OPTIMIZE

MERCURY BUSINESS AVAILABILITY CENTER™

Hardening the Platform

MERCURY™

BUSINESS TECHNOLOGY OPTIMIZATION

Mercury Business Availability Center

Hardening the Platform

Version 6.1

Document Release Date: March 5, 2006

MERCURY™

Mercury Business Availability Center, Version 6.1
Hardening the Platform

This manual, and the accompanying software and other documentation, is protected by U.S. and international copyright laws, and may be used only in accordance with the accompanying license agreement. Features of the software, and of other products and services of Mercury Interactive Corporation, may be covered by one or more of the following patents: United States: 5,511,185; 5,657,438; 5,701,139; 5,870,559; 5,958,008; 5,974,572; 6,137,782; 6,138,157; 6,144,962; 6,205,122; 6,237,006; 6,341,310; 6,360,332, 6,449,739; 6,470,383; 6,477,483; 6,549,944; 6,560,564; 6,564,342; 6,587,969; 6,631,408; 6,631,411; 6,633,912; 6,694,288; 6,738,813; 6,738,933; 6,754,701; 6,792,460 and 6,810,494. Australia: 763468 and 762554. Other patents pending. All rights reserved.

Mercury, Mercury Interactive, the Mercury logo, the Mercury Interactive logo, LoadRunner, WinRunner, SiteScope and TestDirector are trademarks of Mercury Interactive Corporation and may be registered in certain jurisdictions. The absence of a trademark from this list does not constitute a waiver of Mercury's intellectual property rights concerning that trademark.

All other company, brand and product names may be trademarks or registered trademarks of their respective holders. Mercury disclaims any responsibility for specifying which marks are owned by which companies or which organizations.

Mercury provides links to external third-party Web sites to help you find supplemental information. Site content and availability may change without notice. Mercury makes no representations or warranties whatsoever as to site content or availability.

Mercury Interactive Corporation
379 North Whisman Road
Mountain View, CA 94043
Tel: (650) 603-5200
Toll Free: (800) TEST-911
Customer Support: (877) TEST-HLP
Fax: (650) 603-5300

© 2005-2006 Mercury Interactive Corporation, All rights reserved

If you have any comments or suggestions regarding this document, please send them by e-mail to documentation@mercury.com.

AMLIB_SECUR6.1/01

Table of Contents

Welcome to Hardening the Platform v
How This Guide Is Organized v
Who Should Read This Guide vi
Getting More Information vi

Chapter 1: Introduction to Hardening the Mercury Business Availability Center Platform 1
Introduction to Hardening for Mercury Business Availability Center 2
Deploying Mercury Business Availability Center in a Secure Architecture 4
Using the Hardening Guidelines 5

Chapter 2: Using a Reverse Proxy in Mercury Business Availability Center 7
Overview of Reverse Proxies 8
Security Aspects of Using Reverse Proxies 8
Mercury Business Availability Center and Reverse Proxies 9
Specific and Generic Reverse Proxy Mode Support for Mercury Business Availability Center 11
Using a Reverse Proxy with a Single Machine Installation 13
Using a Reverse Proxy with a Distributed Server Installation 20

Chapter 3: Using SSL in Mercury Business Availability Center.....29
Introducing SSL Deployment in Mercury Business Availability Center.....30
Mercury Business Availability Center Components Supporting SSL.....33
SSL-Supported Topologies in Mercury Business Availability Center.....34
Configuring SSL from the Application Users to the Centers Server.....35
Configuring SSL from the Data Collectors to the Core Server.....37
Configuring SSL From the Centers Server to Data Collectors.....47
Configuring the Web Guard to Support SSL.....57
Setting Java Runtime Environment to Work With Client/Server Certificates.....58

Chapter 4: Using Basic Authentication in Mercury Business Availability Center.....63
Introducing Basic Authentication Deployment in Mercury Business Availability Center.....64
Mercury Business Availability Center Components Supporting Basic Authentication.....66
Configuring Basic Authentication Between the Centers Server and Application Users.....68
Configuring Basic Authentication Between the Core Server and the Data Collectors.....71
Auto Upgrading Data Collectors Remotely when Using Basic Authentication.....77

Index.....79

Welcome to Hardening the Platform

This guide provides you with detailed instructions on hardening the Mercury Business Availability Center platform.

Note: This guide is not relevant to Mercury Managed Services customers.

How This Guide Is Organized

The guide contains the following chapters:

Chapter 1 Introduction to Hardening the Mercury Business Availability Center Platform

Describes the concept of a secure Mercury Business Availability Center platform and discusses the planning and architecture required to implement a secure platform.

Chapter 2 Using a Reverse Proxy in Mercury Business Availability Center

Describes how to use a reverse proxy with Mercury Business Availability Center in order to help secure Mercury Business Availability Center architecture.

Chapter 3 Using SSL in Mercury Business Availability Center

Describes how to configure the Mercury Business Availability Center platform to support Secure Sockets Layer (SSL) communication.

Chapter 4 Using Basic Authentication in Mercury Business Availability Center

Describes how to configure the Mercury Business Availability Center platform to support communication using basic authentication.

Who Should Read This Guide

This guide is intended for the following users of Mercury Business Availability Center:

- Mercury Business Availability Center administrators
- Security administrators

Readers of this guide should be highly knowledgeable about enterprise system security.

Getting More Information

For information on using and updating the Mercury Business Availability Center Documentation Library, reference information on additional documentation resources, typographical conventions used in the Documentation Library, and quick reference information on deploying, administering, and using Mercury Business Availability Center, refer to *Getting Started with Mercury Business Availability Center*.

1

Introduction to Hardening the Mercury Business Availability Center Platform

This chapter introduces the concept of a secure Mercury Business Availability Center platform and discusses the planning and architecture required to implement a secure platform. It is strongly recommended that you read this chapter before proceeding to the following chapters, which describe the actual hardening procedures.

This chapter describes:	On page:
Introduction to Hardening for Mercury Business Availability Center	2
Deploying Mercury Business Availability Center in a Secure Architecture	4
Using the Hardening Guidelines	5

Introduction to Hardening for Mercury Business Availability Center

The Mercury Business Availability Center platform is designed so that it can be part of a secure architecture, and can therefore meet the challenge of dealing with the security threats to which it could potentially be exposed.

The hardening guidelines deal with the configuration required to implement a more secure (hardened) Mercury Business Availability Center platform. The hardening guidelines relate to both single machine (where all servers are installed on the same machine) and distributed (where all Servers are installed on separate machines) deployments of Mercury Business Availability Center.

The hardening information provided is intended primarily for Mercury Business Availability Center administrators, and for the technical operator of each component that is involved in the implementation of a secure Mercury Business Availability Center platform (for example, the Web server). These people should familiarize themselves with the hardening settings and recommendations prior to beginning the hardening procedures.

Note: From version 6.1 onwards, Mercury Business Availability Center uses an improved, proactive security mechanism that validates HTTP inputs for security violations.

Before You Start

In order to best use the hardening guidelines given here for your particular organization, you should do the following before starting the hardening procedures:

- Evaluate the security risk/security state for your general network, and use the conclusions when deciding how to best integrate the Mercury Business Availability Center platform into your network.
- Review all the hardening guidelines.

A good understanding of the Mercury Business Availability Center technical framework and Mercury Business Availability Center security capabilities will facilitate designing a solid plan for implementing a secure Mercury Business Availability Center platform.

Note: The hardening information provided in this section is not intended as a guide to making a security risk assessment for your computerized systems.

You should also note the following points when using the hardening guidelines:

- Verify that the Mercury Business Availability Center platform is fully functioning before starting the hardening procedures.
- Follow the hardening procedure steps chronologically in each chapter. For example, if you decide to configure the Mercury Business Availability Center servers to support SSL, read Chapter 3, "Using SSL in Mercury Business Availability Center" and then follow all the instructions chronologically.
- The Mercury Business Availability Center components do not support basic authentication with blank passwords. Do not use a blank password when setting basic authentication connection parameters.
- The hardening procedures are based on the assumption that you are implementing only the instructions provided in these chapters, and not performing other hardening steps not documented here.
- Where the hardening procedures focus on a particular distributed architecture, this does not imply that this is the best architecture to fit your organization's needs.
- It is assumed that the procedures included in the following chapters will be performed on machines dedicated to the Mercury Business Availability Center platform. Using the machines for other purposes in addition to Mercury Business Availability Center may yield problematic results.

Tip: You can print out the hardening procedures and check them off as you implement them.

Deploying Mercury Business Availability Center in a Secure Architecture

The secure architecture referred to in this document is a typical DMZ architecture using a device as a firewall. The basic concept of such an architecture is to create a complete separation, and avoid direct access between the Mercury Business Availability Center clients and the Mercury Business Availability Center servers.

One of the more secure and recommended solutions to deploy Mercury Business Availability Center in such an environment is by using a reverse proxy.

Mercury Business Availability Center 6.0 fully supports the reverse proxy secure architecture solution.

The following security objectives can be achieved by using a reverse proxy in DMZ proxy HTTP/HTTPS communication with Mercury Business Availability Center:

- No Mercury Business Availability Center logic or data resides on the DMZ.
- No direct communication between Mercury Business Availability Center clients and servers is permitted.
- No direct connection from the DMZ to the Mercury Business Availability Center database is required.
- The protocol used to communicate with the reverse proxy can be HTTP/S. HTTP can be statefully inspected by firewalls if required.
- A static, restricted set of redirect requests can be defined on the reverse proxy.

- Most of the Web server security features are available on the reverse proxy (authentication methods, encryption, and others).
- The reverse proxy screens the IP addresses of the real Mercury Business Availability Center servers as well as the architecture of the internal network.
- The only accessible client of the Web server is the reverse proxy.
- This configuration supports NAT firewalls.
- The reverse proxy requires a minimal number of open ports in the firewall.
- The reverse proxy provides good performance compared to other bastion host solutions.

It is strongly recommended that you use a reverse proxy with Mercury Business Availability Center to achieve a secure architecture. For details on configuring a reverse proxy for use with Mercury Business Availability Center, see Chapter 2, "Using a Reverse Proxy in Mercury Business Availability Center."

If you must use another type of secure architecture with your Mercury Business Availability Center platform, contact Mercury Customer Support to determine which architecture is the best one for you to use.

Using the Hardening Guidelines

The chapters in this guide discuss the following hardening topics:

- **Using a reverse proxy in Mercury Business Availability Center** – This chapter contains information on using a reverse proxy with Mercury Business Availability Center in order to help secure Mercury Business Availability Center architecture. For details, see Chapter 2, "Using a Reverse Proxy in Mercury Business Availability Center."
- **Configuring the Mercury Business Availability Center platform to use SSL communication** – This chapter contains information on configuring each Mercury Business Availability Center component to support Secure Sockets Layer (SSL) communication. For details, see Chapter 3, "Using SSL in Mercury Business Availability Center."

- **Configuring the Mercury Business Availability Center platform to use basic authentication** – This chapter contains information on configuring each Mercury Business Availability Center component to support communication using the basic authentication protocol. For details, see Chapter 4, "Using Basic Authentication in Mercury Business Availability Center."

Note: For information on the ports used by each Mercury Business Availability Center component, see "Port Management" in *Deploying Servers*.

At various points in the hardening guidelines, you are instructed to configure the Web server on a Mercury Business Availability Center server machine to support required security settings—for example, when setting up SSL or basic authentication support. Instructions for configuring these settings can be found in the appropriate Web server documentation, available at the following sites:

- for IIS 5.0/6.0 – the Microsoft Web site (http://www.microsoft.com)
- for Apache – the Apache Jakarta Web site (http://jakarta.apache.org)

2

Using a Reverse Proxy in Mercury Business Availability Center

This chapter describes the security ramifications of reverse proxies and contains instructions for using a reverse proxy with Mercury Business Availability Center.

This chapter describes:	On page:
Overview of Reverse Proxies	8
Security Aspects of Using Reverse Proxies	8
Mercury Business Availability Center and Reverse Proxies	9
Specific and Generic Reverse Proxy Mode Support for Mercury Business Availability Center	11
Using a Reverse Proxy with a Single Machine Installation	13
Using a Reverse Proxy with a Distributed Server Installation	20

Overview of Reverse Proxies

A reverse proxy is an intermediate server that is positioned between the client machine and the Web server(s). To the client machine, the reverse proxy seems like a standard Web server that serves the client machine's HTTP protocol requests with no dedicated client configuration required.

The client machine sends ordinary requests for Web content, using the name of the reverse proxy instead of the name of a Web server. The reverse proxy then sends the request to one of the Web servers. Although the response is sent back to the client machine by the reverse proxy, it appears to the client machine as if it is being sent by the Web server.

Note: This chapter discusses only the security aspects of a reverse proxy. It does not discuss other aspects of reverse proxies, such as caching and load balancing.

Security Aspects of Using Reverse Proxies

A reverse proxy functions as a bastion host. It is configured as the only machine to be addressed directly by external clients, and thus obscures the rest of the internal network. Use of a reverse proxy enables the application server to be placed on a separate machine in the internal network, which is a significant security objective.

This chapter discusses the use of a reverse proxy in DMZ architecture, the more common security architecture available today.

DMZ is a network architecture in which an additional network is implemented, enabling you to isolate the internal network from the external one. Although there are a few common implementations of DMZs, this chapter discusses the use of a DMZ and reverse proxy in a back-to-back topology environment.

