QUEST SOFTWARE®

vRanger®

Version 5.4.1

User Guide

© 2012 Vizioncore, Inc.
ALL RIGHTS RESERVED.

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Quest Software, Inc.

The information in this document is provided in connection with Quest products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Quest products. EXCEPT AS SET FORTH IN QUEST'S TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, QUEST ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL QUEST BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF QUEST HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Quest makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Quest does not make any commitment to update the information contained in this document.

If you have any questions regarding your potential use of this material, contact:

Quest Software World Headquarters
LEGAL Dept
5 Polaris Way
Aliso Viejo, CA 92656
www.quest.com
email: legal@quest.com

Refer to our Web site (*www.quest.com*) for regional and international office information.

Patents

Protected by U.S. Patents 8,046,550; 8,060,476; and 8,135,930.Additional patents pending.

Trademarks

Quest, Quest Software, the Quest Software logo, AccessManager, ActiveRoles, Aelita, Akonix, Benchmark Factory, Big Brother, BridgeAccess, BridgeAutoEscalate, BridgeSearch, BridgeTrak, BusinessInsight, ChangeAuditor, CI Discovery, Defender, DeployDirector, Desktop Authority, Directory Analyzer, Directory Troubleshooter, DS Analyzer, DS Expert, Foglight, GPOADmin, Help Desk Authority, Imceda, IntelliProfile, InTrust, Invirtus, iToken, JClass, JProbe, LeccoTech, LiteSpeed, LiveReorg, LogADmin, MessageStats, Monosphere, NBSpool, NetBase, NetControl, Npulse, NetPro, PassGo, PerformaSure, Point, Click, Done!, Quest vToolkit, Quest vWorkSpace, ReportADmin, RestoreADmin, ScriptLogic, SelfServiceADmin, SharePlex, Sitraka, SmartAlarm, Spotlight, SQL Navigator, SQL Watch, SQLab, Stat, StealthCollect, Storage Horizon, Tag and Follow, Toad, T.O.A.D., Toad World, vAutomator, vConverter, vEcoShell, VESI, vFoglight, vPackager, vRanger, vSpotlight, vStream, vToad, Vintela, Virtual DBA, VizionCore, Vizioncore vAutomation Suite, Vizioncore vEssentials, Vizioncore vWorkflow, WebDefender, Webthority, Xaffire, and XRT are trademarks of Quest Software, Inc. and its subsidiaries. For a complete list of Quest Software trademarks, see *www.quest.com/legal/trademarks.aspx*. All other trademarks are property of their respective owners..

User Guide
Version 5.4.1 - Private Release
May 2012

Table of Contents

Introduction to this Guide **8**
About vRanger 9
vRanger Documentation Suite 9
vRanger Licensing 9
About Quest Software, Inc. 10
Contacting Quest Software 11
Contacting Quest Support 11

vRanger Overview **12**
Virtualization and VMware Overview 13
vRanger Overview 13
VMware vSphere 5 Support 13
My Jobs View 13
Startup Wizard 13
Resource Management 14
Flexibility in Backup Options 14
Restore 14
Restore a Repository 15
Email Notification and Reports 15
Direct-to-Target Architecture 15
Skipping VMs and Virtual Hard Disks 15
VMware VSS Support 15
PowerShell Access and vAPI 15
Encryption 15
Replication 16
Reports 16
Integration with vMotion 17

Configuration **18**
Configuration 19
Configuration through the Startup Wizard 19
Launching the Startup Wizard 19
Select a License 20
VirtualCenter and Host Information 21
Host Licensing Information 23

Virtual Appliance Information 24
Add Repositories 25
Set up Mail Server Information/Add Email Addresses 27
Configuration without Using the Startup Wizard 28
Add a VirtualCenter 28
Add an ESX Server 28
Add a Repository 29
Configure an SMTP Server 29
Import License File 31
Manage Host Licensing 32
Manage Job Configuration Settings 33
Cataloging Options 34
Configuring Linux File Level Restoration 36
Configuration Options 37
Supplemental Instructions - Additional Repository Types 38
Adding an NFS Repository 39
Adding an FTP Repository 40
Adding an SFTP Repository 41
Adding an NVSD Repository 42
Adding an EMC Data Domain Boost (DDB) Repository 43
Configuring The vRanger Virtual Appliance 46
Virtual Appliance Overview 46
The Virtual Appliance Scratch Disk 46
Deploying the Virtual Appliance 49
Deploying Multiple Virtual Appliances 49
Configuring the Virtual Appliance 50
Deploying Multiple VAs From a Template - VI Client 54
Deploying Multiple VAs From a Template - PowerCLI 55
Disabling DRS for the vRanger Appliance 58
Adding the Virtual Appliance to vRanger 58
Changing Virtual Appliance Configuration 60
Increasing The VA Scratch Disk Size 62
Disabling DRS for the vRanger Appliance 63

Using vRanger 64

Interface Overview 65
My Inventory View 67
My Inventory Pane 68
Working Inventory Pane 74
My Jobs View 75
My Jobs Pane 75

Activity Pane 77
My Repositories View 80
My Repositories Pane 80
Working Repository Pane 81
My Reports View 83
My Reports Toolbar 83
Report Wizard 85

Backup 86
Backup Overview 87
Backup Limitations 87
Backup Transport Options 87
HotAdd 87
SAN Backups 88
LAN 89
Transport Selection Order 89
Configuring A Backup Job 91
Step 1: Optional Configurations 91
Step 1: Selecting An Inventory Node 96
Step 2: Exclusions and Inclusions 99
Step 3: Repository Selection 101
Step 4: Transport Selection 102
Step 5: Configure Backup Options 104
Step 6: Retention Policies & Space Saving Technologies 106
Step 8: Email Notification Selection 114
Step 9: Summary 115

Restore 117
The vRanger Restore Process 118
Performing a Full Restore 118
Performing a File-Level Restore - Windows 130
FLR- Limitations and Requirements 130
File Level Restore - Catalog Search 130
File Level Restore - Manual 132
Performing a File-Level Recovery - Linux 132
Restoring from a Manifest 133
Full Restore from Manifest 133
FLR from Manifest 142

Replicate 144
About Replication 145
Replication Requirements 145

How Replication Works ... 145
Replication With the Virtual Appliance ... 147
Replication Modes ... 148
Choosing a Replication Method ... 150
Configuring Transport Failover Options ... 151
Enabling/Disabling The Custom Transport Option ... 152
Pre-seeding Replication Jobs ... 152
Creating a Replication Job ... 153
Failover ... 162
Test Failover ... 162
Failover ... 164
Failback ... 165

Reports ... 167
Reports ... 168
My Reports View ... 168
My Reports Pane ... 168
Report Wizard ... 170
Create a Summary Report ... 176

Using the vRanger Console ... 177
vRanger Console Overview ... 178
vRanger Console - One-time Configuration ... 178
vRanger vAPI Cmdlets ... 179
Cmdlet Help ... 182

Integrating vRanger ... 183
Integrating vRanger ... 184
vRanger and Tape Backup Vendors ... 184
Deduplication with NetVault SmartDisk and vRanger ... 190
Data Domain and vRanger Repositories ... 191
Monitoring vRanger ... 191
Configuring Event Monitoring ... 192

Appendix A: vAPI Cmdlet Details ... 195

Appendix B: Using the VI Editor ... 300
VI Editor Overview ... 300
Starting VI ... 300
Changing Modes ... 300
Editing Commands ... 301
Moving the Cursor ... 301

Inserting Text 302
Deleting Text 302
Saving Your Changes 302

Index 304

Introduction to this Guide

This chapter provides information about what is contained in this document and the vRanger® documentation suite. In addition, it describes licensing.

This chapter contains the following sections:

About vRanger 9
vRanger Documentation Suite 9
vRanger Licensing 9
About Quest Software, Inc. 10

About vRanger

vRanger is the market-leading backup, recovery and replication solution for virtual environments. Featuring a new platform technology, vRanger reduces the backup window, provides smarter backup options, and offers more scalability features while using fewer resources.

vRanger capabilities:

- Performs incremental, differential or full image backups of virtual machines.
- Quickly restores the entire virtual machine or just specific files.
- Manages disaster recovery strategies and protects critical data in virtual environments
- Replicates virtual machines to more than one destination, allowing users to accomplish multiple HA and DR objectives based on their specific needs.

vRanger Documentation Suite

The documentation suite is made up of the core set plus documents intended for advanced configurations. Documentation is provided in a combination of online help and PDF.

- **Online Help:** You can access online help through Help in the Menu bar on the four main screens of the interface.
- **PDF:** The *Getting Started Guide*, *What's New Guide*, *System Requirements Guide, Installation and Setup Guide,* and *User Guide*, are provided as PDF files. The PDF guides are included in the zip file downloaded from Quest. Adobe® Reader® is required.

vRanger Licensing

There are three levels of vRanger Licensing, each with different available features:

- vRanger Standard
- vRanger Pro
- vReplicator

	vRanger Standard*	vRanger Pro	vReplicator**
Backup	X	X	
Restore	X	X	
VA-Based Backup/Restore		X	
Replication		X	X
CBT	X	X	X

	vRanger Standard*	vRanger Pro	vReplicator**
ABM	X	X	X
Lan Free Backup and Restore		X	
Catalog		X	
Windows FLR	X	X	
Linux FLR		X	

A license for vRanger controls the number of source CPUs that you can configure for backup. For licensing purposes, a dual-core processor is counted as a single CPU. Therefore, one ESX Server with two dual-core processors would use two CPUs of a license.

If you do not have a permanent license, an evaluation license will be generated for you automatically. Every time you use vRanger to perform a backup using an evaluation license, your Internet browser will open to the Quest website. After you import a permanent license, this will no longer happen. You can import a license through the Startup Wizard or the Configuration Options dialog.

Upgrades from previous versions of vRanger will yield vRanger Pro installations, with the functionality described on the previous page.

Upgrades from previous versions of vReplicator will result in vReplicator only functionality, as described on the previous page.

Note Please note that only one version of vRanger can be installed on a machine at any one time. For example, you cannot install vRanger SE and vReplicator on the same machine.

About Quest Software, Inc.

Quest Software simplifies and reduces the cost of managing IT for more than 100,000 customers worldwide. Our innovative solutions make solving the toughest IT management problems easier, enabling customers to save time and money across physical, virtual and cloud environments. For more information about Quest go to *www.quest.com.*

Contacting Quest Software

Email	*info@quest.com*
Mail	Quest Software, Inc. World Headquarters 5 Polaris Way Aliso Viejo, CA 92656 USA
Web site	*www.quest.com*

Refer to our web site for regional and international office information.

Contacting Quest Support

Quest Support is available to customers who have a trial version of a Quest product or who have purchased a Quest product and have a valid maintenance contract. Quest Support provides unlimited 24x7 access to our support portal. Visit our support portal at: *www.quest.com/support.*

From our support portal, you can do the following:

- Retrieve thousands of solutions from our Knowledge Base
- Download the latest releases and service packs
- Create, update and review Support cases

View the Global Support Guide for a detailed explanation of support programs, online services, contact information, policies and procedures. The guide is available at: *www.quest.com/support.*

vRanger Overview

This chapter provides an overview of vRanger features.

Virtualization and VMware Overview 13
vRanger Overview 12

Virtualization and VMware Overview

As companies grow, their technology needs change. These changes are often implemented within complex systems running business-critical applications. Usually there is an increased demand for hardware and software resources. To manage this demand, many companies establish virtual environments. Doing so can increase an organization's agility and efficiency while lowering its costs.

Most companies have a number of specialized physical servers and workstations that are underutilized. Virtualizing such an environment increases and balances utilization by consolidating the physical machines into a single physical host that runs multiple VMs.

vRanger Overview

vRanger offers comprehensive data protection and management that supports vSphere. Delivering fast and reliable backup and recovery, vRanger can capture the entire image of a VM - allowing you to recover the complete machine or an individual file. All jobs can be executed without interrupting service--that is, while the source machine is running. Multiple ESX(i) hosts can be used for simultaneous jobs, reducing the backup window. To save time and disk storage space after a full backup, you can run differential or incremental backup jobs to capture only the changed data. You can run backup and restore jobs immediately or schedule them. You can track the progress of all jobs and their component tasks. vRanger allows you to connect to multiple vCenters and back up multiple ESX(i) hosts.

VMware vSphere 5 Support

This version of vRanger offers full support for the vSphere 5 platform, bringing additional functionality, platform security, and no scripting requirements.

My Jobs View

vRanger's **My Jobs** view offers a central location for creating, managing, executing, and scheduling backup jobs. Here you can view detailed information, including all statuses for current and recent jobs and tasks. You can start, stop, disable, edit, and cancel jobs. You can monitor job and task progress. **My Jobs** features a search tool that includes multiple job and task criteria, allowing you to search by type, status, date, and so forth. You can also save your search criteria in a template, which you can access at any time to run future searches more efficiently.

Startup Wizard

The **Startup Wizard** is the tool that you will use to configure vRanger for your environment. Redesigned for ease of use, it moves you quickly through adding a license, VC, host, repository, SMTP server, and email addresses. Using the **Startup Wizard** or the **Configuration Options** dialog, you can manage your settings and update them at any time.

Resource Management

Through the **Configuration Options** dialog, you can manage a core vRanger resource--tasks. Set a limit for the maximum number of tasks running on a host and repository. Limit the number of tasks that can be run off of a LUN. Limit the time a task can take to complete. You can manage another important resource through this dialog--space on the host. It is vital to set an accurate minimum space requirement on each datastore to allow for snapshot growth during backup.

Flexibility in Backup Options

vRanger offers a wide variety of options that allow you to configure your backup and restore jobs to meet your business needs.

- To save both time and storage space, you can capture only changes made since the last backup by executing an incremental. A differential will offer you the changes made to the VMDK since the last full job.
- With Changed Block Tracking (CBT) enabled, vRanger only reads the portions of a disk that have changed since the last backup. This often results in shorter backup operations, and reduced resource consumption on network and storage elements.

