugin Example - III • 508
Parentheses • 499
Persistent Cookies • 267
Phishing URLs Service • 121, 126
Phishing URLs Summary • 127
Placeholders • 518
Plugins • 66, 502
Plugins Overview • 407, 502
Policies • 28
Predefined Lookup Data Sets • 127, 129, 131, 132, 383, 385
Pre-Defined Reports • 404, 661
Preparing for Change • 294, 296
Profile Criteria • 238, 619
Profile Modes • 255, 275
Protect Mode • 255, 256, 260
Protecting and Auditing Custom Information with Lookup Data • 363, 382, 385
Providing Accurate IP Blocking • 455, 457
Providing Full Visibility of Client IP Addresses • 455, 456
Providing Multi Level Protection Using Security Policies • 180

R

Regenerating the List of Optimization Issues • 282, 288
Registering MXs • 489
Regular Expression Syntax in Brief • 498

Regular Expressions • 440, 485, 498, 504
Regular Expressions in SecureSphere • 275, 498
Regular Security Events • 560
Reinstating Rejected Discovered Servers • 42, 43
Relative Modifiers • 496, 498
Releasing Blocked Sources • 112
Remedy Server Configuration Prerequisites • 416
Remote File Inclusion • 121, 141
Remote File Inclusion Summary • 141
Removing MXs from SOM • 495
Renaming Sites • 50, 51
Report Columns • 246, 636
Report Management Main Tasks • 240
Report Terminology • 228
Reporting • 28, 87, 227, 661
Reports Introduction • 227
Reverse Proxies • 443, 462, 482, 498
Reviewing Optimization Issues • 282, 283, 287
Risk Details and Vulnerability Management Workflow • 158, 161
Risk Management • 28, 143
Risk Management Filter Criteria • 610
Risk Management Process Overview • 144
Rule.dn Associated IDs • 549
Run on Every Event • 355, 357, 358, 359
Running a Report • 241, 242, 250, 251
Running a Scan • 32, 33, 37
Running a Web Scanner Assessment Job • 148, 150

S

Sample SecureSphere User Types • 80
Scanning and Assessing Web Services • 145, 146, 147
Scheduling Reports • 241, 242, 249, 251
SecureSphere Architecture • 26
SecureSphere Main Features • 26
Securing Web Applications • 192, 219, 449
Securing Web Transport • 192, 202
Security Policies • 179, 383, 477
Security Policy Exceptions • 100, 104, 185, 186, 190, 192, 193, 194, 290, 405
Security Policy Main Tasks • 189
Security Policy Structure • 184
Security Policy Types • 180, 181, 190, 193, 195
Sending a Vulnerability • 164, 177
Sensitive Data Dictionary Groups Use Cases • 369
Server Group Applied Policies Information • 60
Server Group Best Practices • 52
Server Group Definition Parameters • 53, 56, 59
Server Group Server Information • 58
Server Group Services and Ports • 58
Service Discovery Detailed Report • 613, 651
Service Discovery Detailed Report Columns • 652
Service Discovery Execution Report • 612, 614, 651
Service Discovery Execution Summary Report Columns • 654
Service Discovery Reports • 237
Session Cookies • 266
Setting Up Application Groups • 341
Setting Up SSL with Certificate Authentication • 489, 495
Setting Up SSL with Certificate Authentication on MX • 490, 492
Setting Up SSL with Certificate Authentication on SOM • 490
Setting Up the Other Authentication Options • 494
Setting Vulnerability Priority • 164, 171, 173
Setting Vulnerability Tracking Status • 171, 175
Signature Tasks • 326
Signatures • 325
Signatures Not Applied by Default • 326

Signatures Workflow • 325, 326
Single Part Signatures • 496, 497
Site Best Practices • 50, 52
SOAP Actions • 261, 262, 274
Special Parameter Delimiter Plugin • 505, 510
Special Parameter Delimiter Plugin Example: • 510
Specifying Names and Targets to Exclude from Malware Identification • 472, 473
Specifying Session Identifiers and Authentication Methods • 303, 304
SQL Connection User Tracking • 301, 316
SQL Connection User Tracking Architecture • 316, 317
SQL Connection User Tracking Overview • 317
SQL Injection • 121, 142
SQL Injection Summary • 142
SSL Certificate Expiration • 493
SSL Ciphers • 75, 76
Standard Dictionaries • 333
Standard Fields • 560, 561, 562
Static URLs • 265
Stream Signature • 182, 201
Summary of Global Objects • 361, 362
Supporting Web Load Balancers • 455
Susceptible Directory Group • 222
Switching Between the URL Modes • 260, 271, 275
Switching the Mode of Multiple URLs • 272, 275
System Event Criteria • 233, 603, 610
System Event Policies • 360, 434
System Events • 561
System Events Reports • 232, 237, 670