The following are the main security advantages of using a reverse proxy in such an environment:

- No DMZ protocol translation occurs. The incoming protocol and outgoing protocol are identical (only a header change occurs).
- Only HTTP access to the reverse proxy is allowed, which means that stateful packet inspection firewalls can better protect the communication.
- A static, restricted set of redirect requests can be defined on the reverse proxy.
- Most of the Web server security features are available on the reverse proxy (authentication methods, encryption, and more).
- The reverse proxy screens the IP addresses of the real servers as well as the architecture of the internal network.
- The only accessible client of the Web server is the reverse proxy.
- This configuration supports NAT firewalls (as opposed to other solutions).
- The reverse proxy requires a minimal number of open ports in the firewall.
- The reverse proxy provides good performance compared to other bastion solutions.

Mercury Business Availability Center and Reverse Proxies

Mercury Business Availability Center supports a reverse proxy in DMZ architecture. The reverse proxy is an HTTP mediator between the Mercury Business Availability Center data collectors/application users and the Mercury Business Availability Center server(s). If a reverse proxy is being used for application users, Mercury Business Availability Center must be configured to recognize its use. If not, the Mercury Business Availability Center URL optimization mechanism will not be able to properly calculate absolute paths. If a reverse proxy is being used for data collectors, only the data collectors and reverse proxy must be configured to recognize its use.

Mercury Business Availability Center servers can be installed using the following two architectures:

- **Single machine installation** – The Data Processing, Core, and Centers Servers reside on the same machine. To configure a reverse proxy for this architecture, see "Using a Reverse Proxy with a Single Machine Installation" on page 13.
- **Distributed server installation** – The Data Processing, Core, and Centers Servers reside on separate machines. To configure a reverse proxy for this architecture, see "Using a Reverse Proxy with a Distributed Server Installation" on page 20.

You can connect the following to Mercury Business Availability Center via a reverse proxy:

- **Mercury Business Availability Center data collectors**
- **Mercury Business Availability Center application users**

The following table describes the components that can be connected via a reverse proxy to each type of Mercury Business Availability Center server:

Mercury Business Availability Center Server	Mercury Business Availability Center Components Connecting Via Reverse Proxy to Server
Single machine installation (all servers installed on the same machine)	Data collectors Application users
Core Server	Data collectors
Centers Server	Application users

Specific and Generic Reverse Proxy Mode Support for Mercury Business Availability Center

Mercury Business Availability Center servers reply to application users by sending a base URL that is used to calculate the correct references in the HTML requested by the user. When a reverse proxy is used, Mercury Business Availability Center must be configured to return the reverse proxy base URL, instead of the Mercury Business Availability Center base URL, in the HTML with which it responds to the user. Note that if the reverse proxy is being used for data collectors only, configuration is required only on the data collectors and reverse proxy, and not on the Mercury Business Availability Center server(s).

You use the following three parameters to configure Mercury Business Availability Center support of a reverse proxy. These parameters are located on the Infrastructure Settings Manager page, accessible from Platform Administration's Setup and Maintenance tab, in the **Foundation** > **Platform Administration** context.

Parameter Name	Description
Enable Reverse Proxy	Controls whether the reverse proxy machine is enabled or disabled.
HTTP Reverse Proxy IPs	Defines the IP addresses of the reverse proxy or proxies used to communicate with the Centers Server(s). Note that if you define IP addresses for this parameter, Mercury Business Availability Center works in Specific Mode. If no IP addresses are defined for this parameter (the default option), Mercury Business Availability Center works in Generic Mode.
Default Virtual Centers Server URL	Defines the base URL of the machine (reverse proxy, load balancer, or other type of machine) to which the application user initially sends its request. Note that if the **Local Virtual Centers Server URL** parameter is defined for a specific machine, this URL will be used instead of the **Default Virtual Centers Server URL** for the specifically defined machine.

Note: Once you change the Mercury Business Availability Center base URL, it is assumed that the client is initiating HTTP sessions using the new base URL. You must therefore ensure that the HTTP channel from the client to the new URL is enabled.

Specific Mode

If you are working in this mode, each time an application user HTTP request causes Mercury Business Availability Center to calculate a base URL, the base URL is replaced with the value defined for either the **Default Virtual Centers Server URL** or the **Local Virtual Centers Server URL** (when defined), if the HTTP request came through one of the IP addresses defined for the **HTTP Reverse Proxy IPs** parameter. If the HTTP request did not come through one of these IP addresses, the base URL that Mercury Business Availability Center receives in the HTTP request is the base URL that is returned to the client.

Generic Mode

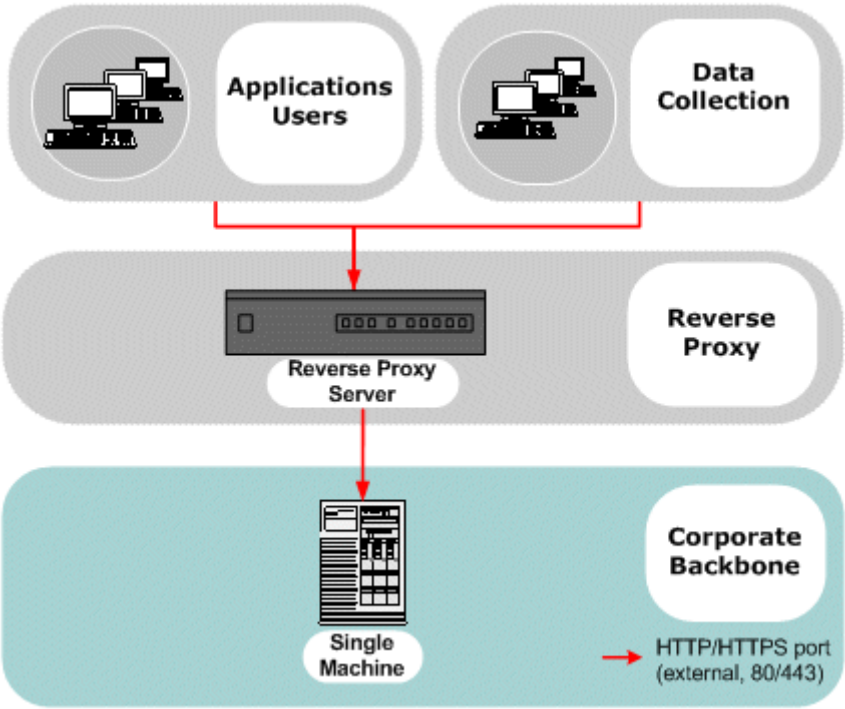
If you are working in this mode, each time an HTTP request causes the Mercury Business Availability Center application to calculate a base URL, the base URL is replaced with the value defined for either the **Default Virtual Centers Server URL** or the **Local Virtual Centers Server URL** (when defined).

Note that when using this mode, you must ensure that all Mercury Business Availability Center clients are accessing the Mercury Business Availability Center servers via the URL defined for the **Default Virtual Centers Server URL** or the **Local Virtual Centers Server URL** parameters.

Using a Reverse Proxy with a Single Machine Installation

This section contains the following information regarding the use of a reverse proxy when Mercury Business Availability Center Core, Centers, and Data Processing servers are installed on the same machine (as illustrated in the diagram below):

- Reverse Proxy Configuration – see page 14
- Mercury Business Availability Center-Specific Configuration – see page 17
- Limitations – see page 18
- Apache 2.0.x – Example Configuration – see page 19



Reverse Proxy Configuration

In this topology, all contexts must point to the same machine, on which the Mercury Business Availability Center servers are installed.

Reverse proxy Mercury Business Availability Center support should be configured differently in each of the following cases:

Scenario #	Mercury Business Availability Center Components Behind the Reverse Proxy
1	Data collectors (Business Process Monitor, Client Monitor, Real User Monitor, SiteScope)
2	Application users
3	Data collectors and application users

Note: Different types of reverse proxies require different configuration syntaxes. For an example of an Apache 2.0.x reverse proxy configuration, see "Apache 2.0.x – Example Configuration" on page 19.

Support for Mercury Business Availability Center Data Collectors

The following configuration is required if only data collectors are connected via a reverse proxy to your single machine Mercury Business Availability Center installation:

Requests for... on the Reverse Proxy Server	Proxy Request to be Served by:
/topaz/*	http://[Mercury Business Availability Center server]/topaz/*
/ext/*	http://[Mercury Business Availability Center server]/ext/*
/mam-collectors/*	http://[Mercury Business Availability Center server]/mam-collectors/*

Support for Mercury Business Availability Center Application Users

The following configuration is required if only application users are connected via a reverse proxy to your single machine Mercury Business Availability Center installation:

Requests for... on the Reverse Proxy Server	Proxy Request to be Served by:
/MercuryAM/*	http://[Mercury Business Availability Center server] /MercuryAM/*
/mercuryam/*	http://[Mercury Business Availability Center server] /mercuryam/*
/topaz/*	http://[Mercury Business Availability Center server] /topaz/*
/webinfra/*	http://[Mercury Business Availability Center server] /webinfra/*
/filters/*	http://[Mercury Business Availability Center server] /filters/*
/TopazSettings/*	http://[Mercury Business Availability Center server] /TopazSettings/*
/opal/*	[Mercury Business Availability Center server] /opal/*
/mam/*	[Mercury Business Availability Center server] /mam/*
/mam_images/*	[Mercury Business Availability Center server] /mam_images/*
/mcrs/*	[Mercury Business Availability Center server] /mcrs/*

Support for Both Mercury Business Availability Center Data Collectors and Application Users

The following configuration is required if both data collectors and application users are connected via a reverse proxy to your single machine Mercury Business Availability Center installation:

Requests for... on the Reverse Proxy Server	Proxy Request to be Served by:
/MercuryAM/*	http://[Mercury Business Availability Center server] /MercuryAM/*
/mercuryam/*	http://[Mercury Business Availability Center server] /mercuryam/*
/topaz/*	http://[Mercury Business Availability Center server] /topaz/*
/webinfra/*	http://[Mercury Business Availability Center server] /webinfra/*
/filters/*	http://[Mercury Business Availability Center server] /filters/*
/TopazSettings/*	http://[Mercury Business Availability Center server] /TopazSettings/*
/opal/*	[Mercury Business Availability Center server] /opal/*
/mam/*	[Mercury Business Availability Center server] /mam/*
/mam_images/*	[Mercury Business Availability Center server] /mam_images/*
/mcrs/*	[Mercury Business Availability Center server] /mcrs/*
/ext/*	[Mercury Business Availability Center server] /ext/*
/mam-collectors/*	[Mercury Business Availability Center server] /mam-collectors/*

For some reverse proxies, a reverse pass is also required. The reverse pass changes the HTTP headers returned from the server to relative headers. For an example of a reverse pass, see "Apache 2.0.x – Example Configuration" on page 19.

Mercury Business Availability Center-Specific Configuration

In addition to configuring the reverse proxy to work with Mercury Business Availability Center, you must configure Mercury Business Availability Center to work with the reverse proxy.

Note: Mercury Business Availability Center must be configured only if application users are connected via a reverse proxy to Mercury Business Availability Center. If the reverse proxy is being used for data collectors only, skip the instructions in this section.

To configure Mercury Business Availability Center to work with the reverse proxy:

1 On the Infrastructure Settings Manager page (accessible from Platform Administration's Setup and Maintenance tab), in the **Foundations** > **Platform Administration** context, set the following parameters:

- **Default Virtual Centers Server URL** – Verify that this parameter represents the URL of the machine (reverse proxy, load balancer, or other type of machine) used to access the Mercury Business Availability Center machine.
- **Local Virtual Centers Server URL** (optional) – If you must use more than one URL (the one defined for the **Default Virtual Centers Server URL** parameter) to access the Mercury Business Availability Center machine, define a **Local Virtual Centers Server URL** for each machine through which you want to access the Mercury Business Availability Center machine. If this parameter is set, the **Default Virtual Centers Server URL** is overridden.

- **HTTP Reverse Proxy IPs** (optional) – Configure the IP addresses of the reverse proxy or proxies used to communicate with the Mercury Business Availability Center machine. If the IP address of the reverse proxy sending the HTTP request is included in the list of IP addresses defined for this parameter, the URL returned to the client is either the **Default Virtual Centers Server URL** or the **Local Virtual Centers Server URL** (when defined). If the IP address of the reverse proxy sending the HTTP request is not included in the list of IP addresses defined for this parameter, the Mercury Business Availability Center machine returns the base URL that it receives in the HTTP request.

Note: If no IP addresses are defined for this parameter (the default option), Mercury Business Availability Center works in Generic Mode and the Mercury Business Availability Center machine returns the **Default Virtual Centers Server URL** or the **Local Virtual Centers Server URL** (when defined) to the client in all cases.

- **Enable Reverse Proxy** – Set this parameter to **true**. Note that this must be done after the above parameters have been configured.

2 Restart the Mercury Business Availability Center service on the Mercury Business Availability Center machine.

Note: Once you change the Mercury Business Availability Center base URL, it is assumed that the client is initiating HTTP sessions using the new base URL. You must therefore ensure that the HTTP channel from the client to the new URL is enabled.

Limitations

If you configured Mercury Business Availability Center to work in Generic Mode, all the Mercury Business Availability Center clients must access the Mercury Business Availability Center machine via the reverse proxy.

Apache 2.0.x – Example Configuration

Below is a sample configuration file that supports the use of an Apache 2.0.x reverse proxy in a case where both data collectors and application users are connecting to a single machine Mercury Business Availability Center installation.