  Note vSphere supports CBT, and most VMs running in this environment can utilize it. The VMs must be Hardware Version 7, and have been created and hosted in ESX4 or 4i hosts. VMs that are created in ESX3.5 or below must be migrated to Hardware Version 7 for CBT to be supported.
- Active Block Mapping (ABM) filters deleted data blocks so that only active blocks are scanned and streamed to the repository. White space detection eliminates the need to compress, stream, and write zero blocks during the replication process.
- You can execute backups to CIFS, NFS (version 3), FTP, or SFTP share. In addition, s you may also deduplicate your backups by using a NetVault SmartDisk (v1.5 or later) instance or Data Domain system with DD Boost as a repository.
- Cataloging allows searching and browsing of savepoints for files inside the VM images. Cataloging must be enabled at the start of the backup job to function.

Restore

Use vRanger to restore a full VM image or an individual file. When a full VM is restored, all data, applications, configuration settings, OS, and patches are recovered. You can configure each savepoint to a destination. You can rename the VM. You can even configure the restore to skip disks. When you selectively restore disks from a VM, you do so without interrupting service. That is, the remaining disks are not impacted. Automatic data validation confirms that the contents of the backup match the contents of the restore. FLR allows you to restore individual files quickly.

Note Backup and Restore can only be run over a LAN link -- using a WAN link will result in a failed job.

Restore a Repository

Backups that are stored in CIFS, NFS, FTP, and SFTP repositories are in human readable form. You can access an entire repository's tree structure, including all archives.

Email Notification and Reports

Easily configure and manage SMTP servers and email addresses for notifications and reports. Alerts can provide easy access to backup and restore job statuses. Email access to reporting functionality offers a more detailed look at the jobs that were run.

Direct-to-Target Architecture

vRanger makes it possible to back up even a large number of VMs directly from a source to a target repository. Traffic can be spread across multiple ESX hosts, increasing storage capacity horizontally. This approach decreases your overall backup window and allows you to manage jobs between multiple datacenters.

Skipping VMs and Virtual Hard Disks

You can configure backup jobs to designate which VMs or Virtual Machine Disks (VMDKs) should be skipped during a backup job. You can complete this configuration using the Advanced mode in the Backup Wizard or by adding the backup through My Jobs.

VMware VSS Support

vRanger makes it possible to use Microsoft Volume Shadow Copy Service (VSS) through the Enable Guest Quiescing option in the Backup Wizard.

PowerShell Access and vAPI

vRanger can connect to Windows PowerShell, a command-line shell and scripting language. You can increase efficiency and flexibility through automated scripting and improved infrastructure management. The PowerShell cmdlets are the clients of the vAPI.

PowerShell is at the core of the vRanger Console--a CLI--which you can use to manage the software, VMware, and Windows. vAPI is the web service through which software components can integrate with vRanger. vAPI cmdlets are functions that you can use to configure the software through the vRanger Console.

Encryption

vRanger uses the AES-256 encryption to encrypt backup archives and replication traffic. AES is a symmetric key algorithm that uses (in vRanger's case) a 256 bit encryption key. This combination is approved for the encryption of "Top Secret" government data, and complies with the Federal Information Processing Standard (FIPS).

Encryption for Backups

Encryption for vRanger backups is configured at the repository level, meaning that rather than selecting encryption for each job or task, you simply configure that job to write to an encrypted repository. For each backup task, vRanger will read the encryption status of the target repository. If encryption is selected, the data blocks are read and encrypted before they are sent to the repository.

Encryption for Replication

During replication tasks, vRanger sends data over an SSH tunnel encrypted with AES-256 encryption.

Encryption for vRanger Data

vRanger requires credentials for several connection types; namely local machine credentials, vCenter and host credentials, and credentials for the vRanger database. These credentials are encrypted using AES-256 encryption before they are written to the vRanger database or Windows registry.

Replication

vRanger's replication functionality was designed specifically to take advantage of VMware Infrastructure's enhanced Disaster Recovery capabilities. With a simple Windows-based interface, two replication options, and the ability to replicate to dissimilar storage and server hardware, vRanger is a powerful image-level replication solution that is also easy to use.

In addition, vRanger offers key features that set it far ahead of other replication products for virtualized environments, including:

- A VSS driver that utilizes Microsoft's Volume Shadow Copy Service to pause application writes. This feature will enable quiescing of supported databases to provide a transactionally consistent image at the secondary site.
- Full image, differential or snapshot-synched replication for faster, more efficient replication jobs.
- Skip VMDK feature allows bypassing of non-essential disks for faster jobs.
- Replication to dissimilar hardware with closed target VMs to reduce costs of destination host
- Replicate over LAN/WAN with no dependency on SAN

Reports

The **My Reports** view is a gateway to creating, editing, and distributing custom reports. There are four report types for backup and restore tasks and jobs. Using the Report Wizard, you can give each report type a different look--from headers and footers to content. The report can include custom metrics--which jobs were run, which succeeded, and which failed. You can also schedule a report to run, save it as a PDF, and send it as an email attachment.

Backup report information is organized by VM and includes when backups were run, who ran them, and the status of each backup. This can be done for one or multiple VMs.

Integration with vMotion

vMotion is enabled in vRanger, so you can move powered on VMs between ESX hosts while backup jobs are running.

Configuration

This chapter describes configuration options for vRanger. For installation options and instructions, see the *Installation and Setup Guide.*

This chapter contains the following sections:

Configuration 18
Configuration through the Startup Wizard 19
Configuration without Using the Startup Wizard 28
Configuration Options 37
Supplemental Instructions - Additional Repository Types 38
Configuring The vRanger Virtual Appliance 46

Configuration

Note Much of information in this chapter is identical to that found in the *Installation and Setup Guide*. It is included here as a convenience for those instances when you may wish to change your configuration post-installation.

Before you begin configuring vRanger, confirm that you have access to all user names, passwords, and root passwords for the ESX(i) hosts that you intend to use.

vRanger requires some basic configurations to ensure that the software functions properly. You complete the initial configuration through the Startup Wizard, which launches when you first install vRanger. If you would rather configure settings through the menu commands and icons on the main vRanger screen, you can do so. In either case, you can revise the settings at any time.

After completing the basic configuration process, you can use either the **Startup Wizard** or the **Configuration Options** dialog to manage your settings. Although the **Configuration Options** dialog offers some functionality that the **Startup Wizard** does you can use either dialog to manage host licensing or email server data. Only the **Configuration Options** dialog lets you manage resources on the task level. Given the role that tasks play in making backup and restore jobs, this resource is vital.

- Configuration through the Startup Wizard
- Configuration without Using the Startup Wizard
- Configuration Options

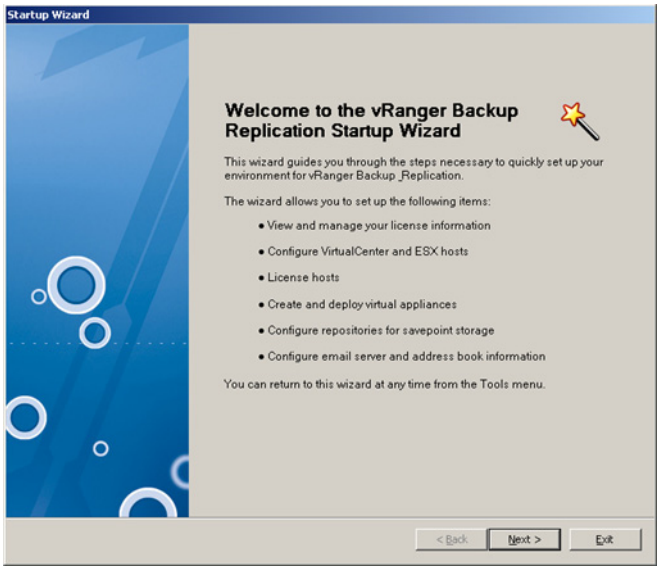
Configuration through the Startup Wizard

You must have an email SMTP server set up to enable notification during job creation. Later in this process, on the Notification Policy Selection dialog, you will have the option of choosing recipients. A plain text email will be sent when the job completes. The notification will include the backup results, if you select that option.

- Select a License
- VirtualCenter and Host Information
- Host Licensing Information
- Virtual Appliance Information
- Add Repositories
- Set up Mail Server Information/Add Email Addresses

Launching the Startup Wizard

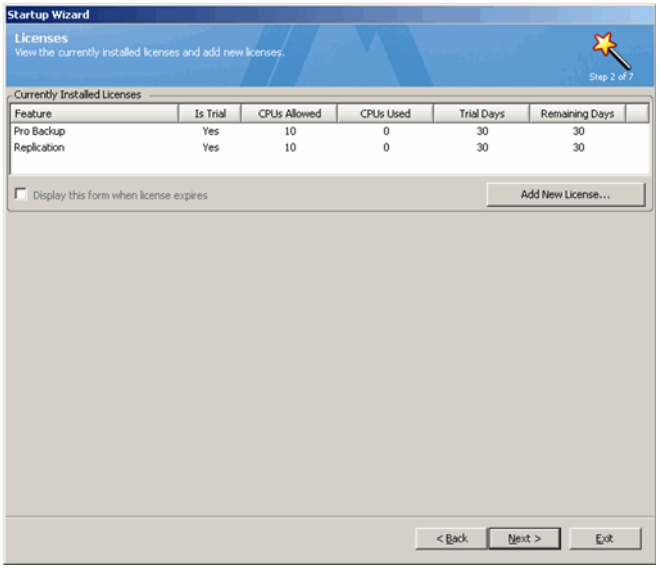
1 If you have just installed the software, click **Next** on the Startup Wizard. If not, access it through **Tools-->Startup Wizard.**



2 Click **Next.**

Select a License

The **Licenses** dialog displays.



1 Click **Add New License.**

Note The license file for vRanger versions 5.2 and later is different from those used in version 5.1 and 5.0. Do not use a license file for previous versions of this software.

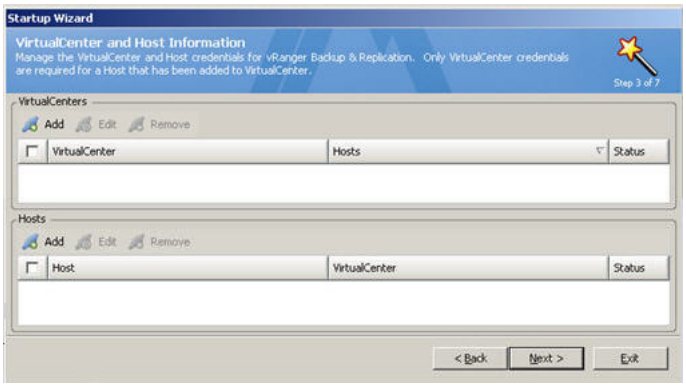
2 Select the license.

3 Click **Open.**

4 Click **Save.** Then, click **Next.**

VirtualCenter and Host Information

The **Virtual Center and Host Information** dialog displays. Before vRanger can add objects to the **My Inventory** pane, you must configure a connection to a vCenter and/or an ESX(i) Server.



Step 1: Adding a vCenter

1 In the VirtualCenters section, click Add.

2 In the **DNS Name** or **IP** field, enter the FQDN or IP address of the vCenter server.

3 In the **User Name** field, enter the user name of an account with privileges on the vCenter server. See the section “Configuring vCenter Permissions” in the *vRanger Installation and Setup Guide* for the required permissions for a vRanger vCenter account.

Caution The user name for the vCenter credential should be entered in the “username@domain” format, rather than “domain\username”. Please avoid special characters in the user name.

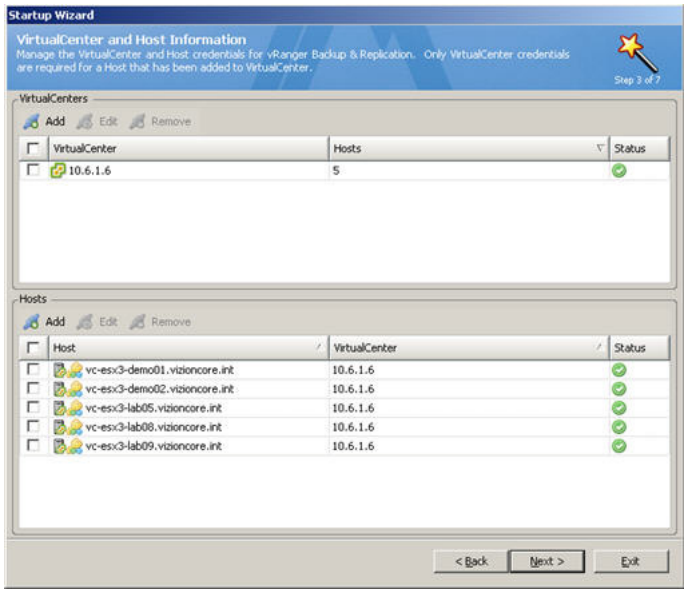
4 In the **User Password** field, enter the password for the account used above.

Note Please avoid special characters in the password.

5 In the **User Password** field, enter the password for the account used above.

6 Click **Save**.

The vCenter will display in the VirtualCenters section and on the VirtualCenter Information dialog. The Hosts managed by that vCenter will display in the Hosts section.

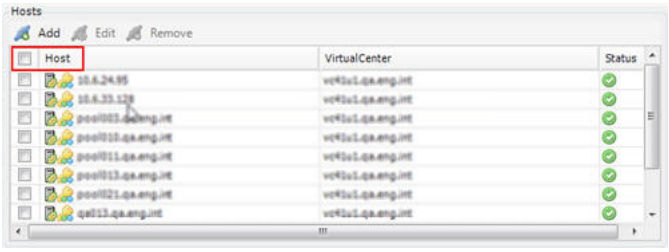


Note that the hosts displayed show the icon for VC credentials , showing the vCenter icon and a large key.

Step 2: Host Credentials

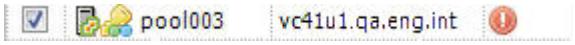
vCenter credentials are sufficient for operations that use only the vStorage API. You will need to credential each host in order for Direct-to-Target backups and replications (operations that utilize the Service Console) to work properly.