T

Taking Action on Alerts and Violations • 95, 104, 186
Task Followed Action Configuration Guidelines • 407, 409, 410
Task Report Criteria • 605
Task Types • 407, 408
Tasks Overview • 407, 408
The SecureSphere WAF System • 25
ThreatRadar • 23
ThreatRadar Fraud Prevention Services • 89, 110, 122, 182, 219, 386, 627
ThreatRadar Services • 121
Toggling between Data and Network View in the Risk Explorer • 158
Topology Examples • 315
TOR IPs Service • 121, 131
TOR IPs Summary • 132
Transparent Inspection • 24
Transparent Reverse Proxy • 485
Troubleshooting Hashed Users • 704
Tuning a Web Profile APU Policy • 294, 296
Tuning a Web Profile Policy • 289
Tuning Security Policies in the Setup Window • 60, 70, 74, 186, 189, 193
Tuning Web Profiles • 257, 271

U

Understanding Aggregated Alerts • 98, 99, 107, 671
Understanding Alerts, Events and Violations • 82, 631
Understanding Community Defense Summary • 135, 140
Understanding Data Analysis View Data Sources • 248
Understanding Malware Types • 472, 475, 477
Understanding Recommended Malware Detection Policies • 477, 480, 481
Understanding Report Types • 230, 241
Understanding Scan History • 37
Understanding Server Group Parameters • 53, 56
Understanding Single Collection Placeholders • 519, 520
Understanding System Events • 116
Understanding the Alerts Table • 92, 95

Understanding the Discovery and Classification Window • 31
Understanding the Execution Plan • 438
Understanding the List of Local IP Addresses Identified as Infected by Malware • 463, 466, 471, 473
Understanding the Malware Infected Local IP Addresses Table • 471, 472
Understanding the Nested Collection Placeholder • 519, 520, 546
Understanding the Reports Window • 228
Understanding the Risk Details Window • 160
Understanding the Risk Explorer • 156, 157
Understanding the Risk Management Window • 145
Understanding the System Events Table • 116
Understanding the Tasks List • 407, 408, 409
Understanding the Violations Table • 106, 107
Understanding Threat Levels • 472, 475, 476, 477
Understanding Universal User Tracking • 300, 626
Understanding UUT Components • 300
Understanding Violations • 107
Understanding Web Application Parameters • 71, 72
Understanding Web APU Profiling Dynamics • 294, 298
Understanding Web Profiles in SecureSphere • 254
Understanding Web Services Parameters • 62, 65
Universal User Tracking • 299
Universal User Tracking Architecture • 300, 301
Unmitigating a Blocked Vulnerability • 175, 177
URL Patterns • 254, 256, 260, 266, 271, 286
URL Redirection • 452
URL Rewrite • 69, 365, 451, 483, 486
URL Rewrite Architecture • 451
URL Rewrite Groups • 453
URL to Parameter Plugin • 286, 505
URL to Parameter Plugin Example - I • 506
URL to Parameter Plugin Example - II • 506
Use Case
    ISP Implementation • 81
    Security Officer Management • 81
    Small to Medium Business Implementation • 81
Using a Query to Populate a Lookup Data Set • 387, 388, 389, 390
Using Default Security Policies • 60, 70, 73, 185, 189, 195
Using Regular Expressions in Signatures • 498, 500
Using Sensitive Data Dictionaries Configuration Guidelines • 369, 370
Using Sensitive Data Dictionary Groups • 337, 363, 369
Using Service Discovery to Populate a SecureSphere Site • 32, 43
Using the Advanced Filter • 117, 120, 610, 614
Using the Basic Filter • 117, 118, 252
Using the Quick Filter • 117

V

Validating SSL Client Certificates • 438, 459, 462, 483
Verifying that FireEye Passes Malware Information to SecureSphere • 469, 471
Viewing a Change Tracking - External Objects Policy • 407
Viewing a Signature's Details • 325, 327
Viewing a Specific Gateway Details • 91
Viewing a URL Pattern • 266, 272, 278
Viewing a Web Profile Automatic Update Policy • 297
Viewing a Web Profile Policy • 262, 290, 293, 296
Viewing Alerts in the Alerts Window • 92
Viewing and Editing Cookies • 266, 269
Viewing and Editing URLs • 271, 272