Note: In the example below, the Mercury Business Availability Center machine's DNS name is **AM_server**.

---

1 Open the **<Apache machine root directory>\conf\httpd.conf** file.

2 Enable the following modules:

- **LoadModule proxy_module modules/mod_proxy.so**
- **LoadModule proxy_http_module modules/mod_proxy_http.so**

3 Add the following lines:

```
ProxyRequests off

<Proxy *>
        Order deny,allow
        Deny from all
        Allow from all
</Proxy>

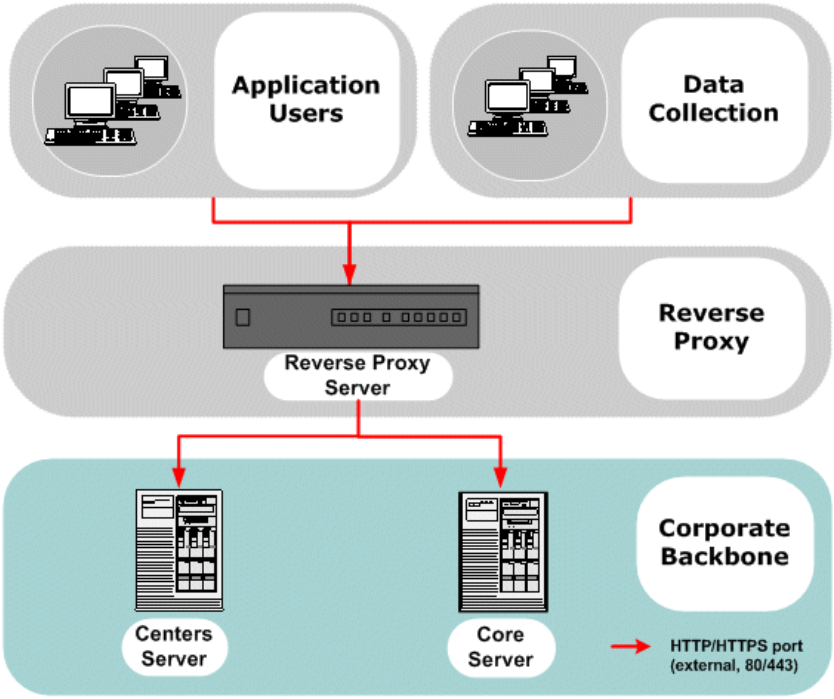
ProxyPass           /mercuryam      http://AM_server/mercuryam
ProxyPassReverse    /mercuryam      http://AM_server/mercuryam
ProxyPass           /MercuryAM      http://AM_server/MercuryAM
ProxyPassReverse    /MercuryAM      http://AM_server/MercuryAM
ProxyPass           /topaz          http://AM_server/topaz
ProxyPassReverse    /topaz          http://AM_server/topaz
ProxyPass           /ext            http://AM_server/ext
ProxyPassReverse    /ext            http://AM_server/ext
ProxyPass           /webinfra       http://AM_server/webinfra
ProxyPassReverse    /webinfra       http://AM_server/webinfra
```

ProxyPass	/filters	http://AM_server/filters
ProxyPassReverse	/filters	http://AM_server/filters
ProxyPass	/TopazSettings	http://AM_server/TopazSettings
ProxyPassReverse	/TopazSettings	http://AM_server/TopazSettings
ProxyPass	/opal	http://AM_server/opal
ProxyPassReverse	/opal	http://AM_server/opal
ProxyPass	/mam	http://AM_server/mam
ProxyPassReverse	/mam	http://AM_server/mam
ProxyPass	/mam_images	http://AM_server/mam_images
ProxyPassReverse	/mam_images	http://AM_server/mam_images
ProxyPass	/mam-collectors	http://AM_server/mam-collectors
ProxyPassReverse	/mam-collectors	http://AM_server/mam-collectors
ProxyPass	/mcrs	http://AM_server/mcrs
ProxyPassReverse	/mcrs	http://AM_server/mcrs

Using a Reverse Proxy with a Distributed Server Installation

This section contains the following information regarding the use of a reverse proxy when the Core Server and Centers Server are installed on separate machines (as illustrated in the diagram below):

- Reverse Proxy Configuration – see page 21
- Mercury Business Availability Center-Specific Configuration – see page 25
- Limitations – see page 27
- Apache 2.0.x – Example Configuration – see page 27



Reverse Proxy Configuration

In this topology, the reverse proxy context is divided into two sections:

- communication that is redirected to the Core Server
- communication that is redirected to the Centers Server

Reverse proxy Mercury Business Availability Center support should be configured differently in each of the following cases:

Scenario #	Mercury Business Availability Center Components Behind the Reverse Proxy
1	Data collectors (Business Process Monitor, Client Monitor, Real User Monitor, SiteScope)
2	Application users
3	Data collectors and application users

Note: Different reverse proxies require different configuration syntaxes. For an example of an Apache 2.0.x reverse proxy configuration, see "Apache 2.0.x – Example Configuration" on page 27.

Support for Mercury Business Availability Center Data Collectors

The following configuration is required on the reverse proxy for data collectors to connect via the reverse proxy to the Core Server:

Requests for... on the Reverse Proxy Server	Proxy Request to be Served by:
/topaz/*	http://[Core Server]/topaz/*
/ext/*	http://[Core Server]/ext/*
/mam-collectors/*	http://[Core Server]/mam-collectors/*

Support for Mercury Business Availability Center Application Users

The following configuration is required on the reverse proxy for application users to connect via the reverse proxy to the Centers Server:

Requests for... on the Reverse Proxy Server	Proxy Request to be Served by:
/MercuryAM/*	http://[Mercury Business Availability Center server] /MercuryAM/*
/mercuryam/*	http://[Mercury Business Availability Center server] /mercuryam/*
/topaz/*	http://[Mercury Business Availability Center server] /topaz/*
/webinfra/*	http://[Mercury Business Availability Center server] /webinfra/*
/filters/*	http://[Mercury Business Availability Center server] /filters/*
/TopazSettings/*	http://[Mercury Business Availability Center server] /TopazSettings/*
/opal/*	[Mercury Business Availability Center server] /opal/*
/mam/*	[Mercury Business Availability Center server] /mam/*
/mam_images/*	[Mercury Business Availability Center server] /mam_images/*
/mcrs/*	[Mercury Business Availability Center server] /mcrs/*

Support for Both Mercury Business Availability Center Data Collectors and Application Users

The following configuration is required on the reverse proxy if data collectors are connecting to the Core Server and application users are connecting to the Centers Servers via the same reverse proxy:

Priority	Requests for... on the Reverse Proxy Server	Proxy Request to be Served by:
1	/topaz/topaz_api/*	http://[Core Server]/topaz/topaz_api/*
1	/topaz/sitescope/*	http://[Core Server]/topaz/sitescope/*
1	/ext/*	http://[Core Server]/ext/*
1	/mam-collectors/*	http://[Core Server]/mam-collectors/*
2	/MercuryAM/*	http://[Mercury Business Availability Center server] /MercuryAM/*
2	/mercuryam/*	http://[Mercury Business Availability Center server] /mercuryam/*
2	/topaz/*	http://[Mercury Business Availability Center server] /topaz/*
2	/webinfra/*	http://[Mercury Business Availability Center server] /webinfra/*
2	/filters/*	http://[Mercury Business Availability Center server] /filters/*
2	/TopazSettings/*	http://[Mercury Business Availability Center server] /TopazSettings/*
2	/opal/*	[Mercury Business Availability Center server] /opal/*
2	/mam/*	[Mercury Business Availability Center server] /mam/*

Priority	Requests for... on the Reverse Proxy Server	Proxy Request to be Served by:
2	/mam_images/*	[Mercury Business Availability Center server] /mam_images/*
2	/mcrs/*	[Mercury Business Availability Center server] /mcrs/*

The priority column on the left means that the requests in priority 1 must be handled before those in priority 2. Make sure your reverse proxy supports priority handling logic, which enables a specific expression to be handled before a more generic one, if required. For example, the **/topaz/topaz_api/*** expression must be handled before the **/topaz/*** expression.

For some reverse proxies, a reverse pass is also required. The reverse pass changes the HTTP headers returned from the server to relative headers. For an example of a reverse pass, see "Apache 2.0.x – Example Configuration" on page 27.

Mercury Business Availability Center-Specific Configuration

In addition to configuring the reverse proxy to work with Mercury Business Availability Center, you must configure Mercury Business Availability Center to work with the reverse proxy.

Note: Mercury Business Availability Center must be configured only if application users are connected via a reverse proxy to Mercury Business Availability Center. If the reverse proxy is being used for data collectors only, skip the instructions in this section.

To configure Mercury Business Availability Center to work with the reverse proxy:

1 On the Infrastructure Settings Manager page (accessible from Platform Administration's Setup and Maintenance tab), in the **Foundations** > **Platform Administration** context, set the following parameters:

- **Default Virtual Centers Server URL** – Verify that this parameter represents the URL of the machine (reverse proxy, load balancer, or other type of machine) used to access the Centers Server.
- **Local Virtual Centers Server URL** (optional) – If you must use more than one URL (the one defined for the **Default Virtual Centers Server URL** parameter) to access the Centers Server, define a **Local Virtual Centers Server URL** for each machine through which you want to access the Centers Server. If this parameter is set, the **Default Virtual Centers Server URL** is overridden.
- **HTTP Reverse Proxy IPs** (optional) – Configure the IP addresses of the reverse proxy or proxies used to communicate with the Centers Server. If the IP address of the reverse proxy sending the HTTP request is included in the list of IP addresses defined for this parameter, the URL returned to the client is either the **Default Virtual Centers Server URL** or the **Local Virtual Centers Server URL** (when defined). If the IP address of the reverse proxy sending the HTTP request is not included in the list of IP addresses defined for this parameter, the Centers Server returns the base URL that it receives in the HTTP request.

Note: If no IP addresses are defined for this parameter (the default option), Mercury Business Availability Center works in Generic Mode and the Centers Server returns the **Default Virtual Centers Server URL** or the **Local Virtual Centers Server URL** (when defined) to the client in all cases.

- **Enable Reverse Proxy** – Set this parameter to **true**. Note that this must be done after the above parameters have been configured.

2 Restart the Mercury Business Availability Center service on the Mercury Business Availability Center machine.

Note: Once you change the Mercury Business Availability Center base URL, it is assumed that the client is initiating HTTP sessions using the new base URL. You must therefore ensure that the HTTP channel from the client to the new URL is enabled.

Limitations

If you configured Mercury Business Availability Center to work in Generic Mode, all the Mercury Business Availability Center clients must access the Mercury Business Availability Center servers via the reverse proxy.

Apache 2.0.x – Example Configuration

Below is a sample configuration file that supports the use of an Apache 2.0.x reverse proxy in a case where data collectors are connecting to the Core Server and application users are connecting to the Centers Servers through the same reverse proxy.

Note: In the example below, the Core Server DNS name is **AM_core** and the Centers Server DNS name is **AM_centers**.

---

1 Open the **<Apache machine root directory>\conf\httpd.conf** file.

2 Enable the following modules:

- **LoadModule proxy_module modules/mod_proxy.so**
- **LoadModule proxy_http_module modules/mod_proxy_http.so**

3 Add the following lines:

```
ProxyRequests off

<Proxy *>
        Order deny,allow
        Deny from all
        Allow from all
</Proxy>

ProxyPass               /ext                http://AM_core/ext
ProxyPassReverse        /ext                http://AM_core/ext
ProxyPass               /topaz/topaz_api    http://AM_core/topaz/topaz_api
ProxyPassReverse        /topaz/topaz_api    http://AM_core/topaz/topaz_api
ProxyPass               /topaz/sitescope    http://AM_core/topaz/sitescope
ProxyPassReverse        /topaz/sitescope    http://AM_core/topaz/sitescope
ProxyPass               /mam-collectors     http://AM_core/mam-collectors
ProxyPassReverse        /mam-collectors     http://AM_core/mam-collectors

ProxyPass               /mercuryam          http://AM_centers/mercuryam
ProxyPassReverse        /mercuryam          http://AM_centers/mercuryam
ProxyPass               /MercuryAM          http://AM_centers/MercuryAM
ProxyPassReverse        /MercuryAM          http://AM_centers/MercuryAM
ProxyPass               /topaz              http://AM_centers/topaz
ProxyPassReverse        /topaz              http://AM_centers/topaz
ProxyPass               /webinfra           http://AM_centers/webinfra
ProxyPassReverse        /webinfra           http://AM_centers/webinfra
ProxyPass               /filters            http://AM_centers/filters
ProxyPassReverse        /filters            http://AM_centers/filters
ProxyPass               /TopazSettings      http://AM_centers/TopazSettings
ProxyPassReverse        /TopazSettings      http://AM_centers/TopazSettings
ProxyPass               /opal               http://AM_centers/opal
ProxyPassReverse        /opal               http://AM_centers/opal
ProxyPass               /mam                http://AM_centers/mam
ProxyPassReverse        /mam                http://AM_centers/mam
ProxyPass               /mam_images         http://AM_centers/mam_images
ProxyPassReverse        /mam_images         http://AM_centers/mam_images
ProxyPass               /mcrs               http://AM_centers/mcrs
ProxyPassReverse        /mcrs               http://AM_centers/mcrs
```

3

Using SSL in Mercury Business Availability Center

This chapter describes how to configure your Mercury Business Availability Center platform to support communication using the Secure Sockets Layer (SSL) channel.