1. In the Hosts section, select one or more hosts. To select all hosts, select the upper-most checkbox



2. Click **Edit.**
3. The Host Connections dialog appears. If you have selected multiple hosts, the DNS Name or IP section will display “[multiple connections]”
4. Enter the user name and password for the account to be used.
5. Enter the root password for the selected hosts. If you hosts use different root passwords, you will need to configure them in like groups based on the root password.
6. Click **Connect**. The icon next to the host will change to , showing that the vCenter icon has been replaced with a key.

If there is an authentication problem, such as connections via the root account being disabled, the status icon will display an alarm:



If this happens, you will need to edit the host connection to use a non-root user account. Make sure that the Add user if user does not exist option is selected.

Note ESXi requires the use of the root account. You cannot use a non-root account with ESXi.



7 When all connections have been made and verified, click **Next.**

Step 3: Adding Hosts [Optional]

If you have hosts that are not managed by a vCenter, you can add them individually.

1 In the Host section, click **Add.**

2 In the **DNS Name** or **IP** field, enter the FQDN or IP address of the Host.

3 In the **User Name** field, enter an account for the host. You may use the root account if connections via root are enabled, or a non-root account. If the account does not yet exist on the host, select **Add user if user does not exist**.

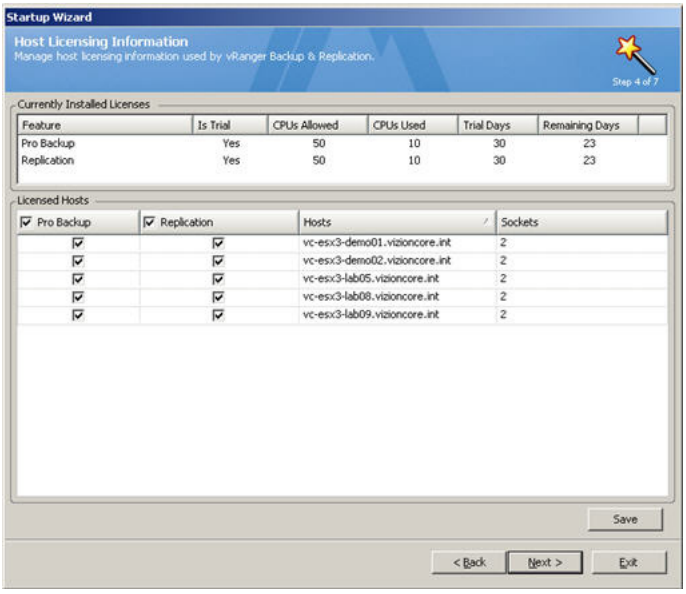
4 In the **User Password** field, enter the password for the account used above.

5 In the **Root Password** field, enter the password for the root account.

6 Click **Connect**. The Host will be display in the Hosts section.

Host Licensing Information

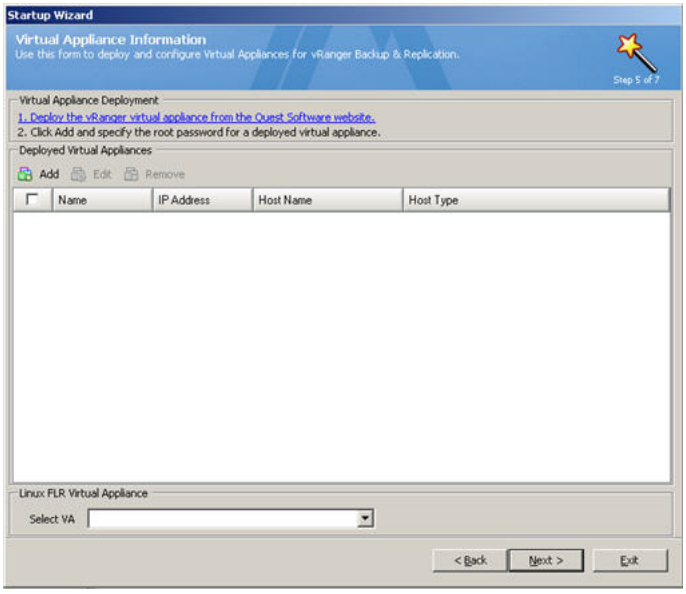
The **Host Licensing Information** dialog displays. It indicates the total number of licenses purchased, the number being used currently, and the number available.



1. View the **CPUs Allowed** value to confirm the number of sockets that are licensed for protection. The number of licenses currently allocated is shown in the **CPUs Used** column.
2. The Licensed Hosts section lists the managed hosts, the number of sockets for each host, and the licensed features for that host.
   a. The **Sockets** column indicates the license "cost" of each ESX Host.
   b. In the **Pro Backup and**/or **Replication** column, select or deselect ESX Hosts until you have used all of the available licenses, or until all of your ESX Hosts are protected.
3. Click **Save.**
4. Click **Next.**

Virtual Appliance Information

vRanger uses a virtual appliance for both Linux file-level recovery and for replication to and from VMware ESXi servers.



To simplify the startup procedure, the instructions for deploying and configuring the virtual appliance have been moved to:

Configuring The vRanger Virtual Appliance

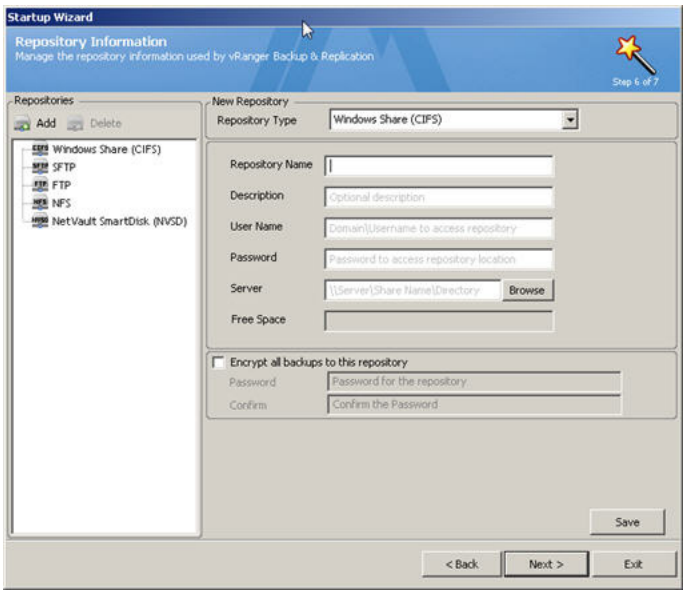
You may skip the virtual appliance configuration step now and continue with the Startup Wizard. You may access the Virtual Appliance Configuration dialog at any time via the **Tools> Options** menu.

Add Repositories

vRanger uses repositories to store backup archives. Repositories can be either a CIFS, NFS (version 3), FTP, or SFTP share. In addition, s you may also deduplicate your backups by using a NetVault SmartDisk (v1.5 or later) instance or Data Domain system with DD Boost as a repository.

Note For more information on **NetVault SmartDisk**, see the SmartDisk page of the Quest Software website: *http://www.quest.com/netvault-smartdisk/*. For more information on Data Domain's **DD Boost** technology, see the Data Domain website: *http://www.datadomain.com/products/dd-boost.htAdminiml*

The procedure below shows mounting a CIFS share to the My Repositories pane. The procedures for the other repository types are described in Supplemental Instructions - Additional Repository Types.



1 If the repository you want to add is listed on the left of the screen, click it to populate the fields on the right.Click **Update** and **Next.** If it is not listed, follow the steps below.

2 In the **Repository Type** section, select **Windows Share (CIFS),** if necessary**.**

3 Populate the **Repository Name** field.

This value will display in the **My Repositories** pane.

4 Populate the **Description** field.

5 Enter a username and password in the appropriate fields.

6 In the **Server** field, type the name of the server you want, or browse for the desired server.

Note If you wish to use the **Encrypt all backups** feature**, make certain to retain** the password you enter in the following steps. There is no back door or admin-level password. If the password is unknown, the backups will not be usable.

7 Select **Encrypt all backups to this repository** if you wish these backups to be password-protected.

Note Encryption is not supported for NetVault SmartDisk and Data Domain Boost repositories.

8 Enter a Password for the encrypted repository -- confirm the password by re-entering it.

9 Click **Save** -- the connection to the repository is tested and the repository is added to the My Repositories pane and the Repository Information dialog.

10 Click **Next**.

Set up Mail Server Information/Add Email Addresses

The Mail Server Information dialog displays. When you configure an SMTP server, select one that does not require authentication so that there are no obstacle to email notification transmission.

Step 1: SMTP Server

1. In the **SMTP Server** field, enter the DNS name or IP of an SMTP server.
2. In the **From Address** field, enter the email address that vRanger should use.

   Note Make sure that you use a valid email account registered to the SMTP server that you added.
3. Click **Send Test Email.**
4. If desired, select **Do not send email notifications for successful jobs**.

Step 2: Address Book

1. To add recipients, click **Add.**
2. Populate the **Name** and **Email Address** fields.
3. Click **Save** to add the information to the Email Addresses list.
4. Repeat steps 1-3 to include additional contacts.
5. Click **Finish.**

Configuration without Using the Startup Wizard

These instructions describe the process for configuring vRanger without using the Startup Wizard. During this process you will:

- Add a VirtualCenter
- Add an ESX Server
- Add a Repository
- Configure an SMTP Server
- Add Email Address
- Import License File
- Manage Host Licensing
- Manage Job Configuration Settings
- Configuring Linux File Level Restoration

Note To configure multiple hosts under the same vCenter, the system retains the data you entered to add credentials for the first host.

Add a VirtualCenter

1. To add a VirtualCenter, navigate to the **My Inventory** view. From the **Add icon** , select **VirtualCenter.**
2. On the **VirtualCenter Credentials** dialog, populate the **DNS Name or IP, User Name,** and **User Password** fields. Leave the **Port Number default** (443) as is. Click **Connect.**

   Note The user name for the vCenter credential should be entered in the “username@domain” format, rather than “domain\username”. Please avoid special characters in the user name.
3. vRanger attempts to connect to the vCenter using the current configuration. If it is successful, the vCenter displays in the **My Inventory** pane.

Add an ESX Server

1. From the **My Inventory** view, select the **Add Icon-->ESX Host.**

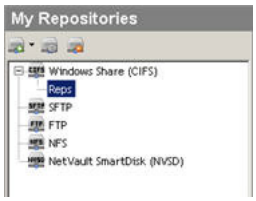
   Note If the connection is unsuccessful, confirm that the configuration settings are correct and that the vCenter is visible to the computer where vRanger is installed.
2. On the **Add Host Credentials** dialog, populate the **DNS Name or IP, User Name, User Password,** and **Root Password** fields. Leave the **Port Number** default (22). Click **Connect.** The host is added to the tree structure under the vCenter. Confirm this noting that a gold key icon displays adjacent to it in the **My Inventory** pane.

Add a Repository

1 To add repository, navigate to the My Repositories pane. Click to highlight the desired repository type. Right-click the node and select **Add-->Windows Share CIFS**.

Note These instructions describe the process for adding a CIFS repository. The procedures for the other repository types are described in Supplemental Instructions - Additional Repository Types.

2 On the **Add Repository** dialog, populate the **Repository Name, DNS Name or IP, Target Directory, User Name,** and **User Password** fields. The Description field is optional**.**



3 In the **Server** field, type the name of the server you want.

Note If you wish to use the **Encrypt all backups** feature**, make certain to retain** the password you enter in the following steps. There is no back door or admin-level password. If the password is unknown, the backups will not be usable.

4 Select **Encrypt all backups to this repository** if you wish these backups to be password-protected.

5 Enter a Password for the encrypted repository -- confirm the password by re-entering it.

6 Click **Update** -- the connection to the repository is tested and the repository is added to the My Repositories pane and the Repository Information dialog.

7 Click **OK**.

Configure an SMTP Server

1 Select **Tools-->Options.**

2 Under the General node on the Configuration Options dialog, click **Email Configuration.**

3 Populate the **SMTP Server** field.

4 Leave the default (25) in the **Port** field.

5 In the **From Address** field, enter an email address from which notifications should be sent.

6 Click **Send Test Email.**

The confirmation message confirms the server connection and successful transmission to the email address you entered.



7 Click **OK.**

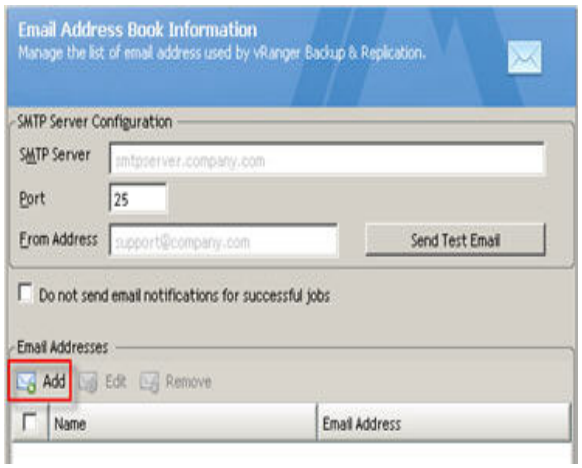
8 If desired, select **Do not send email notifications for successful jobs**.