Viewing and Filtering Signatures • 325, 327
Viewing Blocked Sources • 111
Viewing Discovered Server Details in Tabular View • 39
Viewing Discovered Servers in Graphical Views • 39
Viewing Discovered Vulnerabilities in Graphical Views • 163, 166
Viewing Discovered Vulnerabilities in Tabular View • 163, 164, 165
Viewing External Users • 425, 432
Viewing Generated Reports • 228, 241, 251
Viewing Profiles • 258, 259
Viewing Results • 242, 248
Viewing System Events • 114
Viewing the Activity Log • 169, 172, 177
Viewing URLs • 260
Viewing Violations in the Violations Window • 106
Viewing Web Error Pages and Responses • 112
Viewing, Manually Adding and Editing Action URLs • 303, 305, 308, 311
Violation Analysis Tools • 86, 108, 109
Vulnerability • 28
Vulnerability Aggregation • 166
Vulnerability Criteria • 603, 610
Vulnerability Management Filter Criteria • 610
Vulnerability Management Workflow • 163

W

Web Application Applied Policies Information • 73
Web Application Custom Policies • 182, 183, 192, 219, 223, 370, 371, 372, 374
Web Application Definitions Parameters • 67, 72, 225, 441, 448
Web Application Firewall • 23
Web Application Profiles • 73, 253, 259, 261, 281, 282
Web Application Signatures Policy • 182, 184, 219, 223
Web Application User Tracking • 254, 259, 271, 301, 302, 313, 393, 626, 633
Web Application User Tracking Pane Parameters • 306
Web APU (Automatic Profile Update) • 276, 277, 294
Web Custom Policies Match Criteria • 211, 625
Web Error Pages and Response Pages in Alerts • 109, 112
Web Profile - Learned Hosts Report • 619, 622, 655
Web Profile - Learned Hosts Report Columns • 658
Web Profile - Susceptible Directories Report • 619, 623, 655
Web Profile - Susceptible Directories Report Columns • 658
Web Profile - Web Application User Tracking Report • 619, 623, 655
Web Profile - Web Application User Tracking Report Columns • 659
Web Profile and Web Profile Policy • 182, 219, 222
Web Profile Components • 254
Web Profile Cookies Report • 619, 621, 655
Web Profile Cookies Report Columns • 657
Web Profile Graphs • 280
Web Profile Menu • 259
Web Profile Optimization • 258, 260, 281
Web Profile Optimization Issues • 281, 282, 283
Web Profile Optimization Overview • 281, 285, 288
Web Profile Policy • 222, 257, 289, 630
Web Profile Reports • 238
Web Profile Structure • 255
Web Profile System Definitions • 298
Web Profile Tasks • 256
Web Profile URL Report • 619, 655
Web Profile URL Report Columns • 655
Web Service Applications • 69

Web Service Applied Policies Information • 70
Web Service Correlated Validation Policy • 182, 183, 216
Web Service Custom Policies • 182, 183, 192, 211, 458
Web Service Definition Parameters • 65
Web Service Enrichment Policy Match Criteria • 631, 632
Web Service Operations • 66, 263, 448
Web Service Reverse Proxy • 69
Web Service Vulnerabilities Overview • 147
Web to Database User Tracking • 301, 311
Web to Database User Tracking Configuration • 312
Web Worm Policy • 182, 219, 222
Working with Action Sets and Followed Actions • 77, 150, 177, 191, 198, 345, 434, 435, 457
Working with Alert Reports • 231, 232
Working with Application Groups • 339, 341
Working with Collection Placeholders • 519, 545, 547
Working with Configuration Reports • 231, 234
Working with Data Types • 376
Working with Default Action Sets • 348
Working with Dictionaries • 326, 329
Working with Discovered Servers • 32, 39, 42
Working with Discovery Reports • 231, 236
Working with Global Objects • 361
Working with Global Ports Groups • 35, 58, 363, 366
Working with Open Tasks • 407, 411
Working with Placeholders • 354, 355, 356, 518
Working with Pre-Defined Vulnerability Reports • 231, 236
Working with Profile Reports • 231, 237
Working with Reports • 230
Working with SecureSphere Scans • 31, 32, 377
Working with Signatures • 183, 254, 325, 326
Working with SOM • 488
Working with Standard Placeholders • 518, 519
Working with Stored Procedures • 400, 401
Working with System Event Policies • 434
Working with System Event Reports • 231, 233
Working with Task Reports • 231, 234
Working with Tasks • 337, 353, 360, 407
Working with the CVE Knowledge Base • 164, 165, 171
Working with the Dashboard • 87
Working with ThreatRadar Reputation Services • 122
Working with Vulnerability Reports • 231, 235
Writing Signatures • 496
Writing Signatures and Regular Expressions • 333, 496, 628

X

XFF Forwarded Addresses As Source IP Address Exceptions • 187
XML Parsing • 262