This chapter describes:	On page:
Introducing SSL Deployment in Mercury Business Availability Center	30
Mercury Business Availability Center Components Supporting SSL	33
SSL-Supported Topologies in Mercury Business Availability Center	34
Configuring SSL from the Application Users to the Centers Server	35
Configuring SSL from the Data Collectors to the Core Server	37
Configuring SSL From the Centers Server to Data Collectors	47
Configuring the Web Guard to Support SSL	57
Setting Java Runtime Environment to Work With Client/Server Certificates	58

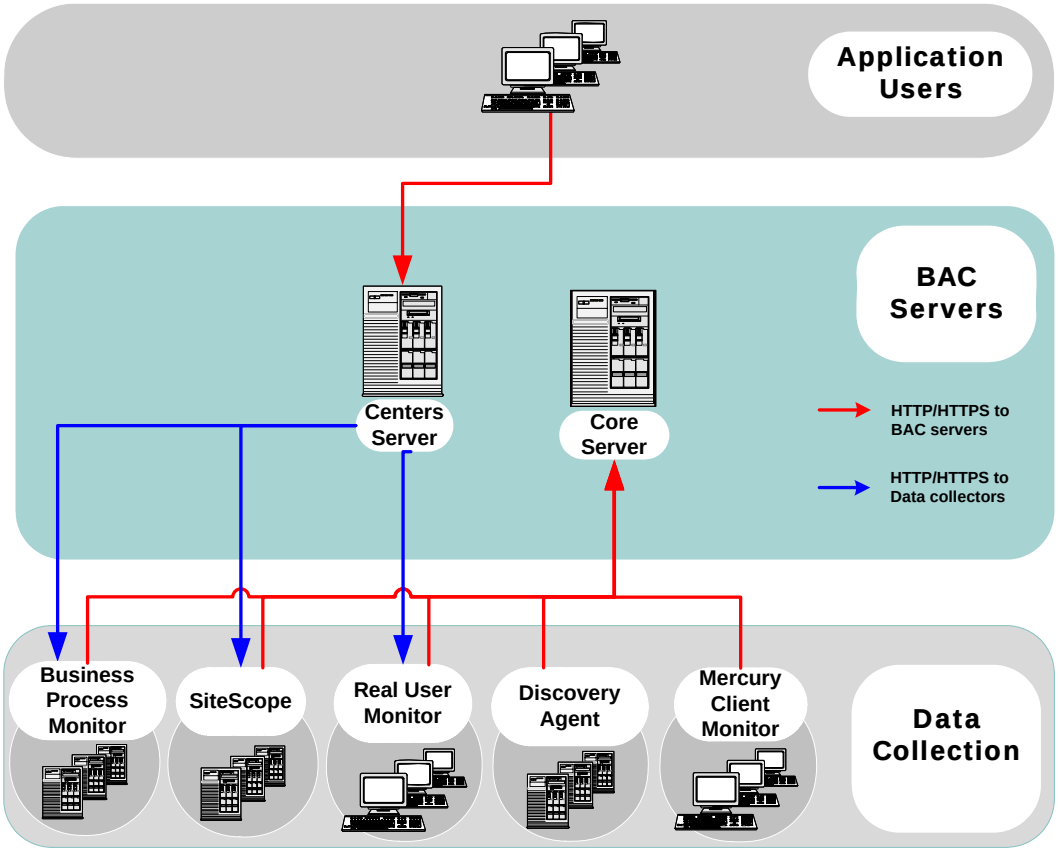
Introducing SSL Deployment in Mercury Business Availability Center

The Mercury Business Availability Center platform fully supports the SSL version 3.0 protocol. The SSL channel is configured on the Mercury Business Availability Center servers/clients as required.

SSL provides Mercury Business Availability Center with the following:

- **Server authentication** – provides authentication of the Mercury Business Availability Center server used for communication.
- **Client authentication** – provides authentication of the client communicating with the Mercury Business Availability Center server.
- **Encrypted channel** – encrypts the communication between the client and the server using a variety of ciphers.
- **Data integrity** – helps ensure that the information sent by one side over SSL is the same information received by the other side.

Possible SSL channels in Mercury Business Availability Center are illustrated in the following diagram:



Overview of Configuring SSL in Mercury Business Availability Center

The section "SSL-Supported Topologies in Mercury Business Availability Center" on page 34 discusses the various Mercury Business Availability Center-SSL topologies that are supported and provides links to each configuration step that is required.

Before proceeding with the configuration steps, ensure that:

- the Mercury Business Availability Center platform is operating as it is supposed to without an SSL channel
- you read this chapter in its entirety before you begin performing the configuration
- you define your secure communication requirements (only use an SSL channel where necessary)
- you consult the section "SSL-Supported Topologies in Mercury Business Availability Center" on page 34 to determine which topology is most suitable for the specific Mercury Business Availability Center-SSL architecture you are using

---

Note: The configuration specified for each Mercury Business Availability Center server is also relevant for a single machine installation, in which all the servers reside on the same machine.

Special SSL Configuration Considerations

The following points should be taken into consideration when configuring SSL in Mercury Business Availability Center:

- If the default or local virtual Centers Server URL has been configured to support HTTPS, you must set the Centers Server's JRE to trust the server-side certificate returned by the URL configured for the virtual Centers Server. For details on configuring the default and local virtual Centers Server URL, see Chapter 2, "Using a Reverse Proxy in Mercury Business Availability Center."

  For example, if you have configured the Centers Server to use a secure Reverse Proxy (HTTPS channel only) and have defined a URL of **https://myReverseProxy:443**, you import the certificate returned from the myReverseProxy Web server into the Mercury Business Availability Center Centers Server's JRE truststore.
- If you change your Web server to support SSL only (SSL required mode), the Web Guard must be configured to use SSL. For details on configuring the Web Guard, see "Configuring the Web Guard to Support SSL" on page 57.

➤ Business Process Monitors use certificates issued to the IP address of the Business Process Monitor Web server and not to the Web server name. For details on enabling SSL between the Centers Server and Business Process Monitors, see "Enabling SSL From the Centers Server to the Business Process Monitor Agent" on page 51.

Mercury Business Availability Center Components Supporting SSL

You set a Mercury Business Availability Center server to support SSL by configuring the Web server installed on the Mercury Business Availability Center server to support SSL.

You configure Mercury Business Availability Center clients to support SSL by defining the appropriate settings for each particular type of client, as described in the relevant client sections later in this chapter.

Note: For each client configuration, the HTTPS URL must match the SSL certificate common name that is used by the Web server for server-side authentication.

Mercury Business Availability Center Servers Supporting SSL

Mercury Business Availability Center Core and Centers servers require Web servers to communicate with their clients.

The servers can be configured to support SSL using one of the following Web servers:

	Microsoft IIS	Apache Web Server
Operating System	Windows 2000 Windows 2003	Windows 2000 Windows 2003

Mercury Business Availability Center Clients Supporting SSL

The following Mercury Business Availability Center clients support SSL communication with the Mercury Business Availability Center servers:

- **Browsers** – when used as Mercury Business Availability Center machine (when Mercury Business Availability Center is installed on a single machine) or Centers Server clients
- **Data collectors** – Business Process Monitor, Client Monitor, Real User Monitor, SiteScope, and Discovery Agent, when used as Mercury Business Availability Center machine (when Mercury Business Availability Center is installed on a single machine) or Core Server clients

SSL-Supported Topologies in Mercury Business Availability Center

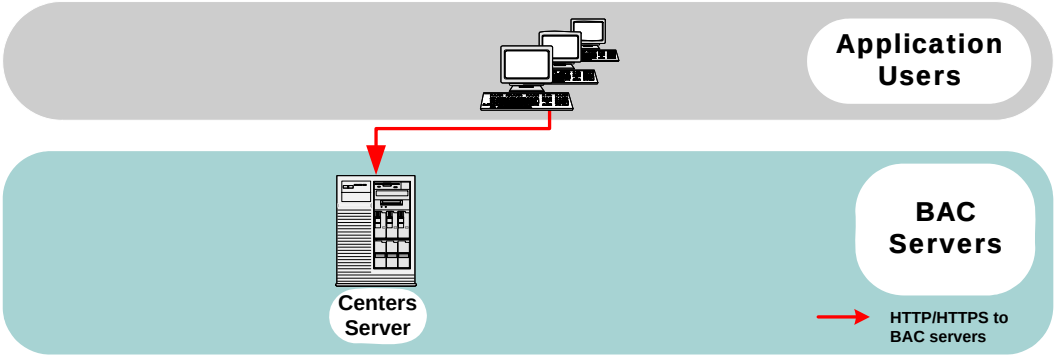
SSL optional topologies in Mercury Business Availability Center 6.0 are divided into two main categories:

- Interactive clients that communicate with Mercury Business Availability Center Centers servers using SSL.
- Data collectors that communicate with Mercury Business Availability Center Core servers using SSL.

  Client authentication using a client-side certificate is optional with Mercury Business Availability Center clients.

Configuring SSL from the Application Users to the Centers Server

The instructions in this section describe how to enable SSL from the application users to the Centers Server..



This section describes:

- SSL Configuration for the Centers Server – see below
- SSL Configuration for the Application Users – see page 36

SSL Configuration for the Centers Server

To configure a Mercury Business Availability Center Centers Server (or a Mercury Business Availability Center machine, in the case of a single machine installation) to support SSL, you must enable SSL support on the Web server used by the Centers Server.

To enable SSL support on the Web Server:

- **Microsoft Internet Information Server (IIS) 5.0 and 6.0** – See http://support.microsoft.com/default.aspx?scid=kb;EN-US;298805 for information on enabling SSL for all interaction with the Web server. Note that SSL should be enabled for the entire IIS Web Site under which you installed the Mercury Business Availability Center applications.

➤ **Apache HTTP Server 2.0.x** – See http://httpd.apache.org/docs-2.0/ssl/ for information on enabling SSL for all interaction with the Web server, using mod_ssl. SSL should be enabled for all the directories in use by Mercury Business Availability Center, as configured in the Apache configuration file (**httpd.conf**).

After performing the above procedures, the Web server installed on the Centers Server machine is configured to support HTTPS communication.

SSL Configuration for the Application Users

Mercury Business Availability Center application users (Centers Server clients) use Web browsers to communicate with the Centers Server. The Web browsers can be configured to support SSL.

When a session is started between the browser and the Centers Server, the Centers Server's Web server sends the browser a server-side certificate that was issued by a Certification Authority (CA). If the certificate used by the Web server is issued by a known CA, the certificate can generally be validated by the browser and no configuration is required. However, if the CA is not trusted by the browser, the browser machine must be configured to validate the server-side certificate that is sent. For instructions on setting CA certificate recognition in the browser and configuring browser certificate validation, refer to your browser vendor documentation.

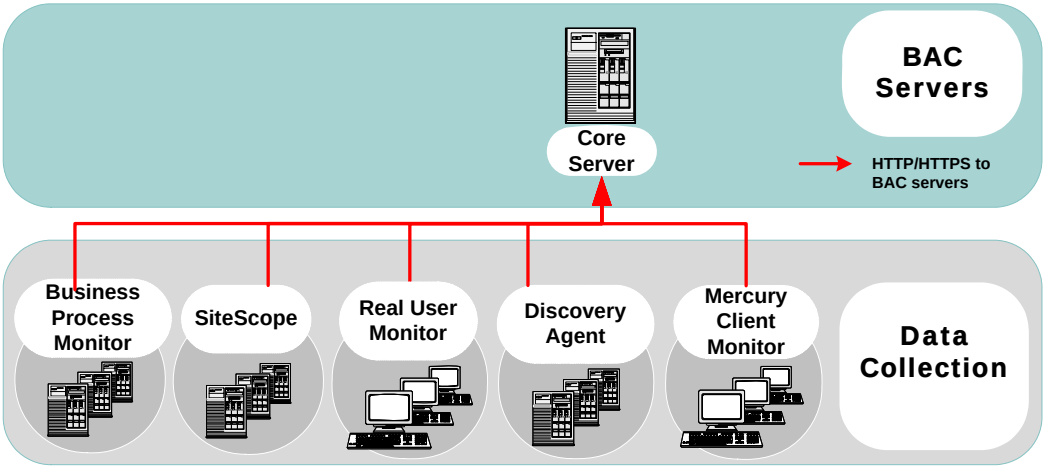
For example, if you are working with Internet Explorer 6.0, you can import a certificate to the truststore used by the browser.

To import a certificate to the truststore used by the browser:

1. Select **Tools** > **Internet Options** and click the **Content** tab.
2. Click the **Certificates** button.
3. In the **Trusted Root Certification Authorities** tab, click **Import**.
4. Link to the certificate you want to trust and import it.

Configuring SSL from the Data Collectors to the Core Server

The instructions in this section describe how to enable SSL from the Data collectors to the Mercury Business Availability Center Core Server.



This section describes:

- SSL Configuration for the Core Server – see below
- SSL Configuration for the Data Collectors – see page 38

SSL Configuration for the Core Server

To configure a Mercury Business Availability Center Core Server (or a Mercury Business Availability Center machine, in the case of a single machine installation) to support SSL, you must enable SSL support on the Web server used by the Core Server.