Add Email Address

1 Select **Tools-->Options.**

2 Under the General node on the Configuration Options dialog, click **Email Configuration.**

3 In the **Email Addresses** section, click **Add.**



4 Click the **Add** button.

5 Enter the name and address information. Click **OK**.

Edit Email Address

1 Select **Tools-->Options.**

2 Under the **General** node, click **Email Configuration.**

3 In the **Email Addresses** section, select the email address that you want to edit. Click **Edit**.

4 Update the information and click **Update.**

5 The information is updated and moved to the **Email Addresses** section.

6 Click **OK.**

Remove Email Address

1 Select **Tools-->Options.**

2 Under the General node on the Configuration Options dialog, click **Email Configuration.**

3 In the **Email Addresses** section, select the email address that you want to delete.

4 Click **Remove.** The listing is deleted.

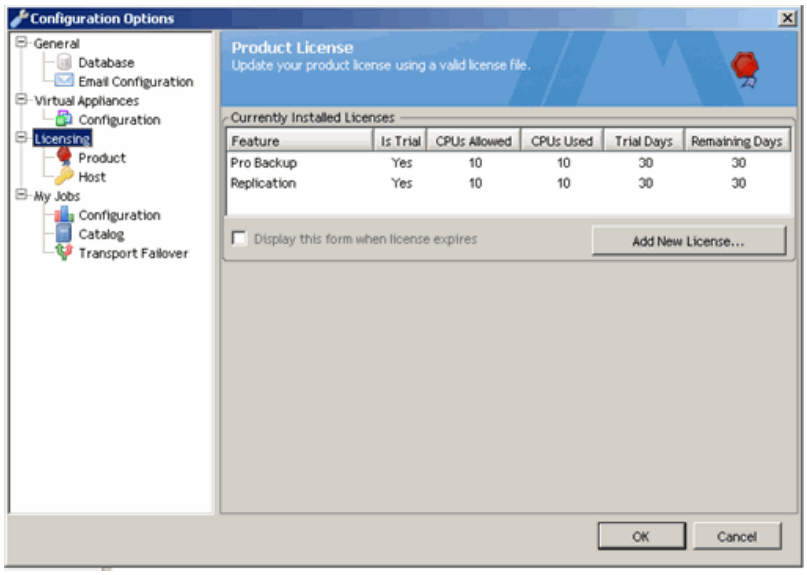
5 Click **OK.**

Import License File

Before you start this process, confirm that you have placed your license file in a location that you can reach easily.

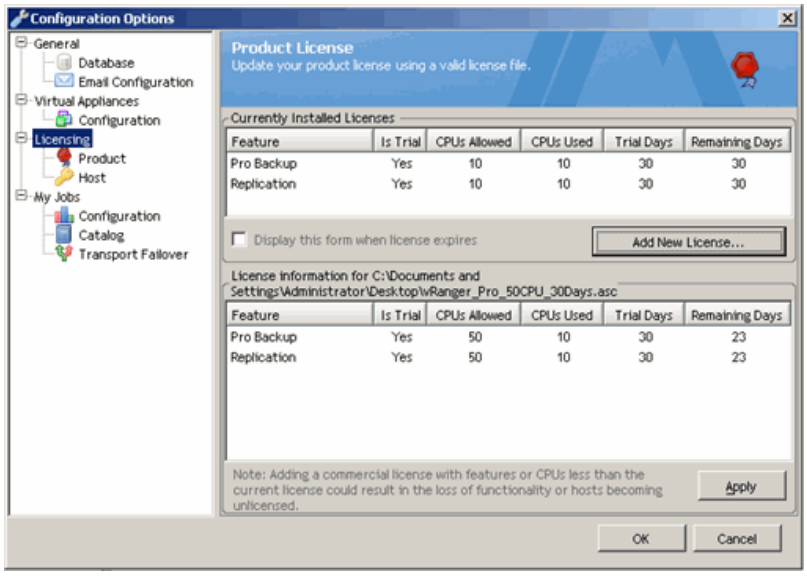
1 Select **Tools-->Options.**

2 Under the **Licensing** node on the **Configuration Options** dialog, click **Product.**



3 Click **Add New License.**

4 Browse to the location where you stored the license file. Click to select it and click **Open.** The new license information is added to the dialog.



5 Click **Apply.**

The dialog refreshes with the new license information added.

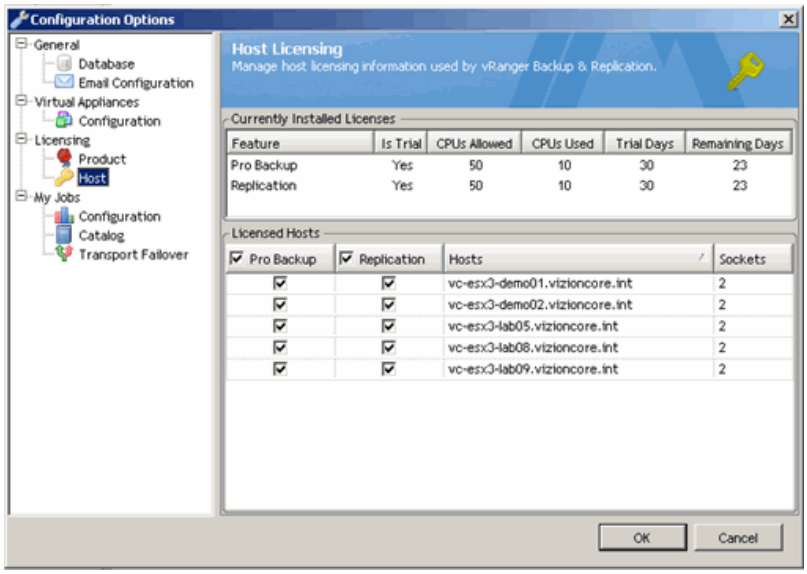
6 Click **OK.**

Manage Host Licensing

Before you start this process, make sure that you have imported the license file and confirmed that it is adequate for managing the hosts that you want to use. The Host Licensing Summary section includes the total number of CPU licenses, the number being used currently, and the number of licenses remaining. These numbers will increment and decrement based on the selections you make on this dialog.

1 Select **Tools-->Options.**

2 Under the **Licensing** node, click **Host.**



3 In the Currently Installed Licenses section, view the **CPUs Allowed** value to confirm the number of sockets that are licensed for protection. The number of licenses currently allocated is shown in the **CPUs Used** column.

4 The Licensed Hosts section lists the managed hosts, the number of sockets for each host, and the licensed features for that host.

   a The **Sockets** column indicates the license "cost" of each ESX(i) Host.

   b In the **Pro Backup and**/or **Replication** column, select or deselect ESX(i) hosts until you have used all of the available licenses, or until all of your ESX(i) hosts are protected.

5 Click **OK.**

Manage Job Configuration Settings

Through the **Configuration Options** dialog, you can set limits for the number of tasks running, the timeframe for task execution, and the space allowed on a datastore to accommodate snapshot growth.

The maximum number of tasks fields are defined in the table below:

On vRanger	the maximum number of backup, restore and replication tasks running on this instance of vRanger at any one time
off a LUN	the maximum number of tasks allowed on any particular logical storage device
on a host	the maximum number of tasks running on a host at any given time

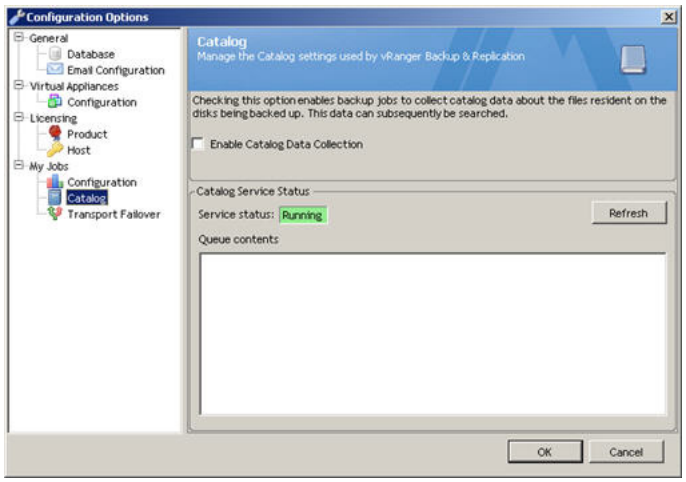
on each repository	the maximum number of tasks allowed on any specific data store at any specific time
locally	the number of backup, restore and replication tasks running on the machine on which vRanger is installed
per VA	the number of backup, restore and replication tasks running on a vRanger virtual appliance.

1. Select **Tools-->Options.**
2. Under the My Jobs node, click **Configuration.**
3. In the **Resource Management** section, set a maximum for the number of tasks that can be run:
   - on vRanger (>=1) -- Default is 100.
   - off a LUN (1-5) -- Default is 3.
   - on a host (1-4) -- Default is 1.
   - on each repository (>=1) -- Default is 3.
   - locally (1 - 20) -- Default is 3.
   - per VA (1 -18) -- Default is 2.
4. In the **Timeout** section, populate the **Task Timeout** (1-999 hours) field -- Default is 24
5. In the **Space needed on host** (in GBs) field, enter a number -- Default is 6.
6. Click **OK.**

Cataloging Options

The vRanger cataloging function must be enabled before data collection can begin. The Catalog dialog includes the option to enable or disable catalog collection, and displays the status of the Catalog Service.

In order for Catalog collections to function, the Catalog Service must display a status of “Running”.



1. Select **Tools-->Options.**
2. Under the My Jobs node, click **Catalog.**
3. Select **Enable Catalog Data Collection**.
4. Click OK.

Note Cataloging must also be enabled for each backup job for which you wish to collect catalog data.

Catalog Filtering

While there are thousands (or hundreds of thousands) of files in a typical VM, most are not relevant to file level recovery operations. In order to streamline cataloging operations, and reduce impact to the catalog database, vRanger filters files to be indexed in two ways:

- **Path** - by default, vRanger does not catalog any files in the directories listed below. Path filtering is determined by entries in the PathFilterTokens.txt file, located at *C:\Program Files\Quest Software\vRanger\Service\Configuration*.

Program Files	$Txf
Windows	RECYCLER
$Extend	System Volume Informa-
$TxfLog	tion
	I386

- **File** - By default, vRanger does not catalog files of the type below. File filtering is determined by entries in the FilesFilterTokens.txt file, located at *C:\Program Files\Quest Software\CatalogManager\Config\Files*.

.lnk	$Boot	$Reparse
$MFT	$BadClus	$RmMetadata
$Volume	$Secure	$Repair
$AttrDef	$UpCase	$Tops
$BitMap	$Quota	$TxfLog
	$ObjID	

Note File filtering applies to un-filtered paths. If a path is filtered, files in that path do not need to be.

For most situations, the default filtering options will be sufficient. If you want to filter out additional paths or files, simply add the path or file to the appropriate text file.

Configuring Linux File Level Restoration

To simplify the startup procedure, the instructions for deploying and configuring the virtual appliances have been moved to Configuring The vRanger Virtual Appliance. The same virtual appliance is used for Linux FLR and for replication.

Configuration Options

Through this dialog you can customize the basic settings that you selected in the Startup Wizard. You can access the Configuration Options dialog through Tools in the menu bar. The left pane of the dialog is set up in a nested structure.

General	• **Database**: Use the **Database** dialog to enter a database server address and a username. • **Delete Jobs data:** In order to reduce the size of your vRanger database, you may delete job data older than the configured date from the database. **Note** Deleted data will no longer be available for reporting. • **Email Server**: On this dialog, you can enter an SMTP server address and verify the connection to the port (default is 25). You can confirm successful transmission to the email address listed in the **From Address** field by clicking the **Send Test Email** button. • **Email Address Book**: Here you can add and manage the email addresses of recipients to whom you would like reports sent. **Note** The **New Email Address Section** becomes the **Edit Email Address** section when you select a listing in the **Email Addresses** section of the dialog.
Virtual Appliances	• **Configuration:** This dialog links to the vRanger virtual appliance, and allows you to add configured VA's for use by vRanger.
Licensing	• **Product License**: On this dialog you can view the current product licensing details and import a license when your current one expires or needs to be replaced. • **Host**: Use this dialog to allocate licensing to the hosts you added to the system. Refer to the **Host Licensing Summary** to view the total number of CPU licenses you are allowed, the number of licenses being used, and the number of licenses that remain.

My Jobs	**Configuration** • **Resource Management**: This section gives you access to task management information. **-Maximum number of tasks running on vRanger** (>=1) Default is 100. **-Maximum number tasks running off of a LUN** (1 - 5) Default is 3. **-Maximum number tasks running on a host** (1- 4) Default is 1. **-Maximum number tasks running per repository** (>=1) Default is 3. **-Maximum number tasks running locally** (1 - 20) Default is 3. **-Maximum number tasks running per VA** (1 - 18) Default is 2. • **Timeout**: In this section, you can set the period of time after which a task times out. The range is between 1 and 999 hours. • **Space needed on Hosts**: You can define the additional datastore space needed for snapshot growth during backup. Populate this field in GB--Minimum space needed on host (in GBs). The snapshot log file grows on the source while the VM is archived and copied to the destination. The default value is 6. **Catalog** • **Catalog Service:** Click the checkbox to enable catalog data collection • **Catalog Service Status:** Status is listed as **Available** or **Unavailable –** click **Refresh** to recheck the status. • **Queue contents:** lists catalog queues. **Transport Failover** Use this screen to set the relative priority for the available transport options. This priority will be used by vRanger to automatically address transport failures. These default options can be changed at the job level.

Supplemental Instructions - Additional Repository Types

The main procedures describe the process for adding a CIFS repository. To add an NFS, SFTP, FTP, or NetVault SmartDisk (NVSD) repository, see the appropriate section below.

- Adding an NFS Repository
- Adding an FTP Repository

- Adding an SFTP Repository
- Adding an NVSD Repository
- Adding an EMC Data Domain Boost (DDB) Repository

Adding an NFS Repository

The section below provides the information required to configure an NFS repository in vRanger.

NFS Requirements

- vRanger supports the NFS3 protocol, which does not provide for authentication. vRanger will use the NFSNobody account to connect to the repository.
- The NFSNobody account must have R,W, X permissions for the target directory.
- Your */etc/exports* file must contain the export directory and a list of the IPs that will connect to the repository. You may list each individually (the vRanger machine and any ESX hosts using direct-to-target) or use a wildcard *. The export directory should be shared with read/write asynchronous access. For an export directory of */backups*, this would look like:

```
[root@NFSServer etc]# cat exports
/backups        *(rw,async)
```

To add an NFS Repository:

1 To add an NFS repository, navigate to the My Repositories pane. Right-click anywhere in the My Repositories pane. Select **Add--> NFS.**

1 On the **Add Network File Share Repository** dialog, populate the fields below:

   a **Repository Name** - a descriptive name for the repository.

   b **Description** - a long form description for the repository. This is an optional field.

   c **DNS Name or IP** - the IP or FQDN for the repository

   d **Export Directory** - specify the Export directory, which is similar in concept to a network share. You will need to create a target sub-directory in the export directory.

   e **Target Directory** - this is a sub-directory of the NFS Export directory. This is the location to which savepoints will be written.

2 Select **Encrypt all backups to this repository** if you wish these backups to be password-protected.

3 Enter a Password for the encrypted repository -- confirm the password by re-entering it.

   Caution If you wish to use the **Encrypt all backups** feature**, make certain to retain** the password you enter in the following steps. There is no back-door or admin-level password. If the password is unknown, the backups will not be usable.

4 Click **OK** -- the connection to the repository is tested and the repository is added to the My Repositories pane and the Repository Information dialog.

5 vRanger will check the configured repository location for existing manifest data to identify existing savepoints. If any are found, you will be prompted to take one of three actions:

- Import as Read-Only - with this option, all savepoint data will be imported into the vRanger database, but only for restores. You will not be able to backup to this repository.
- Import -all savepoint data will be imported into the vRanger database. vRanger will be able to use the repository for backups and restores. vRanger requires read and write access to the directory.
- Overwrite - the savepoint data will be retained on the disk, but will not be imported into vRanger. vRanger will ignore the existence of the existing savepoint data and treat the repository as new.

Adding an FTP Repository

The process for adding an FTP repository is described below.

To add an FTP Repository:

1. To add an FTP repository, navigate to the My Repositories pane. Right-click anywhere in the My Repositories pane. Select **Add--> FTP.**
2. On the **Add FTP Repository** dialog, populate the fields below:
   a. **Repository Name** - a descriptive name for the repository.
   b. **Description** - a long form description for the repository. This is an optional field.
   c. **DNS Name or IP** - the IP or FQDN for the repository
   d. **Target Directory** - this is the directory on the FTP server to which the repository will be written.
   e. **User name** - the user name for the account that vRanger will use to connect to the repository.
   f. **Password** - the password for the user account above.
   g. **Port** - the port that vRanger will use to connect to the FTP server.The default FTP port is 21
   h. **Timeout** - this value determines how long to wait before terminating an idle connection. The default value is 600 seconds
3. Select **Encrypt all backups to this repository** if you wish these backups to be password-protected.
4. Enter a Password for the encrypted repository -- confirm the password by re-entering it.

   Caution If you wish to use the **Encrypt all backups** feature**, make certain to retain** the password you enter in the following steps. There is no back-door or admin-level password. If the password is unknown, the backups will not be usable.
5. Click **OK** -- the connection to the repository is tested and the repository is added to the My Repositories pane and the Repository Information dialog.
6. vRanger will check the configured repository location for existing manifest data to identify existing savepoints. If any are found, you will be prompted to take one of three actions:
   - Import as Read-Only - with this option, all savepoint data will be imported into the vRanger database, but only for restores. You will not be able to backup to this repository.

- Import -all savepoint data will be imported into the vRanger database. vRanger will be able to use the repository for backups and restores. vRanger requires read and write access to the directory.
- Overwrite - the savepoint data will be retained on the disk, but will not be imported into vRanger. vRanger will ignore the existence of the existing savepoint data and treat the repository as new.

Adding an SFTP Repository

The process for adding an SFTP repository is described below.

To add an SFTP Repository:

1 To add an SFTP repository, navigate to the My Repositories pane. Right-click anywhere in the My Repositories pane. Select **Add--> SFTP.**

1 On the **Add SFTP Repository** dialog, populate the fields below:

a **Repository Name** - a descriptive name for the repository.

b **Description** - a long form description for the repository. This is an optional field.

c **DNS Name or IP** - the IP or FQDN for the repository

d **Target Directory** - this is the directory on the SFTP server to which the repository will be written.

e **User name** - the user name for the account that vRanger will use to connect to the repository.

f **Password** - the password for the user account above.

g **Port** - the port that vRanger will use to connect to the SFTP server.The default SFTP port is 22.

h **Timeout** - this value determines how long to wait before terminating an idle connection.

2 Select **Encrypt all backups to this repository** if you wish these backups to be password-protected.

3 Enter a Password for the encrypted repository -- confirm the password by re-entering it.

Caution If you wish to use the **Encrypt all backups** feature**, make certain to retain** the password you enter in the following steps. There is no back-door or admin-level password. If the password is unknown, the backups will not be usable.

4 Click **OK** -- the connection to the repository is tested and the repository is added to the My Repositories pane and the Repository Information dialog.

5 vRanger will check the configured repository location for existing manifest data to identify existing savepoints. If any are found, you will be prompted to take one of three actions:

- Import as Read-Only - with this option, all savepoint data will be imported into the vRanger database, but only for restores. You will not be able to backup to this repository.
- Import -all savepoint data will be imported into the vRanger database. vRanger will be able to use the repository for backups and restores. vRanger requires read and write access to the directory.

- Overwrite - the savepoint data will be retained on the disk, but will not be imported into vRanger. vRanger will ignore the existence of the existing savepoint data and treat the repository as new.

Adding an NVSD Repository

The process for adding a NetVault SmartDisk (NVSD) repository is described below.

To add an NVSD Repository:

1 To add an NVSD repository, navigate to the My Repositories pane. Right-click anywhere in the My Repositories pane. Select **Add--> NVSD.**

1 On the **Add NVSD Repository** dialog, populate the fields below:

a **Repository Name** - a descriptive name for the repository.

b **Description** - a long form description for the repository. This is an optional field.

c **DNS Name or IP** - the IP or FQDN for the NVSD server.

d **User name/Password** - the credentials for the account that vRanger will use to connect to the repository.

Note If you have enabled WebDav authentication on your NVSD server, then use those credentials here. If you have not enabled WebDav authentication, then no credentials are required. See the *Quest NetVault SmartDisk Administrators Guide* for more information.

e **Target Directory** - this is the directory on the NVSD server to which the repository will be written. You may select an existing NVSD repository (if one exists) from the drop-down, or enter a name for your repository to have it created.

f **Port** - the port that vRanger will use to connect to the NVSD server.The default port is 37453.

g **Timeout** - this value determines how long to wait before terminating an idle connection.

Note Note that encryption is not supported for NVSD repositories.

2 Click **Ok**.

3 Click **OK** -- the connection to the repository is tested and the repository is added to the My Repositories pane and the Repository Information dialog.

4 vRanger will check the configured repository location for existing manifest data to identify existing savepoints. If any are found, you will be prompted to take one of three actions:

- Import as Read-Only - with this option, all savepoint data will be imported into the vRanger database, but only for restores. You will not be able to backup to this repository.
- Import -all savepoint data will be imported into the vRanger database. vRanger will be able to use the repository for backups and restores. vRanger requires read and write access to the directory.
- Overwrite - the savepoint data will be retained on the disk, but will not be imported into vRanger. vRanger will ignore the existence of the existing savepoint data and treat the repository as new.

Adding an EMC Data Domain Boost (DDB) Repository

The process for adding an EMC Data Domain Boost repository is described below. This procedure assumes the following:

- That you have a properly configured Data Domain appliance that is accessible to the vRanger machine.
- That DD Boost is licensed and enabled on the Data Domain device.
- That you have designated a DD Boost User account.

Note Please consult the Data Domain documentation for information on performing the configurations listed above.

To add an EMC Data Domain Boost Repository:

1. To add a DD Boost repository, navigate to the My Repositories pane. Right-click anywhere in the My Repositories pane. Select **Add--> EMC Data Domain Boost (DDB).**
2. On the **Add EMC Data Domain Boost Repository** dialog, populate the fields below:
   a. **Repository Name** - a descriptive name for the repository.
   b. **Description** - a long form description for the repository. This is an optional field.
   c. **DNS Name or IP** - the IP or FQDN for the Data Domain device.
   d. **User name/Password** - these are the credentials for the account selected as the **DD Boost User** on the Data Domain device.

      Note Please consult the Data Domain documentation for more information on setting a DD Boost user name.
3. **Storage Unit -** this is the Storage Unit configured on the Data Domain device. Select one from the drop-down menu, or enter the name for a new Storage Unit and vRanger will create one for you.Click **OK** -- the connection to the repository is tested and the repository is added to the My Repositories pane and the Repository Information dialog.

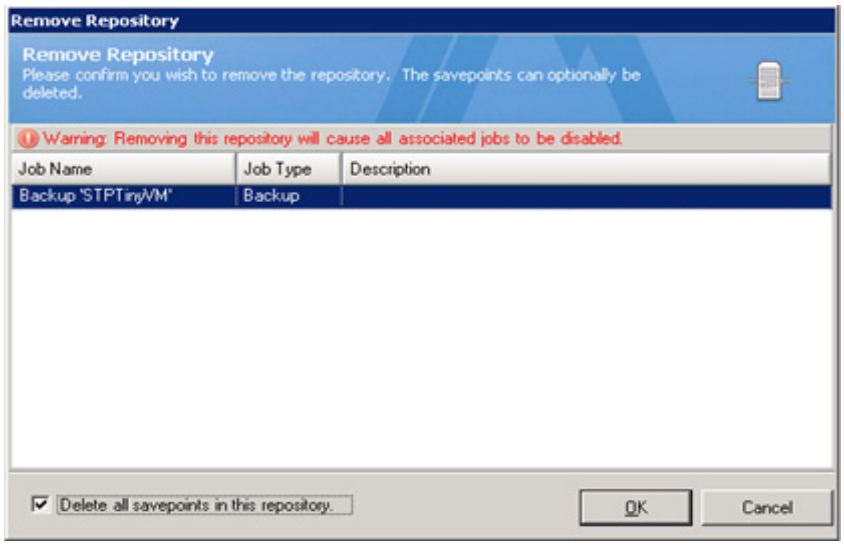
   Note For Data Domain devices running DD OS 5.0, the only characters allowed in storage unit names are alphanumeric characters, “-”, and “_”. For devices running DD OS 5.1, the supported characters are alphanumeric characters and ~!@#$^_+`-={}[],.
4. vRanger will check the configured repository location for existing manifest data to identify existing savepoints. If any are found, you will be prompted to take one of three actions:
   - **Import as Read-Only** - with this option, all savepoint data will be imported into the vRanger database, but only for restores. You will not be able to backup to this repository.
   - **Import** -all savepoint data will be imported into the vRanger database. vRanger will be able to use the repository for backups and restores. vRanger requires read and write access to the directory.
   - **Overwrite** - the savepoint data will be retained on the disk, but will not be imported into vRanger. vRanger will ignore the existence of the existing savepoint data and treat the repository as new.

Removing an EMC Data Domain Boost (DDB) Repository

The process for removing a Data Domain Boost repository is the same as removing any other repository type, except that you also have the option of removing the storage unit used by the repository. There is a hard limit on the number storage units that can be configured on a Data Domain device, which makes removing unused storage units an important maintenance effort.

To remove a DD Boost repository:

1. From the My Repositories view, right-click on the Data Domain Boost repository and select **Remove**.
2. The **Remove Repository** dialogue appears, showing the savepoints in the selected repository.

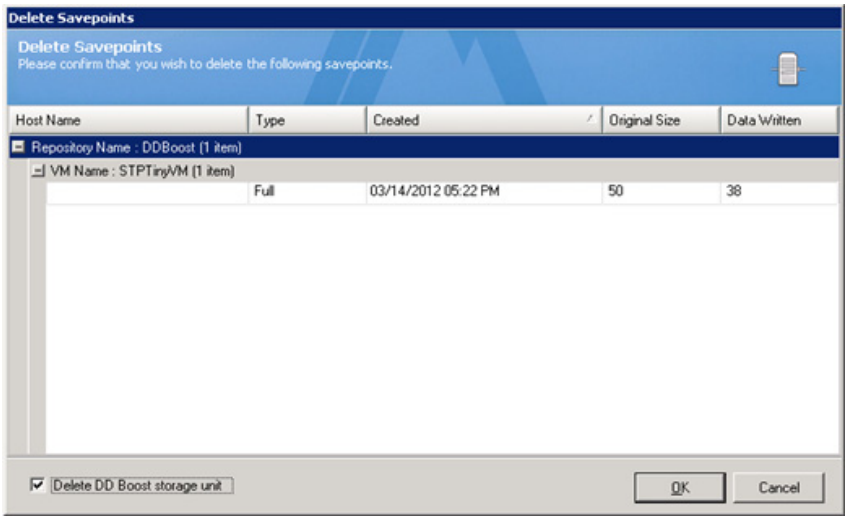


When removing a repository, you have the option of keeping the savepoints on disk or deleting them. In order to remove the storage unit associated with the repository, you will need to remove the savepoints.

3. Select **Delete all savepoints in this repository**. Click **OK**

   Caution This will delete the savepoints from the repository disk, not just the vRanger database. Exercise caution when deleting savepoints.

4. The **Delete Savepoints** dialogue appears.



Select **Delete DD Boost storage unit**, and click **OK**. The savepoints will be deleted, along wit the storage unit associated with the repository.

Configuring The vRanger Virtual Appliance

vRanger uses a virtual appliance for replication to and from ESXi hosts, for file-level recovery (FLR) from Linux machines, and optionally for backups and restores. You may also use the VA on ESX hosts as well to avoid taxing the Service Console resources, which are limited.

Virtual Appliance Overview

The vRanger virtual appliance is a lightweight virtual machine that serves as an operating environment for the components that would normally run in the Service Console. Removing the workload from the Service Console is clearly necessary for ESXi and vSphere 5, but it is advantageous for ESX servers as well. While the Service Console is restricted to a single CPU (CPU 0), with virtual appliances the workload can be spread across the other CPUs available to a host. This provides increased reliability and scalability over operations limited to the Service Console.

The virtual appliance must be deployed to any ESXi host that you will configure for replication (either as a source or a destination). For hosts in a cluster, you may deploy just one VA to the cluster; the VA will be shared among the cluster's other hosts. When deploying a VA to a cluster, you must chose a host in the cluster to which the VA will be associated.