To enable SSL support on the Web Server:

- **Microsoft Internet Information Server (IIS) 5.0 and 6.0** – See http://support.microsoft.com/default.aspx?scid=kb;EN-US;298805 for information on enabling SSL for all interaction with the Web server. Note that SSL should be enabled for the entire IIS Web Site under which you installed the Mercury Business Availability Center applications.

➤ **Apache HTTP Server 2.0.x** – See http://httpd.apache.org/docs-2.0/ssl/ for information on enabling SSL for all interaction with the Web server, using mod_ssl. SSL should be enabled for the entire directories in use by Mercury Business Availability Center as configured in the Apache configuration file (**httpd.conf**):

After performing the above procedures, the Web server installed on the Core Server machine is configured to support HTTPS communication.

SSL Configuration for the Data Collectors

This section provides instructions for configuring the following Mercury Business Availability Center data collectors to support SSL:

➤ Business Process Monitor – see below
➤ Client Monitor Agent – see page 40
➤ SiteScope – see page 42
➤ Real User Monitor – see page 43
➤ Discovery Agent – see page 45

Note: These instructions are necessary only if the Core Server with which the data collector is communicating requires SSL.

Business Process Monitor

Configuring SSL support for the Business Process Monitor involves the following procedures:

➤ Configuring a Connection to the Core Server Using SSL – see below
➤ Configuring an SSL Client-Side Certificate – see page 40

Configuring a Connection to the Core Server Using SSL

When a session is started between the Business Process Monitor and the Core Server, the Core Server sends the Business Process Monitor a server-side certificate that was issued by a Certification Authority (CA). The Business Process Monitor instance should be configured to trust the CA and to communicate via SSL.

To configure the Business Process Monitor to connect to the Core Server using SSL:

1. Obtain the truststore file in PEM format, base64 encoded. The file can consist of the server-side certificate itself, or the certificate of the CA that issued the server-side certificate, or all certificates required for the trust path (all certificates must be placed in the same PEM file).
2. Open Business Process Monitor Admin (**http://<Business Process Monitor machine>:2696**).
3. In the Business Process Monitor page, identify the Business Process Monitor instance you want to configure from the **Instances** list and click the **Edit** button for the instance. The Edit Instance page opens.



4. In the **General** section, change the Core Server URL to: **HTTPS://<Core Server URL>/topaz/**.

Important: The URL must end with **/topaz** and not **/MercuryAM**.

---

5. In the **SSL** section, configure the **SSL authority certificate file** to point to the truststore file (so that the Business Process Monitor instance recognizes the file). The file must be in PEM format and base64 encoded.
6. Click **Save Changes and Restart Instance**.

Configuring an SSL Client-Side Certificate

If the Core Server is supporting SSL with client-side certificates, you must configure a client-side certificate for the Business Process Monitor instance.

To configure a client-side certificate on the Business Process Monitor machine:

1 Open Business Process Monitor Admin (**http://<Business Process Monitor machine>:2696**).

2 In the Business Process Monitor page, identify the Business Process Monitor instance for which you want to use SSL from the **Instances** list and click the **Edit** button for the instance. The Edit Instance page opens.



3 Enter the following SSL parameter values:

- **SSL client certificate file** – the path of the PEM file that holds the client-side certificate
- **SSL private key file** – the path of the PEM file that holds the private key used as a public/private pair key for the public key in the client-side certificate
- **SSL private key password** – the password of the private key, if the private key was encrypted with a password

4 Click **Save Changes and Restart Instance**.

Client Monitor Agent

Configuring SSL support for the Client Monitor Agent involves the following procedures:

- Configuring a Connection to the Core Server Using SSL – see below
- Configuring an SSL Client-Side Certificate – see page 41

Configuring a Connection to the Core Server Using SSL

When a session is started between the Client Monitor Agent machine and the Core Server, the Core Server sends the Client Monitor Agent machine a server-side certificate that was issued by a Certification Authority (CA) recognized by the Core Server. The Client Monitor Agent machine should be configured to trust the CA and to communicate via SSL.

To configure the Client Monitor Agent to connect to the Core Server using SSL:

1 Install the CA certificate on the machine that runs the Client Monitor Agent. The certificate can be located in the local machine's certificate store, or in the certificate store of the user executing the Client Monitor Agent. If you are using a certificate from a well-known authority, it is most likely already available in the certificate store on your machine.

Note: For detailed information on how to install certificates in the Windows certificate store, refer to the Microsoft Web site (http://www.microsoft.com).

---

2 Open the Client Monitor Agent Settings dialog box (**Start** > **Programs** > **Mercury Client Monitor** > **Client Monitor Agent Settings**).

3 In the Core Server URL box, change the **HTTP** in the URL to **HTTPS**.

4 Click **Save & Exit**.

5 Restart the Client Monitor Agent.

Configuring an SSL Client-Side Certificate

If the Core Server is supporting SSL with client-side certificates, you must configure a client-side certificate for the Client Monitor Agent.

To configure a client-side certificate on the Client Monitor Agent machine:

1 Install a client-side certificate in your machine certificate store registry from an authority that is recognized by the Core Server machine. Note that the client-side certificate can be installed in the Windows user's personal certificate store. Make sure you install both the certificate and its corresponding private key.

2 Import the file containing both the certificate and the private key into the user's certificate store by double-clicking the file and following the instructions in the wizard that opens.

3 Verify that the Core Server can validate the client-side certificate by either importing it to the Core truststore, or trusting the CA that issued the client side certificate.

Note: If Client Monitor Agents have already been distributed to clients, they must be redistributed once the above changes have been made.

SiteScope

If the SiteScope machine is required to communicate with the Core Server via SSL, the SiteScope machine must be configured to connect to the Core Server using SSL.

This section details how to enable an SSL connection from SiteScope to the Core Server using the Mercury Business Availability Center Monitor Administration pages, or directly via the SiteScope Administration pages.

Import the certificate/CA certificate used by the Core Server(s) into the SiteScope truststore

SiteScope uses its Java Runtime Environment (JRE) to communicate with the Core Server using SSL. To be able to validate the certificate coming from the Core Server by the JRE used in SiteScope, the certificate, or its issuer, must be trusted by the JRE.

SiteScope's JRE uses a truststore (a store of trusted CAs and certificates) which is located in the file:

<SiteScope root directory>\JRE\lib\security\cacerts

By default, the **cacerts** file contains common CA certificates, so if the Core Server is using a certificate issued by a known issuer, it is likely that no import operation to the truststore will be needed.

If the Core Server is using a certificate issued by an unknown CA, or it is using a self-signed certificate, you must import the certificate used by the Core Server, or the CA certification path that issued the certificate, to the truststore.

Note: The keystore used can be in either PKCS12, or JKS format.

To import a required certificate

1. Ensure that the certificate to import is in PEM encoding.
2. Use the **keytool.exe** utility, located in the <**Mercury Business Availability Center server root directory**>**/JRE/bin** directory.

   The import command should be similar to the following:

   ```
   keytool -import -keystore <SiteScope root directory>\JRE\lib\security\cacerts -storepass changeit -alias myCoreServerCertificate -trustcacerts -file CoreServerCert.pem
   ```

   where **CoreServerCert.pem** is the Core Server certificate sent by the Web server for server-side authentication, or the CA certificate that issued it.

To configure SiteScope for SSL using Mercury Business Availability Center Monitor Administration

1. In the Mercury Business Availability Center monitor tree, right-click the SiteScope object for which you want to configure SSL and select **Edit**.
2. In the Profile Settings section of the Edit SiteScope page, select the **Use SSL** check box.
3. Click **OK** at the bottom of the page.
4. Restart the SiteScope instance.

To configure SiteScope for SSL using the SiteScope interface

1. Select **Preferences** > **Mercury BAC**.
2. In the Mercury Business Availability Center Server Registration page, in the **Optional Settings** section, select the **Use SSL (HTTPS protocol)** check box.
3. Click the **Update** button at the bottom of the page.
4. Restart the SiteScope instance.

Real User Monitor

Configuring SSL support for Real User Monitor involves the following procedures:

- Configuring a Connection to the Core Server Using SSL – see below
- Configuring an SSL Client-Side Certificate – see page 44

Configuring a Connection to the Core Server Using SSL

When a session is started between the Real User Monitor engine and the Core Server, the Core Server sends the Real User Monitor engine a server-side certificate that was issued by a Certification Authority (CA) recognized by the Core Server. The Real User Monitor engine should be configured to trust the CA and to communicate via SSL.

To configure Real User Monitor to connect to the Core Server using SSL:

1. Obtain the truststore file in PEM format, base64 encoded. The file can consist of the server-side certificate itself, or the certificate of the CA that issued the server-side certificate, or all certificates required for the trust path (all certificates must be placed in the same PEM file).
2. Open the Real User Monitor Web Console (**http://<Real User Monitor engine name>:8180**).
3. Click the **Configuration** tab.
4. Under **General**, change the value of the Mercury Business Availability Center URL to:

   HTTPS://<Core Server URL>/topaz/

5. Under **SSL**, select the **Use SSL** check box. In the **SSL certificate authority file** box, enter the path to the truststore file mentioned above.
6. Click **Save Configuration**.

Configuring an SSL Client-Side Certificate

If the Core Server is supporting SSL with client-side certificates, you must configure a client-side certificate for the Real User Monitor engine.

To configure a client-side certificate on the Real User Monitor engine:

1. Open the Real User Monitor Web Console (**http://<Real User Monitor engine name>:8180**).
2. Click the **Configuration** tab.
3. Under **SSL**, ensure that the **Use SSL** check box is selected and that you completed the above configuration.

4 Enter the following SSL parameter values:

- **SSL client certificate file** – the path of the PEM base64 encoded file that holds the client-side certificate.
- **SSL private key file** – the path of the PEM base64 encoded file that holds the private key used as a public/private pair key for the public key in the client-side certificate.
- **SSL private key password** – the password of the private key, if the private key was encrypted with a password.

5 Click Save Configuration.

Discovery Agent

When a session is started between the Discovery Agent and the Core Server, the Core Server sends the Agent a server-side certificate that was issued by a Certification Authority (CA) recognized by the Core Server. The Discovery Agent engine should be configured to trust the CA and to communicate via SSL.

To configure the Discovery Agent to connect to the Core Server using SSL:

1 Set the Discovery Agent JRE to trust the Core Server certificate.

The Discovery Agent truststore file is located at:

%discovery root%\root\lib\security\MAMTrustStoreExp.jks

If the certificate used by the Mercury Business Availability Center server was not issued by a known CA, you need to import the certificate, or the CA's certificate, to the truststore.

The certificate imported should be in PEM format, base64 encoded.

A sample command for importing a server-side certificate is:

```
keytool -import -keystore %discovery
root%\root\lib\security\MAMTrustStoreExp.jks -storepass <password> -alias
“Center Server certificate” -trustcacerts -file myCentersServerCertificate.pem
```

where **myCentersServerCertificate.pem** is the Centers Server certificate sent by Mercury Business Availability Center JVM to the Agent, or the Certification Authority (CA) certificate issued it.

2 Set the new connection parameters in the Discovery Agent.

- Open the file %discovery root%\root\lib\collectors\appilog-remote.properties.
- Configure the URL of the Mercury Business Availability Center server:

  serverIP = <Mercury Business Availability Center server Domain Name>

Note: The SSL connection will probably fail if an IP number is set instead of domain name.

- Configure the port number to use for HTTPS:

  serverPortHttps = <HTTPS port number>

- Set the schema to be used by the Agent to HTTPS:

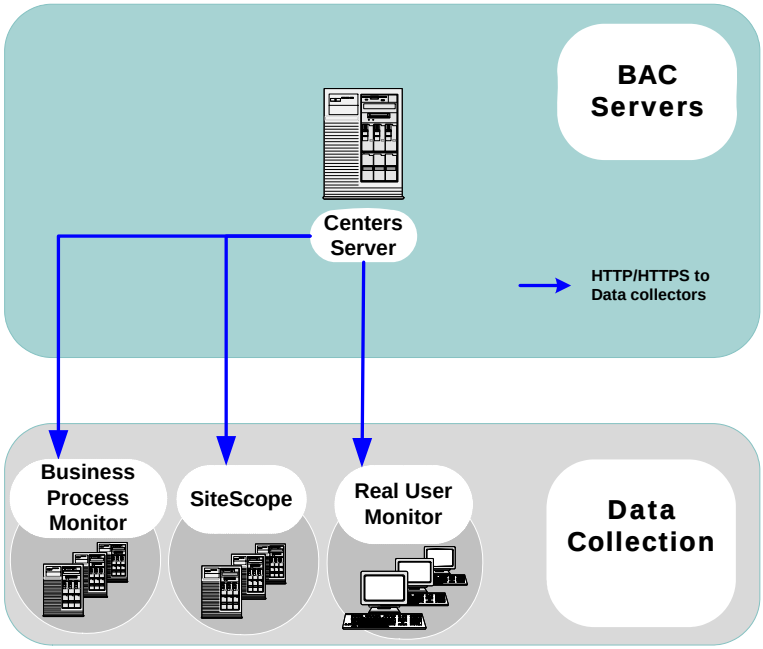
  appilog.agent.probe.protocol = HTTPS

3 Restart the Discovery Agent.

Configuring SSL From the Centers Server to Data Collectors

The instructions in this section describe how to enable SSL from the Centers Server to Business Process Monitor, SiteScope, and Real User Monitor data collectors.

Note: In this situation, the Centers Server acts as a client connecting to the data collector using SSL (if required by the data collector).