In addition, replication via virtual appliance requires that if a VA is used on one host or cluster in a replication job, then a VA must be used on both the source and destination host or cluster. In other words, virtual appliances, when used for replication, must be used in pairs.

When configuring the virtual appliance, special consideration should be given to the amount of resources (CPU and RAM) allocated to the VA as the number of simultaneous tasks the VA will be able to process is directly tied to available resources. In addition, if you will be performing replication tasks using a virtual appliance, carefully consider an appropriate size for the VA scratch disk. See the section below for more information on scratch disk usage.

The Virtual Appliance Scratch Disk

If you will be performing replication tasks with your virtual appliance, you will need to add a second "scratch" disk to the VA. This scratch disk is used to store two types of files:

- *vzmap* files - block maps (in the form of a *vzmap* file) for the VMs replicated to the destination host. This is block map information, and not actual data blocks. These maps are compared to the source VM during each replication pass to identify the data blocks that have changed since the last replication. The *vzmap* files make differential replication significantly faster as they remove the need to scan the destination VM blocks for comparison with the source VM.
- *vzUndo* files - As data is sent to the destination host (via the VA), blocks in the destination disk are written to the undo file before they are overwritten by the changed data. If replication fails and an undo becomes necessary, then the original destination disk blocks are read from the undo file and written to the destination disk to rollback the failed replication. This is a key function designed to provide resiliency in the face of a network failure; if there

is a network failure during the replication pass, the destination VM is not corrupted by incomplete data.

Once the pass is complete, and all data has been received by the destination VA, the undo file is deleted. At that point, the storage space used by the undo file is returned to the VA for use. Note that undo files are not created during the first pass of a replication job. During the first pass, the entire VM is sent to the destination host, but there is no existing data on the destination VMDKs, and therefore no risk of corruption. Data is streamed directly to the VMDK. You do not need to allocate scratch disk space for this scenario.

While the vzmap files are trivial in size (in the order of a few MBs), the undo file can potentially be as large as the VM itself. While the scratch disk needs to be configured to a size sufficient to handle the data of concurrent replication tasks, making it too large wastes valuable storage space. Use the sections below to guide you in determining the proper size for the scratch disk.

Strategies for Sizing the Scratch Disk

The scratch disk needs to be large enough only to hold the permanent *vzmap* files and the temporary *vzUndo* files, plus a small margin for safety. How large that is depends almost entirely on the amount of changed data you are replicating. The amount of changed data is itself a function of the number of VMs you'll be replicating, their total disk size, replication frequency, and the data change rate per VM. It is important to understand all of this data when sizing the scratch disk.

If you are using one VA for a cluster, remember that you must account for all simultaneous replications for the cluster.

Use Historical Data

If you have previously replicated the source VMs, the most accurate method to properly size the scratch disk (without wasting storage space) is to use historical replication data available in the Replicate Task Reports (in the vRanger My Reports view) for the VMs in question. This report will show the amount of data written during each replication task.

The safest method to size your scratch disk based on historical data is to record the highest amount of data written for each VM that you will replicate at once, and size the disk to accommodate those values.

To avoid filling your scratch disk, Quest Software recommends that you add a small margin (10% or so) to the calculated scratch disk size for safety.

Calculating

If you do not have information on the amount of changed data for each VM, you can estimate the appropriate size of the scratch disk based on the VM size and the number of VMs you will be replicating at once.

A general rule for sizing the scratch disk is to choose a percentage of the total VM size to represent the practical limit of changed data for a given replication pass. Only you can decide what is appropriate for your environment. The numbers below are examples given to illustrate the concept:

Assume that you will have four virtual machines that you wish to replicate to a host or cluster at the same time. These VMs are described in the table below.

VM	V M Size	Change Rate	Change Size
1	100 GB	15%	15 GB
2	100 GB	10%	10 GB
3	100 GB	20%	20 GB
4	60 GB	5%	3 GB

For the VMs above, we would need approximately 48 GB of disk space for the undo files, plus a buffer of approximately 10% (for safety's sake). In our example, an appropriate estimate for the scratch disk size for the VMs above would be approximately 55 GB.

Bear in mind that the estimate exercise should be done for every set of VMs that will be replicated to that host or cluster, with the scratch disk being sized to accommodate the largest value obtained.

Caution If the scratch disk runs out of space, replication tasks will fail with the error "*FATAL cant_write can't write vix".*

Options for a Smaller Scratch Disk

As previously stated, the primary driver for the scratch disk size is the amount of changed data that needs to be replicated. If you need to reduce the storage requirements for your scratch disk, you can:

- reduce the amount of time that passes between replication intervals. More frequent replication passes will contain smaller amounts of changed data.
- reduce the number of VMs that you replicate simultaneously. This will send less data through the scratch disk at any one time, allowing it to be smaller. Remember that the *vzundo* files are deleted after the replication pass completes.
- when provisioning the scratch disk, select the Thin Provisioning option. Thin provisioning helps prevent wasted storage space by not allocating blocks until they are written by the VM.

The Scratch Disk on the Source Host

As the scratch disk is used primarily for staging changes before they are written to disk – activity which occurs on the destination host or cluster– the scratch disk on the source side can be kept fairly small. However, in the event that you need to fail over to the D/R site, the replication job will reverse direction and start replicating changes back to the product site (the original source host or cluster). In order for this to occur, the scratch disk on the source side will need to be re-sized to accommodate the changed data.

Scratch Disk Location

When creating the second disk, make sure you place the disk on a datastore with block sizes large enough to support the expected VMDK. The list below shows the maximum file size available for each block size:

- 1 MB block size – 256 GB maximum file size
- 2 MB block size – 512 GB maximum file size
- 4 MB block size – 1024 GB maximum file size
- 8 MB block size – 2048 GB maximum file size

Note This limitation does not apply to VMFS 5.

Deploying the Virtual Appliance

vRanger uses a virtual appliance for both Linux file-level recovery and for replication to and from VMware ESXi servers. The vRanger virtual appliance is available via the Quest website: http://www.quest.com/landing/?id=5414.

Also available on this landing page are:

- A demonstration video describing the VA configuration process
- A short form of the VA Installation Instructions
- Virtual Appliances for older versions of vRanger

Deploying Multiple Virtual Appliances

The vRanger virtual appliance is required when replicating to or from an ESXi server. In addition, replication via virtual appliance requires that if a VA is used on one host in a replication job, then a VA must be used on both hosts. For example, if you are replicating between an ESXi host and an ESX host, you must have a VA deployed on the ESX host as well.

To streamline the deployment process, Quest recommends that you deploy and configure the virtual appliance once, then save it as a template to be used for additional deployments. The Configuring the Virtual Appliance procedures include an optional step for creating a template from the configured VA.

From this template, you have two options to deploy additional VA instances; a manual process using the VI Client, or a scripted process using PowerCLI. Regardless of which process you choose, if you wish to use a static IP address for the virtual appliances, you will need to configure that manually. Instructions for template deployment and network configuration can be found in the sections below:

- Deploying Multiple VAs From a Template - VI Client

- Deploying Multiple VAs From a Template - PowerCLI

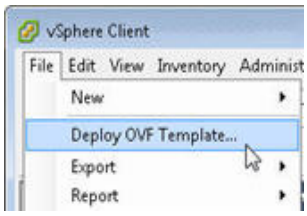
Note The procedures below were capture using VMware vCenter 4.1, update 1. The procedures may differ slightly for previous versions of vCenter. Consult your VMware documentation for the correct procedure for deploying a virtual appliance for your version of vCenter.

To deploy the virtual appliance:

1. Obtain the OVA file for the vRanger virtual appliance. For vRanger 5.4.1, the correct OVA version is 1.6.0.

   Note For the 5.4.1 release of vRanger, this file will be provided directly to customers. Please refer to the *vRanger 5.4.1 Release Notes* for more information.

2. In your vSphere VI Client, click **File**, then **Deploy OVF Template**.



3. In the **Deploy from a file or URL** field, enter the path to the downloaded OVA file. Click **Next**.
4. Continue through the import process, selecting the appropriate host and datastore configuration.

   The OVA import process may take several minutes. When complete, continue to the configuration section below.

Configuring the Virtual Appliance

Before using the virtual appliance, some changes must be made to the default configuration.

Step 1: Change the Default Password

1. From your VI Client, find the VA and power it on.
2. Using the console, login as `root` to the virtual appliance, using the default credentials below:
   - username: root
   - password: vzroot1
3. At the prompt, enter `passwd`. Enter and re-type a new password for the `root` account.

   Caution This is an important security step. Please ensure that you change the default password for the root account.

Step 2: Install VMware Tools

1 From the VI Client, right-click on the virtual appliance. Click **Guest**, then **Install/Upgrade VMware Tools**.

Note This must be done within 5 minutes of powering on or rebooting the virtual appliance. The virtual appliance will automatically install the VMware Tools.



Note No input is necessary for the VMware Tools installation - all prompts will be answered by the installer script.

Step 3: Changing the Hostname

1 Login to the virtual appliance.

2 Using the VI text editor, edit the network configuration file:

```
vi /etc/sysconfig/network
```

Note See Appendix: Using the VI Editor for more information on using the VI editor.

3 Change the hostname to the new value. Save the file.

Step 4: Changing the Network Configuration [Optional]

The virtual appliance is configured by default to use DHCP. You may choose to re-configure network settings to change the hostname, or to use a static IP address.

Note If you will continue using DHCP. Please note that an expired lease or changed address may result in vRanger losing connection with the virtual appliance. MAC Address reservations may be an option utilized for this scenario. If the IP of the virtual appliance changes, you will need to perform an inventory refresh in the vRanger My Inventory view before vRanger can use the virtual appliance(s).

1 Login to the virtual appliance.

2 Using the VI text editor, create a configuration file for eth0:

```
vi /etc/sysconfig/network-devices/ifconfig.eth0/ipv4
```

3 Enter the following text, substituting the appropriate IP address where indicated:

```
ONBOOT=yes
SERVICE=ipv4-static
IP=<static IP of VA>
GATEWAY=<IP of gateway>
PREFIX=<this is the subnet mask - your values may differ>
BROADCAST=<your broadcast IP- the highest IP in the VA's subnet.>
```

4 Save the file.

5 Move the DCHP configuration file to the root directory. This will disable DHCP.

```
mv -v /etc/sysconfig/network-devices/ifconfig.eth0/dhcpcd /root/
```

6 Restart the VM by typing `reboot` at the prompt.

Step 5: Configuring DNS [Optional]

It may be necessary, depending on your environment, to configure the virtual appliance with DNS nameserver information.

1 Login to the virtual appliance.

2 Using the VI text editor, edit the *resolv.conf* file:

```
vi /etc/resolv.conf
```

3 Enter the following text, substituting the appropriate values where indicated:

`search` <*domain1*> <*domain2*> <*domain3*>

`nameserver` <*IP address*>

`nameserver` <*IP address*>

4 Save the file.

5 Restart the VM by typing `reboot` at the prompt.

Step 6: Adding Second "Scratch" Disk

The vRanger virtual appliance uses a scratch disk to stage replicated data to an "undo" file, ensuring that the complete transmission is received before writing to the destination VM. In the event of a network (or other) failure, this prevents the replica VM from becoming corrupted.

See Strategies for Sizing the Scratch Disk for guidance on how to select the best size for the scratch disk.

To add the scratch disk:

1 From the VI Client, right-click on the virtual appliance, and select **Edit Settings**.

2 On the Hardware tab, click **Add**.

3 Select **Hard Disk**. Click **Next.**

4 Select **Create a new virtual disk**.

Configure the size for the disk and select your provisioning method. If storage space is a concern, consider selecting the Thin Provision option. This ensures that only the amount of space required is allocated. For more information on thin provisioning, see the VMware KB article *Using thin provisioned disks with virtual machines*. Click **Next**.

5 Leave the **Virtual Device Node** setting as **SCSI**. Click **Next**.

6 Click **Finish**. Click **Ok**.

Caution Do not reboot the virtual appliance. Rebooting the virtual appliance at this point will prevent the VA Utility script from functioning properly.

Step 7: Load VA Utility script

The *vautil_01.02.01.tar.gz* needs to be manually transferred to the VA. This script prepares the scratch disk for replication.

1. Download the *vautil_01.02.01.tar.gz* files
2. Transfer the files to the VA. You may use scp from a Linux machine, or WinSCP from a Windows machine. An example command is shown below:

   ```
   scp vautil_01.02.01.tar.gz root@10.10.10.10:/root/bin/
   ```

3. Login to the VA and extract the utility scripts.

   ```
   tar zxvf vautil_01.02.01.tar.gz
   ```

4. Using the cd command, change the working directory into the extracted directory.

   ```
   cd /root/bin/vautil_01.02.01
   ```

5. Run the following command:

   ```
   /root/bin/vautil_01.02.01/scratch_scsi.sh -a
   ```

   The command, when successful, will display the following output:

   ```
   ADDED:  <device_added's_name> (Example:  ADDED:  sdb)
   ```

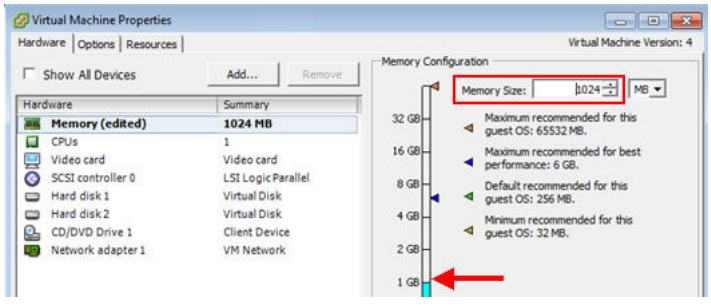
   Note Note that if you added more than one new disk, you will see more than one device name added. (Example: ADDED: sdb sdc)

Step 7: Resource Configuration

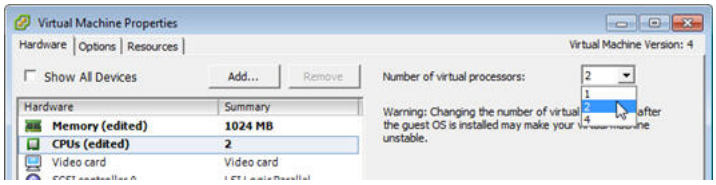
The VA is configured by default with 1 vCPU and 512 MB of RAM. This is sufficient to allow 2 concurrent tasks per VA. If you will need to run more than 2 tasks concurrently, increase the resources allocated to the VA.