This section describes:

- Enabling SSL From the Centers Server to SiteScope – see below
- Enabling SSL From the Centers Server to the Business Process Monitor Agent – see page 51
- Enabling SSL From the Centers Server to the Real User Monitor Engine – see page 54

Enabling SSL From the Centers Server to SiteScope

To enable the Centers Server to communicate with SiteScope using SSL, you must configure the SiteScope monitor to support SSL, configure the Centers Server's Java Runtime Environment (JRE) to trust the SiteScope certificate, and set Mercury Business Availability Center to use HTTPS to connect to the SiteScope monitor. In addition, if the SiteScope Web server has been configured to force client-side authentication, you must add a client-side certificate to Mercury Business Availability Center's keystore.

Configuring the SiteScope Web server to support SSL

To enable a SiteScope monitor to communicate using SSL, you must configure Tomcat 5.0.x to support HTTPS. For detailed information on configuring Tomcat 5.0.x, refer to http://jakarta.apache.org/tomcat/tomcat-5.0-doc/ssl-howto.html.

To configure Tomcat 5.0.x to support HTTPS:

1 Uncomment the following connector element:

```
<Connector className="org.apache.coyote.tomcat5.CoyoteConnector"
port="8443" minProcessors="5" maxProcessors="75" enableLookups="true"
disableUploadTimeout="true" acceptCount="100" debug="0" scheme="https"
secure="true"; clientAuth="false" sslProtocol="TLS"/>
```

Note: If you are not using the default port number 8443 for the SiteScope SSL communications, change the port number in the connector element accordingly.

---

2 Add the following attribute to the connector element:

keystoreFile="myKeyStore"

where myKeyStore is the JKS file that contains the Web server certificate and a corresponding private key.

Note: You can create a self-signed certificate for testing, using the keytool.exe utility, as described in the Jakarta on-line document on SSL (http://jakarta.apache.org/tomcat/tomcat-5.0-doc/ssl-howto.html).

---

3 If the keystore password is different from the default password **changeit**, you must configure it accordingly in the connector element:

keystorePass="your password"

4 Restart Tomcat.

Note: There is an additional HTTP channel that can be used for the direct administration of SiteScope. This channel (which uses the default port 8888) can also be configured to support SSL. For details, see "SiteScope Profile Integration Status" in *Managing SiteScope*.

Configuring the Centers Server's JRE to trust the SiteScope certificate

You need to configure the JRE used by the Centers Server to trust the certificate sent by the SiteScope Web server (for details, see "Setting JRE to Trust a Client/Server Certificate" on page 58).

You must import the SiteScope server-side certificate into the truststore file used by Mercury Business Availability Center. The truststore file is **%mercury_root%/JRE/lib/security/cacerts** and it is a JKS type file.

Tip: You can use the keytool.exe utility to import the SiteScope server-side certificate.

Configuring Mercury Business Availability Center to use HTTPS to connect to a SiteScope monitor

In monitor administration, right-click the SiteScope profile you want to configure in the monitors tree and select **Edit**.

On the Edit SiteScope page, under **Main Settings**, enter the following:

- Check the Use SSL check box.
- Change the port number to the one used by the SSL server.

Adding a client-side certificate to Mercury Business Availability Center's keystore

If the SiteScope Web server has been configured to force client-side authentication you must add a client-side certificate that can be sent to SiteScope, to Mercury Business Availability Center's keystore.

To add a client-side certificate:

1. Set the Mercury Business Availability Center Java Virtual Machine (JVM) to support client-side authentication. For details, see "Setting Java Runtime Environment to Work With Client/Server Certificates" on page 58.

Note: You must define the keystore used byMercury Business Availability Center as described in "Setting Java Runtime Environment to Work With Client/Server Certificates" on page 58.

---

2. Configure SiteScope to trust Mercury Business Availability Center's client-side certificate.

   To configure SiteScope to trust Mercury Business Availability Center's client-side certificate, you must set the Tomcat used by SiteScope to trust the client-side certificate sent by Mercury Business Availability Center. For details, refer to http://jakarta.apache.org/tomcat/tomcat-5.0-doc/ssl-howto.html.

Add the following attributes to the Tomcat HTTPS connector element:

```
<Connector className="org.apache.coyote.tomcat5.CoyoteConnector"
port="8443" minProcessors="5" maxProcessors="75" enableLookups="true"
disableUploadTimeout="true" acceptCount="100" debug="0" scheme="https"
secure="true"; clientAuth="true"
```

- **truststoreFile="your truststore"**
- **truststorePass="truststore password"** (if different to the keystore password)

The default truststore used by Tomcat is **<site scope root>\java\lib\security\cacerts**. You can set a different truststore, or import the client-side certificate used by Mercury Business Availability Center into this **cacerts** file. For details, see "Setting JRE to Trust a Client/Server Certificate" on page 58.

A sample command for importing a client-side certificate is:

```
keytool -import -keystore <SiteScope root directory>\JRE\lib\security\cacerts -
storepass changeit -alias "Centers Server Client Certificate" -trustcacerts -file
myCentersServerClientCertificate.pem
```

where **myCentersServerClientCertificate.pem** is the Centers Server client certificate sent by Mercury Business Availability Center JVM for client-side authentication, or the Certification Authority (CA) certificate issued it.

Enabling SSL From the Centers Server to the Business Process Monitor Agent

To enable the Centers Server to communicate with a Business Process Monitor using SSL, you must configure the Business Process Monitor to support SSL, set Mercury Business Availability Center to use HTTPS to connect to the Business Process Monitor, and configure the Centers Server's Java Runtime Environment (JRE) to trust the Business Process Monitor certificate.

Configuring a Business Process Monitor Web server to support SSL

To enable a Business Process Monitor Web server to support SSL, carry out the following steps:

1. Stop the Business Process Monitor and make sure that all processes are stopped.
2. Open the <**Business Process Monitor root directory**>**ServletContainer****conf****server.xml** file in a text editor.
3. Locate the XML Connector element that is not commented out and comment it out. For example, change:

   ```
   <Connector className="org.apache.catalina.connector.http.HttpConnector"
   port="2696" minProcessors="5" maxProcessors="75" enableLookups="true"
   redirectPort="8443" acceptCount="10" debug="0" connectionTimeout="60000"/>
   ```

   to:

   ```
   <!--<Connector className="org.apache.catalina.connector.http.HttpConnector"
   port="2696" minProcessors="5" maxProcessors="75" enableLookups="true"
   redirectPort="8443" acceptCount="10" debug="0"
   connectionTimeout="60000"/>-->
   ```

4. Locate the XML Connector element with an attribute scheme set to **https** and uncomment it. For example, change:

   ```
   <!--<Connector className="org.apache.catalina.connector.http.HttpConnector"
   port="8443" minProcessors="5" maxProcessors="75" enableLookups="true"
   acceptCount="10" debug="0" scheme="https" secure="true">
   <Factory className="org.apache.catalina.net.SSLServerSocketFactory"
   clientAuth="false" protocol="TLS"/>
   </Connector>-->
   ```

   to:

   ```
   <Connector className="org.apache.catalina.connector.http.HttpConnector"
   port="8443" minProcessors="5" maxProcessors="75" enableLookups="true"
   acceptCount="10" debug="0" scheme="https" secure="true">
   <Factory className="org.apache.catalina.net.SSLServerSocketFactory"
   clientAuth="false" protocol="TLS"/>
   </Connector>
   ```

5 Save the <**Business Process Monitor root directory**>**ServletContainer****conf****server.xml** file.

6 Create a keystore certificate by running the following command:

- **For Windows** – <Business Process Monitor root directory>\JRE\bin\keytool -genkey -alias tomcat -keyalg RSA
- **For Solaris/Linux** – <Business Process Monitor root directory>/JRE/bin/keytool -genkey -alias tomcat -keyalg RSA

7 When prompted for the keystore password, enter **changeit** (all lower case). To choose a different password, see the Tomcat documentation (http://jakarta.apache.org/tomcat/tomcat-4.0-doc/ssl-howto.html).

8 Enter general information about the certificate when prompted for this information.

9 When prompted for the key password for the certificate, use the same password you used previously for the keystore.

10 Copy the **.keystore** file that was created in the home directory of the user with which you ran the above command, to the home directory of the user running Business Process Monitor Admin (by default, the **default** user in Windows, the **root** user in UNIX).

11 If you are working on a Windows platform, select **Start** > **Programs** > **Mercury Business Process Monitor**, then right-click the **Business Process Monitor Admin** link. Select **Properties** from the displayed menu to open the Business Process Monitor Admin Properties window. In the Web Document tab, change the URL box to https://localhost:8443/bpm, then click **Apply**.

12 Restart Business Process Monitor.

Configuring Mercury Business Availability Center to use HTTPS to connect to a Business Process Monitor

The Business Process Monitor sends the Centers Server its parameters—Port, URL, and Schema (HTTP/S)—every few hours. These parameters are automatically discovered by the Business Process Monitor according to the Tomcat configuration done above. The Centers server will use these parameters to communicate with the Business Process Monitor. It is not necessary to manually configure the Centers server.

Note: The certificate at the Business Process Monitor must be issued to the IP of the Business Process Monitor Web server and not to the Web server name.

Configuring the Centers Server's JRE to trust the Business Process Monitor certificate

You must configure the JRE used by the Centers Server to trust the certificate sent by the Business Process Monitor Web server (for details, see "Setting JRE to Trust a Client/Server Certificate" on page 58).

You must import the Business Process Monitor server-side certificate into the truststore file used by Mercury Business Availability Center. The truststore file is **%mercury_root%/JRE/lib/security/cacerts** and it is a JKS type file.

Tip: You can use the keytool.exe utility to import the Business Process Monitor server-side certificate.

Enabling SSL From the Centers Server to the Real User Monitor Engine

To enable the Centers Server to communicate with Real User Monitor using SSL, you must configure Real User Monitor to support SSL, configure the Centers Server's Java Runtime Environment (JRE) to trust the Real User Monitor certificate, and set Mercury Business Availability Center to use HTTPS to connect to Real User Monitor.

Configuring the Real User Monitor Web server to support SSL

To enable a Real User Monitor engine to support SSL communication, you must configure Tomcat 5.0.x to support HTTPS. For detailed information on configuring Tomcat 5.0.x, refer to http://jakarta.apache.org/tomcat/tomcat-5.0-doc/ssl-howto.html.

To configure Tomcat 5.0.x to support HTTPS:

1 Uncomment the following connector element:

<Connector className="org.apache.coyote.tomcat5.CoyoteConnector" port="8443" minProcessors="5" maxProcessors="75" enableLookups="true" disableUploadTimeout="true" acceptCount="100" debug="0" scheme="https" secure="true"; clientAuth="false" sslProtocol="TLS"/>

Note: If you are not using the default port number 8443 for the Real User Monitor SSL communications, change the port number in the connector element accordingly.

---

2 Add the following attribute to the connector element:

keystoreFile="myKeyStore"

where myKeyStore is the JKS file that contains the Web server certificate and a corresponding private key.

Note: You can create a self-signed certificate for testing, using the keytool.exe utility, as described in the Jakarta on-line document on SSL (http://jakarta.apache.org/tomcat/tomcat-5.0-doc/ssl-howto.html).

---

3 If the keystore password is different from the default password **changeit**, you must configure it accordingly in the connector element:

keystorePass="your password"

4 Restart Tomcat.

Configuring the Centers Server's JRE to trust the Real User Monitor certificate

You must configure the JRE used by the Centers Server to trust the certificate sent by the Real User Monitor Web server (for details, see "Setting JRE to Trust a Client/Server Certificate" on page 58).

You must import the Real User Monitor server-side certificate into the truststore file used by Mercury Business Availability Center. The truststore file is **%mercury_root%/JRE/lib/security/cacerts** and it is a JKS type file.

Tip: You can use the keytool.exe utility to import the Real User Monitor server-side certificate.

Configuring the Real User Monitor URL in Mercury Business Availability Center for HTTPS

You must configure the URL of the Real User Monitor engine defined in Mercury Business Availability Center Monitor Administration to include the HTTPS protocol.

To configure the Real User Monitor URL defined in Mercury Business Availability Center Monitor Administration for HTTPS:

1. In Mercury Business Availability Center Monitor Administration, right-click the Real User Monitor engine object you want to configure and select Edit.
2. Open the Advanced Settings section.
3. Change the Real User Monitor Database URL to:

   https://<RUM domain name>:<HTTPS port number>

   where:

   - RUM domain name – is the fully qualified domain name of the Real User Monitor engine
   - HTTPS port number – is the port number used for HTTPS in the Real User Monitor Web server

Configuring the Web Guard to Support SSL

Web Guard is a component in each Mercury Business Availability Center server that tracks the validity of the Mercury Business Availability Center components using HTTP/HTTPS. If you changed your Web server to support only SSL (SSL required mode), the Web Guard must be configured to use SSL.

To configure the Web Guard to use SSL, you must:

- Set the Web Guard's Configuration File to Support SSL – see below
- Set the Web Guard JRE to Support SSL – see page 57

Set the Web Guard's Configuration File to Support SSL

If your Web server supports only SSL, you must configure the Web Guard's configuration file to support SSL.

To configure the Web Guard's configuration file to support SSL:

1. Open the <**Mercury Business Availability Center server root directory**> **/conf/webserver_guard.conf** file.
2. Add the following lines to the bottom of the file:

```
ssl=1
host_name=<host name>
webserver_port=<SSL port number>
```

Set the Web Guard JRE to Support SSL

The Web Guard uses Mercury Business Availability Center servers' JRE to support SSL.

The truststore, which contains the Certification Authorities (CAs) to be trusted by the Web Guard JRE, enables the Web Guard JRE on each Mercury Business Availability Center server to communicate with the Web server(s) requiring SSL. The truststore to be used in the procedure below is: **<Mercury Business Availability Center server root directory>\JRE\lib\security\cacerts**.

For details on how to set the Web Guard JRE to support SSL, see "Setting JRE to Trust a Client/Server Certificate" on page 58.

To enable the JRE, validate the certificate used by the Mercury Business Availability Center Web server.

If you configure the Web server on the Mercury Business Availability Center server to require client authentication as well (an optional SSL handshake setting), the Web Guard JRE must be configured to send a client-side certificate when connecting to the Web server requiring SSL.

To enable the JRE to send a client-side certificate, see "Setting JRE to Use Client/Server-Side Authentication" on page 60.

Setting Java Runtime Environment to Work With Client/Server Certificates

To set the Java Runtime Environment (JRE) to work with client/server certificates, you must set the JRE to trust a client/server certificate and to use client/server-side authentication.

Setting JRE to Trust a Client/Server Certificate

When the JRE is used to connect to an SSL Web server, or whenever it accepts a client-side certificate, it must be able to validate and trust the certificate to establish the SSL session.

To trust and validate a certificate, JRE uses a trusted certificates store called a truststore. If the JRE can find a certificate in its truststore that is identical to the certificate requiring validation, validation is completed and the establishment of the session continues. Otherwise, the JRE will try to validate the digital signature of the certificate signed by the certificate issuer, using the issuing chain.

In order to validate a certificate signed by an issuer, or chain, the issuer's certificate must be included in the truststore used by the JRE. A certificate issuer is a Certification Authority (CA) that signs certificates. If you import the certificate of the CA into the JRE truststore, each certificate issued by this CA can be validated by the JRE.

If the JRE is trying to validate a self-signed certificate (a certificate that is issued by itself), it must import the specific certificate into the JRE truststore.

Configuring the truststore

The following are applicable to the truststore:

- The default truststore file used by the JRE is **<jre root directory>\lib\security\cacerts**
- The cacerts file type is JKS (Java Key Store)
- You can set the truststore used by your JRE instance by adding two system properties to the JVM as parameters:
  - -Djavax.net.ssl.trustStore=<your truststore>
  - -Djavax.net.ssl.trustStorePassword=<your truststore password>

To enable your JRE to validate a certificate, you must import the certificate, or the certificate chain, to the truststore used by your JRE.

To import a required certificate to the truststore:

Add the required certificate or certificate chains to the truststore in PEM format using the **keytool.exe** utility.

The import command should be similar to the following:

```
> keytool -import -alias <your cert alias name> -file <cert file> -keystore <the truststore used by the JRE> -trustcacerts - storepass <store password>
```

Note: The certificate imported to the truststore should be in PEM encoding.

For example, for a server with SSL support called **www.mysslserver.com**, a JRE truststore called **c:\jre142\lib\security\cacerts**, and a CA issued certificate called **mysslserver** found in the file **c:\mycacert.pem**, the following is the correct format for the command to import the required certificate to the truststore:

```
> keytool -import -alias mycacert -file d:\mycacert -keystore c:\jre142\lib\security\cacerts -trustcacerts -storepass changeit
```

Note: The default password of the truststore is **changeit**.

Once the command has been run, the JRE is able to validate the certificate sent by the SSL Web server.

Setting JRE to Use Client/Server-Side Authentication

When the JRE is used as the server-side in an SSL communication channel, it can be required to send a client/server-side certificate. The JRE will use its keystore to look for the certificate and the corresponding private key. To support the sending of certificates by JRE, you must carry out the following steps:

- Import, or create, a keystore containing the certificates and private keys
- Define the keystore parameters in the JVM run-time properties

---

Note: The default keystore used by the JRE is a file called **.keystore** that is located in the user's home directory.

To import, or create, a keystore containing the certificates and private keys:

- The keystore can be either a JKS file or a PKCS#12 file.
- You can create a JKS file with a self-signed certificate using keytool.exe.

  An example of the keytool command for creating a JKS file is:

  ```
  /> keytool -genkey -dname “CN=your name, OU=organization
  unitO=organization” -validity <365> -keystore <new keystore> -alias <key alias>-
  keypass <key password> -storepass <store password>
  ```

  The parameters used are:

  - dname – distinguished name
  - validity – certificate validity

- keystore – the new store to be created, or to which to add the new key
- alias – the new certificate and key alias name in the keystore
- keypass – the password for using the private key
- storepass – the password for using the keystore

- You can generate a self-signed certificate using the keys generated by the previous command.

  An example of the keytool command for creating a JKS file is:

  ```
  /> keytool -selfcert -alias <key alias> -keystore <new keystore>-keypass <key password> -storepass <store password>
  ```

  The parameters used are:

  - keystore - the new store to be created, or to which to add the new key
  - alias - the new certificate and key alias name in the keystore
  - keypass - the password for using the private key
  - storepass - the password for using the keystore

  To define the keystore parameters in the JVM run-time properties:

  After you have created a keystore that contains the required certificates, you must configure JVM to use the keystore.

  To configure JVM to use the keystore, add the following parameters to your JVM instance:

- Djavax.net.ssl.keyStore=<keystore>
- Djavax.net.ssl.keyStorePassword=<keystore password as defined>
- Djavax.net.ssl.keyStoreType=PKCS12 or JKS

4

Using Basic Authentication in Mercury Business Availability Center

This chapter describes how to configure your Mercury Business Availability Center platform to support authentication using the basic authentication protocol.

This chapter describes:	On page:
Introducing Basic Authentication Deployment in Mercury Business Availability Center	64
Mercury Business Availability Center Components Supporting Basic Authentication	66
Configuring Basic Authentication Between the Centers Server and Application Users	68
Configuring Basic Authentication Between the Core Server and the Data Collectors	71
Auto Upgrading Data Collectors Remotely when Using Basic Authentication	77

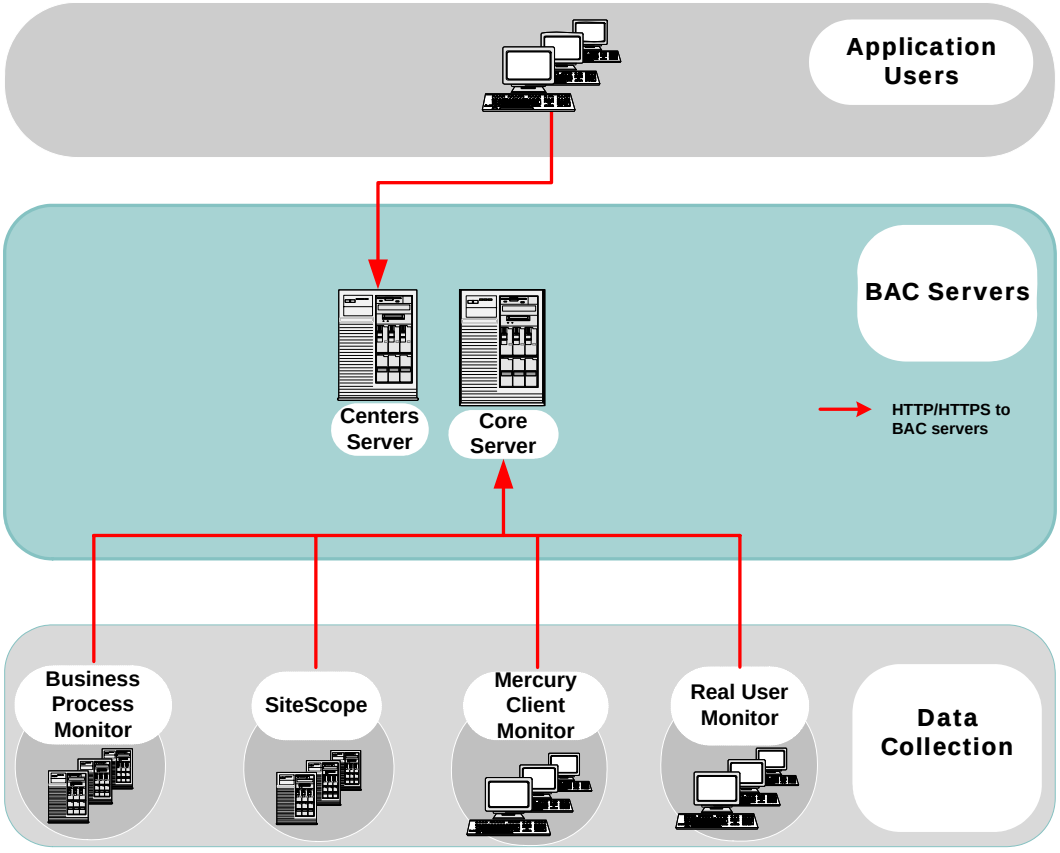
Introducing Basic Authentication Deployment in Mercury Business Availability Center

The Mercury Business Availability Center platform fully supports the basic authentication schema, which provides Mercury Business Availability Center with the ability to authenticate a client communicating with a Mercury Business Availability Center server via HTTP or HTTPS.

The basic authentication schema is based on the client sending its credentials to the server so that the server can authenticate the client. The client's credentials are sent in a base64 encoding format and are not encrypted in any way. If you are concerned about your network traffic being sniffed, it is recommended that you use basic authentication in conjunction with SSL. This sends the client's credentials over an encrypted wire (after the SSL handshake has been completed).

For information on configuring the Mercury Business Availability Center platform to support SSL communication, see Chapter 3, "Using SSL in Mercury Business Availability Center."

Possible basic authentication channels in Mercury Business Availability Center are illustrated in the following diagram:



Overview of Configuring Basic Authentication in Mercury Business Availability Center

Before proceeding with the configuration steps, ensure that:

- the Mercury Business Availability Center platform is operating as it is supposed to without basic authentication
- you read this chapter in its entirety before you begin performing the configuration
- you define your authentication requirements and use basic authentication only where required

---

Note: The configuration specified for each Mercury Business Availability Center server is also relevant for a single machine installation, in which the Centers Server, Core Server, and Data Processing Server all reside on the same machine.

Mercury Business Availability Center Components Supporting Basic Authentication

You set a Mercury Business Availability Center server to support basic authentication by enabling basic authentication support for the Web server installed on the Mercury Business Availability Center server, and for the Web Guard component on the Mercury Business Availability Center server.

You configure Mercury Business Availability Center clients to support basic authentication by defining the appropriate settings for each particular type of client, as described in the relevant client sections later in this chapter.

Web Servers Supporting Basic Authentication

The following table details the Web server–operating system combination that is required for basic authentication support.

	Microsoft IIS	Apache Web Server
Operating System	Windows 2000 Windows 2003	Windows 2000 Windows 2003

Note: Both the Centers Server and Core Server require Web servers to communicate with their clients.

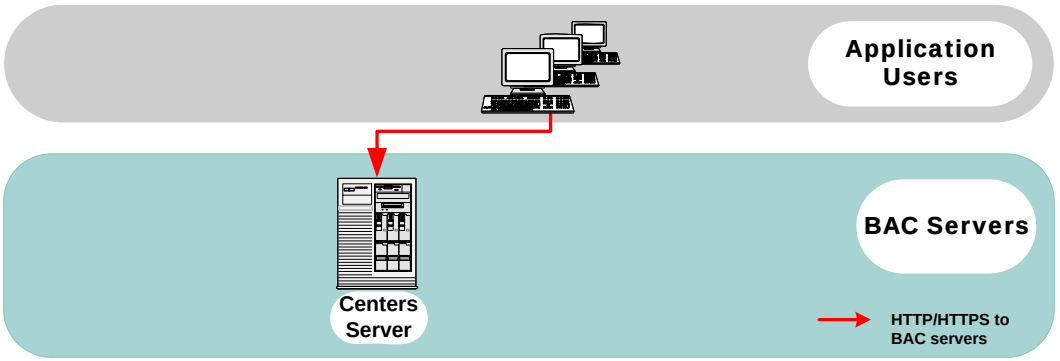
Mercury Business Availability Center Clients Supporting Basic Authentication

The following Mercury Business Availability Center clients support basic authentication communication with the Mercury Business Availability Center servers:

- **Browsers** – when used as Mercury Business Availability Center machine (when Mercury Business Availability Center is installed on a single machine) or Centers Server clients
- **Data collectors** – Business Process Monitor, Client Monitor, Real User Monitor, and SiteScope, when used as Mercury Business Availability Center machine (when Mercury Business Availability Center is installed on a single machine) or Core Server clients

Configuring Basic Authentication Between the Centers Server and Application Users

The instructions in this section describe how to configure the Centers Server (or a Mercury Business Availability Center machine, in the case of a single machine installation) and application users to support basic authentication.



This section describes:

- Basic Authentication Configuration for the Centers Server – see below
- Basic Authentication Configuration for the Application Users – see page 70

Basic Authentication Configuration for the Centers Server

This section provides instructions for configuring the Centers Server (or a Mercury Business Availability Center machine, in the case of a single machine installation) to support basic authentication.

It contains the following instructions:

- Enable Basic Authentication Support on the Web Server – see below
- Enable Basic Authentication Support for the Centers Server Web Guard – see page 69

Enable Basic Authentication Support on the Web Server

The first step in configuring the Centers Server to support basic authentication is to configure the Web server used by the Centers Server.