Note If you observe consistent high CPU and Memory utilization on the virtual appliance, consider adding more resources ensure uninterrupted performance.

1. From the VI Client, right-click the virtual appliance and select **Edit Settings**.
2. In the Virtual Machine Properties dialog, select **Memory**. Move the slider, or use the up arrow in the **Memory Size** field to adjust the allocated memory.



3. In the Virtual Machine Properties dialog, select **CPU.** Select the desired number of vCPUs from the drop-down.



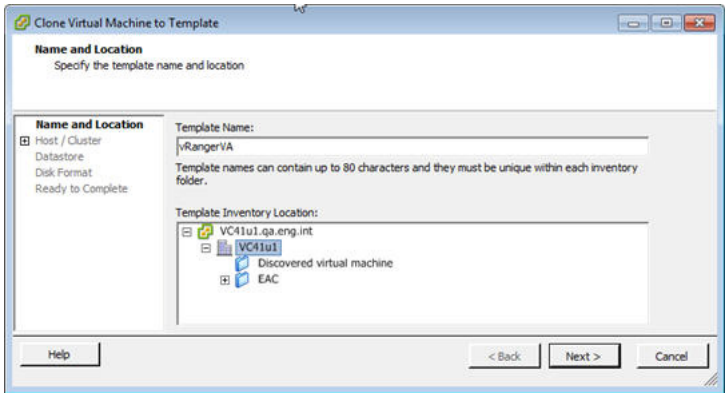
4 Click **Ok** to save the configuration changes.

Step 9: Creating a Template (Optional)

If you need to deploy the vRanger virtual appliance to multiple hosts, it may be helpful to create a virtual machine template from the configured virtual appliance.

1 From the VI Client, right-click the configured virtual appliance and select **Template>Clone to Template**.

2 Enter the template name and select a location. Click **Next**.



3 Select the host or cluster on which the template is to be stored. Click **Next**.

4 Select the datastore in which the template is to be stored. Click **Next**.

5 On the Disk Format dialog, select **Same format as source**. Click **Next**.

6 Review the configurations and click **Finish**.

Deploying Multiple VAs From a Template - VI Client

Unless you have only one ESXi host, you will need to deploy multiple instances of the vRanger virtual appliance. You may deploy additional instances either by using the VI client (as documented below), or via a PowerCLI script as documented in Deploying Multiple VAs From a Template - PowerCLI

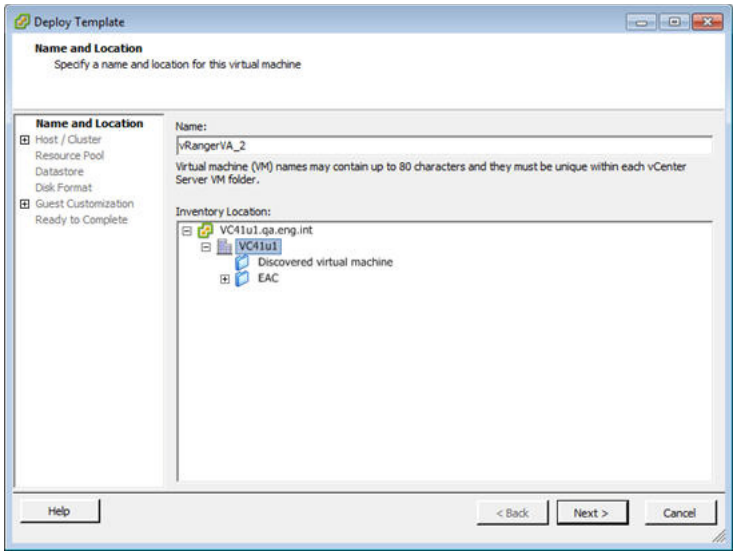
Note If you wish to use a static IP address, or need to configure DNS information, that must be done manually for each deployed instance - see Configuring the Virtual Appliance for instructions.

To deploy an instance of the vRanger virtual appliance from the template created in Step 7: Resource Configuration, follow the procedures below.

1 In the VI Client, find the template you created previously. To view templates, click **View>Inventory>VMs and Templates**.

2 Right-click on the template, and select **Deploy Virtual Machine from this Template**.

3 In the **Name and Location** dialog, enter a name for the VA, and select the appropriate inventory location. Click **Next**.



4 In the **Host/Cluster** dialog, select the host on which the VA is to be run. Click Next.

Note You will need to deploy an instance of the vRanger VA to every ESXi host used as a replication source or destination.

5 Select the datastore in which the VA files will be stored. Click **Next**.

6 In the **Disk Format** dialog, select **Same format as source**. Click **Next.**

7 In the **Guest Customization** dialog, select **Do not customize**. Click **Next**.

8 Review the selections. Click **Finish**.

9 Repeat steps 1-8 as needed for other hosts as needed.

Note After the virtual appliances have been deployed, refer to "Configuring the Virtual Appliance" on page 50 for instructions on how to configure IP and DNS information.

Deploying Multiple VAs From a Template - PowerCLI

Using vSphere PowerCLI and a pre-configured CSV file, you can automate the deployment of the vRanger virtual appliance to multiple hosts. This method requires the initial deployment using DHCP, although you can change the networking options for the virtual appliance after deployment using the instructions in Configuring the Virtual Appliance.

The high level steps required for this process are:

- Using the VI Client, create a Customization Specification

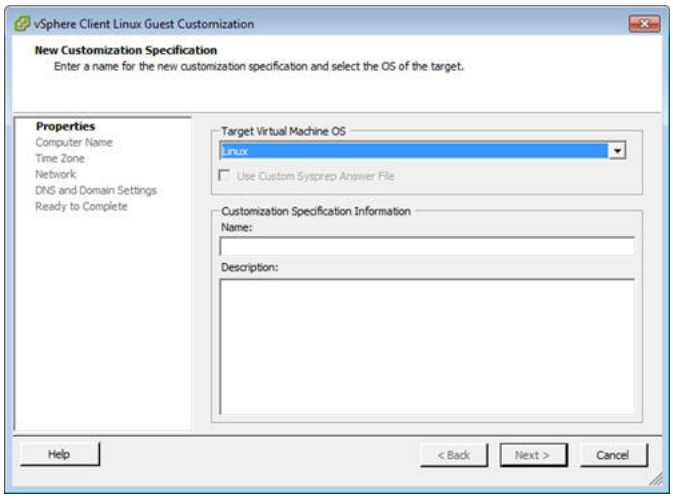
- Populate a CSV with VM, Host, and Datastore information
- Execute the PowerCLI Script

Step 1: Customization Specification

The Customization Specification provides basic configuration information used in creating the VMs.

1 From the VI Client, click V**iew>Management>Customization Specifications Manager**. Click **New**.

The **New Customization Specification Wizard** appears:



2 In the **Target Virtual Machine OS** field, select Linux. Enter a name for the customization specification. Click **Next**.

Note You will need the customization specification name for the deployment script. In the sample script below, the name *vRangerVA* is used.

3 On the Computer Name dialog, select **Use the virtual machine name**. Enter a domain name for the VM. Click **Next**.

Note The VM name will be configured from the CSV file.

4 Select an **Area** and **Location** for the VA's time zone. Click **Next**.

5 On the **Network** dialog, select **Typical Settings**. Click **Next**.

6 Enter a **Primary DNS** server and **DNS Search Path**. Click Next.

Note Due to compatibility issues between the VA OS and VMware Tools, this information will **not** be populated to the deployed VA. The wizard requires at least one DNS server entered before you can continue.

7 Review your configurations and click **Finish**.

Step 2: Populating a CSV

The information required for deploying additional virtual appliances will come from a CSV file imported by the PowerCLI script.

1 Create a CSV file with the headings below:

- VM
- Datastore
- VMhost
- Template

Note It is easiest to create a CSV file in a spreadsheet program such as Microsoft Excel. Once you save the file as a CSV however, only open the file using Notepad.

2 Populate the rows below the headers with the appropriate information:

- VM - this is the name of the VA that will be deployed
- Datastore - the datastore to which the VA will be deployed
- Host - the Host on which the VA will run.
- Template - the name of the template created in Step 7: Resource Configuration.

3 Save the file as a CSV. For best results, open the file using Notepad (or similar) to verify formatting. In the sample script below, this file is named *VADeploy.csv*

Step 3: Executing the PowerCLI Script

The commands below are entered using vSphere PowerCLI.

1 Enter the command below to connect PowerCLI to your vCenter. Substitute the correct values for your environment.

```
Connect-VIServer <vCenter> -User <username>-Password <password>
```

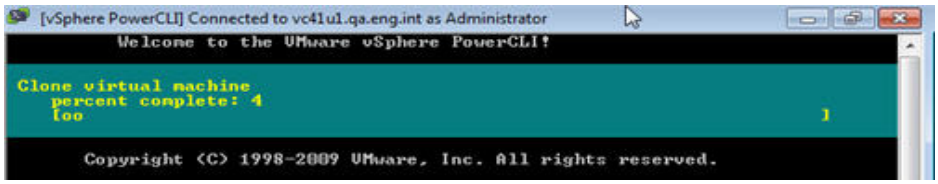
2 Enter the following commands to deploy your virtual appliances:

```
Import-Csv VADeploy.csv | %{
Get-OSCustomizationSpec vRangerVA | Get-OSCustomizationNicMapping | Set-
OSCustomizationNicMapping -IpMode UseDHCP
New-VM -Vmhost $_.VMhost -name $_.VM -Template $_.Template -Datastore
$_.Datastore -OSCustomizationSpec vRangerVA
}
```

The example above uses information that will vary based on your environment, as highlighted in red and italics.

- VADeploy.csv - the name of your CSV file.
- vRangerVA - the name of your Customization Specification.

A status bar indicates deployment progress:



Disabling DRS for the vRanger Appliance

If the vRanger appliance is configured on hosts that are part of a DRS enabled cluster, the VA(s) may be migrated to a different host due to DRS/HA activity. This will cause replication failures as the virtual appliance is no longer in the expected location (as configured in vRanger). To prevent this disruption, it is important to disable DRS for the vRanger virtual appliances.

1. In your VI client, find the cluster(s) containing the vRanger virtual appliances.
2. Right-click on the cluster, and select **Edit Settings**.
3. In the Cluster Settings dialog, select **Virtual Machine Options.**
4. Search for or find the vRanger virtual appliances. Right-click on each virtual appliance, and select **manual**.
5. Click **Ok**.

Adding the Virtual Appliance to vRanger

Once the virtual appliance(s) are configured, you will need to add them to the vRanger Virtual Appliances inventory.

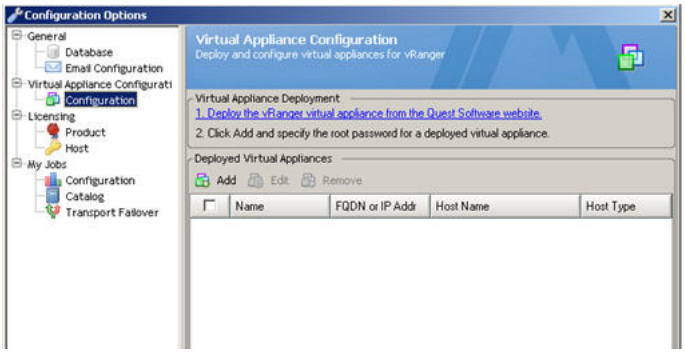
You may add one or more virtual appliances to vRanger during the **Startup Wizard** displayed when vRanger is first launched.

You may also launch the **Startup Wizard** at any point from the Tools menu, or use the **Tools>Options** to launch the **Configuration Options** dialog.

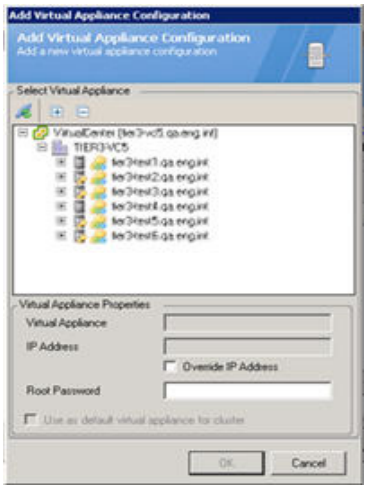
To add a virtual appliance to vRanger:

1. From the vRanger UI, select the **My Inventory** view, and click the refresh icon  to refresh the inventory.

1. From the **Configuration Options** dialog, select **Virtual Appliances**.



2. Click **Add**. The **Add virtual appliance configuration** dialog appears, showing the vCenter inventory tree.



3 Browse the inventory tree, and select the desired virtual appliance.

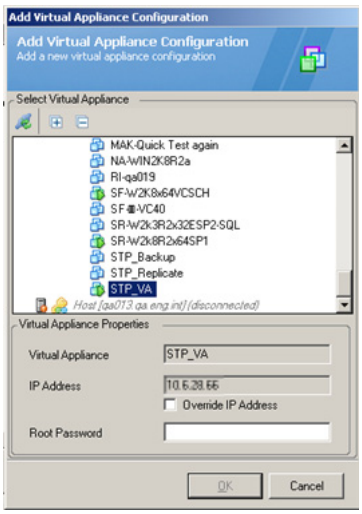
4 By default, vRanger will connect to the vRanger virtual appliance using the first IP address reported by the vSphere API. This IP will be displayed in the IP Address field. If you have only one NIC configured on the virtual appliance, continue to the next step.

or

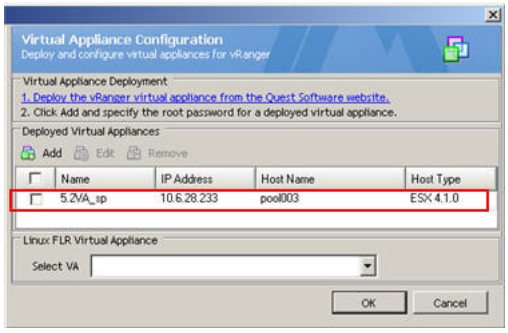
If you have more than one NIC configured on the virtual appliance, you may not want vRanger to connect to the first NIC. Using the **Override IP Address** field, you can configure vRanger with the IP address for the NIC to which vRanger should connect.