Note: On each Web server, make sure that you enable basic authentication only and disable anonymous access. Once you have enabled basic authentication, validate the settings by requesting a Mercury Business Availability Center resource and ensuring that you are prompted to insert basic authentication parameters.

- **Microsoft Internet Information Server (IIS) 5.0 and 6.0** – See http://www.microsoft.com for information on enabling basic authentication for all interaction with the Web server. Note that basic authentication should be enabled for the entire IIS Web Site under which you installed the Mercury Business Availability Center applications.
- **Apache HTTP Server 2.0.x** – See http://httpd.apache.org/docs-2.0/howto/auth.html for information on enabling basic authentication for all interaction with the Web server, using mod_auth. Note that basic authentication should be enabled on all the directories used by the Web server.

Once you have performed the above configuration procedures, when you are using a Microsoft IIS 5.0 or 6.0 Web server, you must make sure that all the folders and files in use by Mercury Business Availability Center have the required NTFS permissions required for the Users connecting to Mercury Business Availability Center.

Enable Basic Authentication Support for the Centers Server Web Guard

Web Guard is a component in each Mercury Business Availability Center server that tracks the validity of the Mercury Business Availability Center components using HTTP/HTTPS. If you configured your Web server to use basic authentication, the Web Guard must also be configured to use basic authentication.

To configure the Web Guard to use basic authentication:

1. Double-click the **<Centers Server root directory>/bin/setsiteauthentication.exe** utility.
2. Select the **Using basic authentication** check box.
3. Enter the following parameter values:
   - **User name –** the user name to be used to log in to the Centers Server
   - **Password –** the user password to be used to log in to the Centers Server
   - **Domain –** the domain name to be used to log in to the Centers Server
4. Restart the Mercury Business Availability Center service on the Centers Server.

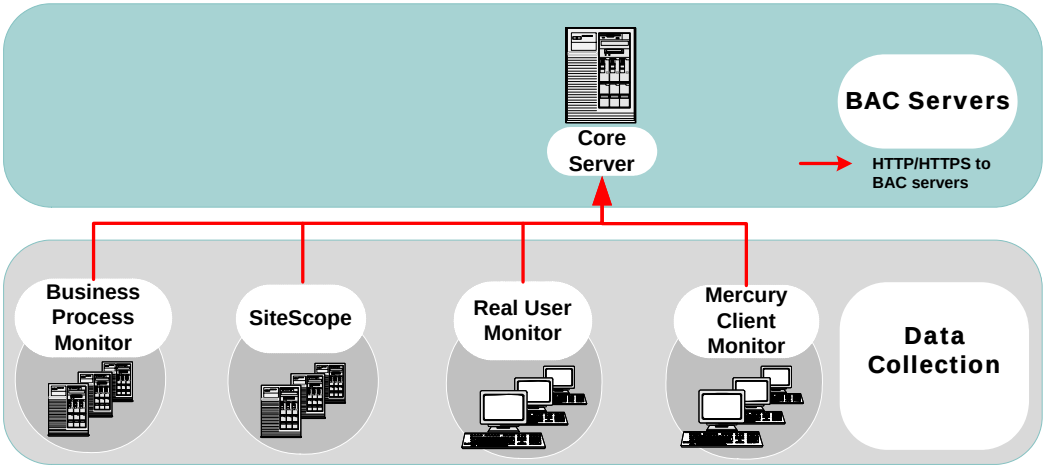
Basic Authentication Configuration for the Application Users

This section provides instructions for configuring the application users (Centers Server clients) to support basic authentication.

To connect as an application user to a Mercury Business Availability Center server that requires basic authentication, it is only necessary for you to know the credentials of the user permitted to log in to the Mercury Business Availability Center Web server. When connecting to the Web server, you will be prompted to enter these credentials. The authentication is then performed automatically.

Configuring Basic Authentication Between the Core Server and the Data Collectors

The instructions in this section describe how to configure the Core Server and the Mercury Business Availability Center data collectors to support basic authentication. To enable basic authentication support, you must make the required changes for the Core Server, as well as for all the Mercury Business Availability Center data collectors connecting to it using HTTP/S.



This section describes:

- Basic Authentication Configuration for the Core Server – see below
- Basic Authentication Configuration for the Data Collectors – see page 74

Basic Authentication Configuration for the Core Server

This section provides instructions for configuring the Core Server (or a Mercury Business Availability Center machine, in the case of a single machine installation) to support basic authentication.

It contains the following instructions:

- Enable Basic Authentication Support on the Web Server – see below
- Enable Basic Authentication Support for the Core Server Web Guard – see page 73

Enable Basic Authentication Support on the Web Server

The first step in configuring the Core Server to support basic authentication is to configure the Web server used by the Core Server.

Note: On each Web server, make sure that you enable basic authentication only and disable anonymous access. Once you have enabled basic authentication, validate the settings by requesting a Mercury Business Availability Center resource and ensuring that you are prompted to insert basic authentication parameters.

- **Microsoft Internet Information Server (IIS) 5.0 and 6.0** – See http://www.microsoft.com for information on enabling basic authentication for all interaction with the Web server. Note that basic authentication should be enabled for the entire IIS Web Site under which you installed the Mercury Business Availability Center applications.

➤ **Apache HTTP Server 2.0.x** – See http://httpd.apache.org/docs-2.0/howto/auth.html for information on enabling basic authentication for all interaction with the Web server, using mod_auth. Note that basic authentication should be enabled on all the directories used by the Web server.

Once you have performed the above configuration procedures, when you are using a Microsoft IIS 5.0 or 6.0 Web server, you must make sure that all of the folders and files in use by Mercury Business Availability Center has the required NTFS permissions required for the Users connecting to Mercury Business Availability Center.

After performing the above procedures, the Web server installed on the Core Server is configured to support basic authentication for HTTP/S communication.

Enable Basic Authentication Support for the Core Server Web Guard

Web Guard is a component in each Mercury Business Availability Center server that tracks the validity of the Mercury Business Availability Center components using HTTP/HTTPS. If you configured your Web server to use basic authentication, the Web Guard must also be configured to use basic authentication.

To configure the Web Guard to use basic authentication:

1. Double-click the **<Core Server root directory>/bin/setsiteauthentication.exe** utility.
2. Select the **Using basic authentication** check box.
3. Enter the following parameter values:
   - **User name –** the user name to be used to log in to the Core Server
   - **Password –** the user password to be used to log in to the Core Server
   - **Domain –** the domain name to be used to log in to the Core Server
4. Restart the Mercury Business Availability Center service on the Core Server.

Basic Authentication Configuration for the Data Collectors

This section provides instructions for configuring the following Mercury Business Availability Center data collectors to support basic authentication:

- Business Process Monitor – see below
- Client Monitor Agent – see page 47
- SiteScope – see page 49
- Real User Monitor – see page 49

Business Process Monitor

If you configured the Core Server to require basic authentication, you must configure the Business Process Monitor to connect to the Core Server using basic authentication.

To configure the Business Process Monitor to use basic authentication:

1. Open the Business Process Monitor Admin (**http://<Business Process Monitor machine>:2696**)
2. In the Business Process Monitor page, identify the Business Process Monitor instance you want to configure from the **Instances** list and click the **Edit** button for the instance. The Edit Instance page opens.



3. In the **Authentication** section, enter the following parameter values:
   - **Authentication user name** – the user name to be used to log in to the Core Server
   - **Authentication user password** – the user password to be used to log in to the Core Server
   - **Authentication domain** – the domain name to be used to log in to the Core Server
4. Click **Save Changes and Restart Instance**.

Client Monitor Agent

If you configured the Core Server to require basic authentication, you must configure the Client Monitor Agent to connect to the Core Server using basic authentication.

To configure the Client Monitor Agent to use basic authentication:

1 Open the Client Monitor Agent Settings dialog box (**Start** > **Programs** > **Mercury Client Monitor** > **Client Monitor Agent Settings**).

2 In the **Security** tab, select the **Use basic authentication** check box and enter the following parameter values:

- **User name** – the domain and user name to be used to log in to the Core Server (in the format domain\user name)
- **Password** – the password to be used to log in to the Core Server

3 Click **Save & Exit**.

4 Restart the Client Monitor Agent.

SiteScope

If you configured the Core Server to require basic authentication, you must configure the SiteScope machine to connect to the Core Server using basic authentication.

To configure the SiteScope machine to use basic authentication:

- If you are configuring SiteScope using Mercury Business Availability Center Monitor Administration, right-click the SiteScope you want to instruct to use basic authentication, and select **Edit**.

  In the **Profile Settings** section of the Edit SiteScope page, enter the following parameter values:

  - **Web server authentication user name** – the user name and domain of the Core Server (in the format domain\user name)
  - **Web server authentication password** – the password of the Core Server

  Click **OK** at the bottom of the page and restart the SiteScope instance.

➤ If you are configuring SiteScope using the SiteScope interface, select **Preferences** > **Mercury AM**.

In the **Optional Settings** section of the Mercury Business Availability Center Server Registration page, enter the following parameter values:

- **Authentication username** – the user name and domain of the Core Server (in the format domain\user name)
- **Authentication password** – the password of the Core Server

Click the **Update** button at the bottom of the page and restart the SiteScope instance.

Real User Monitor

If you configured the Core Server to require basic authentication, you must configure the Real User Monitor engine machine to connect to the Core Server using basic authentication.

To configure the Real User Monitor engine machine to use basic authentication:

1. Open the Real User Monitor Web Console (**http://<Real User Monitor engine name>:8180/rumconsole**).
2. Click the **Configuration** tab.
3. Under **Basic Authentication**, select the **Use basic authentication** check box and enter the following parameter values:
   - **Authentication user name –** the user name to be used to log in to the Core Server
   - **Authentication user password –** the user password to be used to log in to the Core Server
   - **Authentication domain –** the domain name to be used to log in to the Core Server
4. Click **Save Configuration**.

Auto Upgrading Data Collectors Remotely when Using Basic Authentication

You can perform a remote auto update for the Business Process Monitor, Client Monitor, and SiteScope data collectors by supplying parameters required to download the update from the Web server on which it is located. If the Web server from which you are downloading the update is using basic authentication, you must perform the following procedure in Mercury Business Availability Center in order to enable the remote auto upgrade.

To auto upgrade data collectors remotely when using basic authentication:

1. In Mercury Business Availability Center's Platform Administration, select **Data Collection** > **Data Collector Maintenance**. The Data Collector Maintenance page opens.
2. Click the **SiteScope**, **Business Process Monitor**, or **Client Monitor** tab, depending on the type of data collector you want to upgrade.
3. Select the check box for the data collector instance you want to upgrade.

   To make your selections, you can also use the buttons at the bottom of the page for, **Select All**, **Clear All**, and **Invert Selection.**
4. Click **Upgrade** at the bottom of the page. The Upgrade dialog box opens.
5. Select **Use Basic Authentication** and enter the following authentication parameter values:
   - **User Name –** the user name to be used to log in to the Core Server
   - **Password –** the user password to be used to log in to the Core Server
   - **Domain –** the domain name to be used to log in to the Core Server
6. Click **Start Upgrade**. The Actions Status window opens, and the upgrade process continues there. For details of the rest of the procedure, see "Tracking Actions Status" on page 121 in *Platform Administration*.

Index

A

application users
  configuring basic authentication support for 68
  configuring SSL support for 35, 47

B

basic authentication
  supported Mercury Business Availability Center components 66
  using with Mercury Business Availability Center 63
basic authentication support
  configuring for application users 68
  configuring for Centers Server 68
  configuring for Core Server 71
  configuring for data collectors 71
Business Process Monitor
  configuring basic authentication support for 74
  configuring SSL support for 38

C

Centers Server
  configuring basic authentication support for 68
  configuring SSL support for 35, 47
certificates, setting Java Runtime Environment 58
Client Monitor Agent
  configuring basic authentication support for 75
  configuring SSL support for 40
Core Server
  configuring basic authentication support for 71
  configuring SSL support for 37

D

data collectors
  configuring basic authentication support for 71
  configuring SSL support for 37
Discovery Agent, configuring SSL support for 45
distributed deployment, using a reverse proxy with 20

H

hardening, for Mercury Business Availability Center platform 1

J

Java Runtime Environment, working with client/server certificates 58

M

Mercury Business Availability Center
  components supported in basic authentication 66
  components supported in SSL 33
  deploying in a secure architecture 4
  reverse proxy modes 11
  supported SSL topologies 34
  using basic authentication with 63
  using SSL with 29

Mercury Business Availability Center
platform, hardening 1

R

Real User Monitor
configuring basic authentication support for 76
configuring SSL support for 43
remote upgrade, when using basic authentication 77
reverse proxy
mode support for Mercury Business Availability Center 11
using in Mercury Business Availability Center 7

S

secure architecture, for Mercury Business Availability Center 4
security, for Mercury Business Availability Center platform 1
single machine deployment, using a reverse proxy 13
SiteScope
configuring basic authentication support for 75
configuring SSL support for 42
SSL
configuring the Web Guard 57
supported Mercury Business Availability Center components 33
supported topologies in Mercury Business Availability Center 34
using with Mercury Business Availability Center 29
SSL support
configuring for application users 35, 47
configuring for Centers Server 35, 47
configuring for Core Server 37
configuring for data collectors 37

W

Web Guard, configuring for SSL 57