For example, if you have two NICs configured on the virtual appliance, and want vRanger to connect to the second NIC, simply select **Override IP Address**. The **IP Address** field will become enabled. Enter the IP for the second NIC in the IP Address field.

5 Enter the root password for the virtual appliance, and click **OK**.



The virtual appliance will appear in the Deployed Virtual Appliances section:



6 In the Linux FLR Virtual Appliance section, select a deployed virtual appliance to use for Linux file-level recovery.

Changing Virtual Appliance Configuration

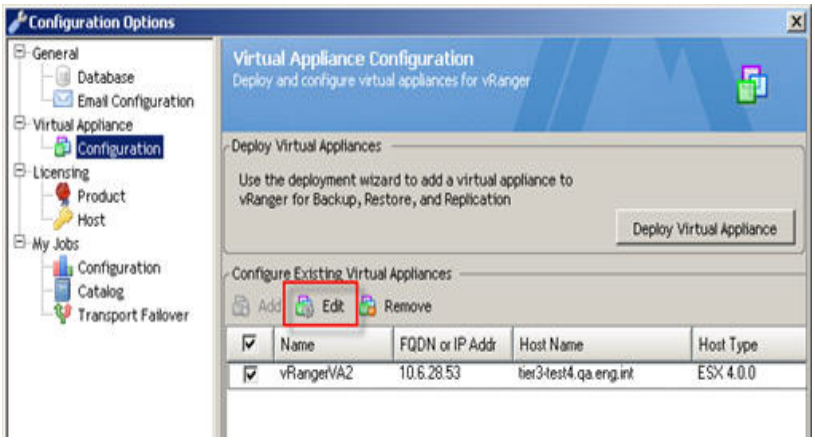
After a virtual appliance is deployed, you may change its configuration from the vRanger interface. If you need to add additional NICs or configure the VA with a static IP address, you will need to do that by logging into the virtual appliance.

Caution Reconfiguring the virtual appliance may cause it to be power cycled, possibly affecting configured jobs. You will be prompted with the jobs configured for the virtual appliance, please review these carefully to avoid service interruption.

1 From the My Inventory view, right-click on the virtual appliance you want to change, and click **Virtual Appliance Configuration**.

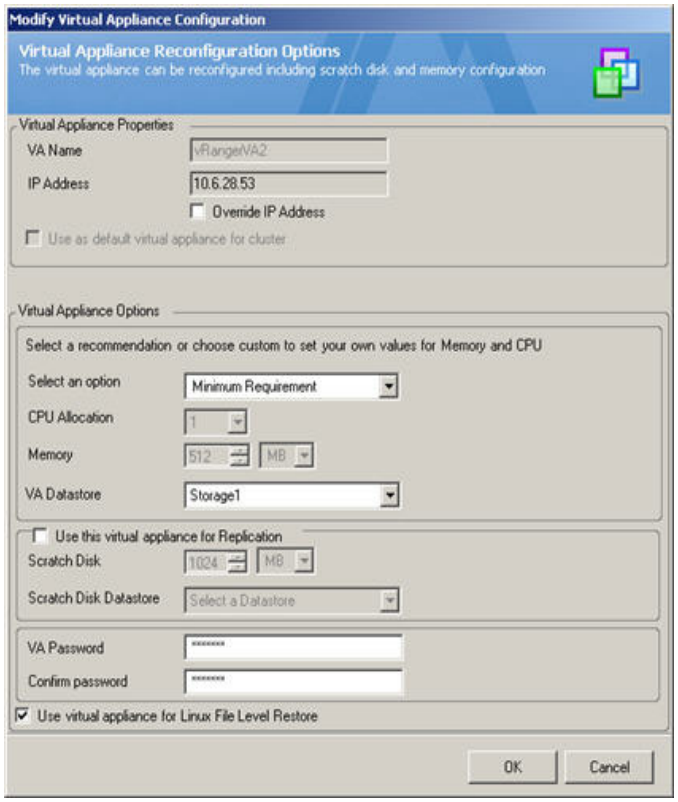
or

From the vRanger toolbar, click **Tools>Options**, then select **Virtual Appliance**. Select the virtual appliance to edit, and click **Edit**.



2 The **Modify Virtual Appliance** dialogue appears.

Click **Ok**.



1 In the Virtual Appliance Properties section, confirm the VA Name. Edit as required.

2 By default, vRanger will connect to the vRanger virtual appliance using the first IP address reported by the vSphere API. This IP will be displayed in the IP Address field. If you have only one NIC configured on the virtual appliance, continue to the next step.

   or

   If you have more than one NIC configured on the virtual appliance, you may not want vRanger to connect to the first NIC. Using the **Override IP Address** field, you can configure vRanger with the IP address for the NIC to which vRanger should connect.

   For example, if you have two NICs configured on the virtual appliance, and want vRanger to connect to the second NIC, simply select **Override IP Address**. The **IP Address** field will become enabled. Enter the IP for the second NIC in the IP Address field.

   Note See Disabling DRS for the vRanger Appliance for manual network configuration procedures.

3 In the **Virtual Appliance Option** section, change the resources allocated to the virtual appliance as needed. You may select one of two preconfigured options, or configure a custom setting.

   - **Minimum Requirement** - this setting allocates 1 CPU and 512 MB of RAM. This is usually sufficient for 2 concurrent tasks per VA.
   - **Quest Recommended** - this setting allocates 2 CPUs and 1 GB of RAM. This is sufficient for 4 concurrent tasks per VA.

- **Custom Setting** - select this value to configure the VA with higher resources for more concurrent tasks per VA.

4 To change the datastore for the VA's primary disk, select the new datastore in the **VA Datastore** field,.

5 **DO NOT** create or modify the VA Scratch Disk from this dialogue.

Caution Rebooting the VA will cause a reboot after the VA configuration change, which will prevent the VA Utility script from functioning properly.

See the following sections for more information:

- For information on creating the Scratch Disk, see Step 6: Adding Second "Scratch" Disk.
- For information on running the VA Utility script, see Step 7: Load VA Utility script.
- For information on resizing the scratch disk, see Increasing The VA Scratch Disk Size.

6 If desired, enter a new password for the virtual appliance.

7 If you wish to perform file-level recovery from backups of Linux virtual machines, you will need to configure a VA to use for Linux FLR. Select **Use virtual appliance for Linux File Level Restore**.

If you select this option for a different VM in the future, then that VM will be used for Linux FLR.

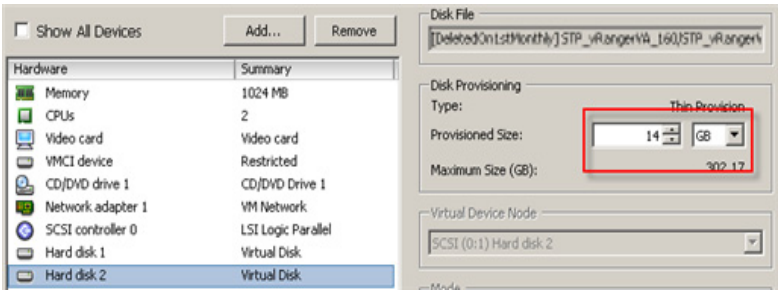
Increasing The VA Scratch Disk Size

If you increase the size of the scratch disk, you must re-run the VA utility script to take advantage of the new storage space.

To increase the VA scratch disk size:

1 In the VI Client, right-click the virtual appliance and select **Edit Settings**.

2 Select the scratch disk (usually Hard Disk 2), and configure the desired size.



3 Click OK and wait for the configuration change to take place.

Caution Do not reboot the virtual appliance

4 Login to the VA, or connect via a terminal application.

5 Using the cd command, change the working directory into the extracted directory.

```
cd /root/bin/vautil_01.02.01
```

6 Run the following command:

```
/root/bin/vautil_01.02.01/scratch_scsi.sh -g
```

The command, when successful, will display the following output:

```
RESIZED:  ${devX} ${devX_new_size_in_bytes}
```

7 Reboot the virtual appliance.

Disabling DRS for the vRanger Appliance

If the vRanger appliance is configured on hosts that are part of a DRS enabled cluster, the VA(s) may be migrated to a different host due to DRS/HA activity. This will cause replication failures as the virtual appliance is no longer in the expected location (as configured in vRanger). To prevent this disruption, it is important to disable DRS for the vRanger virtual appliances.

1 In your VI client, find the cluster(s) containing the vRanger virtual appliances.

2 Right-click on the cluster, and select **Edit Settings**.

3 In the Cluster Settings dialog, select **Virtual Machine Options.**

Search for or find the vRanger virtual appliances. Right-click on each virtual appliance, and select **manual**.

Using vRanger

This chapter describes the main these categories of the interface. It also includes instructions for using basic functionality.

This chapter contains the following sections:This section contains the following topics:

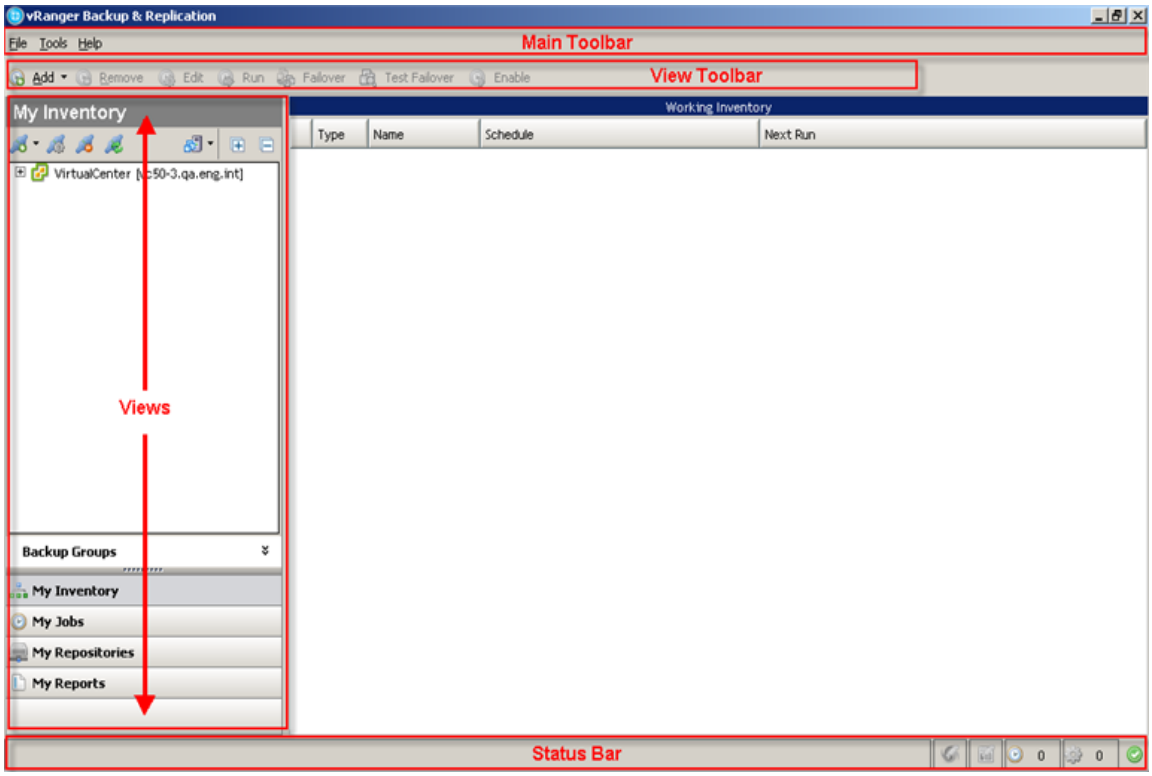
Interface Overview 65
My Inventory View 67
My Jobs View 75
My Repositories View 80
My Reports View 83

Interface Overview

The vRanger interface consists of four views: **My Inventory, My Repositories, My Jobs,** and **My Reports**. Each view features the **Main Toolbar** as well as each pane's **Toolbar**.

In the left pane within these views, data displays nested in a tree structure. In general, the information that displays in the right pane depends on the node that you select in the left pane. Each node has a distinct icon, making navigation easy and hierarchy clear.

The Startup Wizard can be opened by selecting **Tools-->Startup Wizard**. For instructions on using the Startup Wizard to configure the software for your environment, see the *Configuration* chapter.



Main Toolbar

File	The only submenu available under **File** is **Exit**. Click it to close the application.
Tools	• **Catalog Search:** Use this form to search and browse for savepoints that contain the files and folders you're looking for. Restore operations can be launched from the items in the results pane. • **Startup Wizard**: Select this command to launch the **Startup Wizard**, through which you can add a VC, host, repository; license a host; set up a mail server; and add email addresses. • **Options**: When you select this command, the **Configuration Options** dialog displays. It contains information in these categories: **General, Virtual Appliance Configuration, Licensing,** and **My Jobs**.
Help	• **Help Contents**: Select this command to view online help. • **License Information**: The **License File** dialog includes the total number of licensed CPUs, the total number of days that the license covers, and the number of days remaining under the current license. From this dialog, you can import a license file. • **About vRanger Backup & Replication**: Click this option to view version number, copyright information, a link to the website, and licensing terms. Also access the **System Information** dialog, which includes hardware resources, components, software environment, and Internet settings. The dialog features a window and submenus that allow you to view remote computers and system history. The About vRanger screen includes a **System Info** button that will collect information about the vRanger machine. You may export this information to a text to aid interactions with Quest Support.

Status Bar

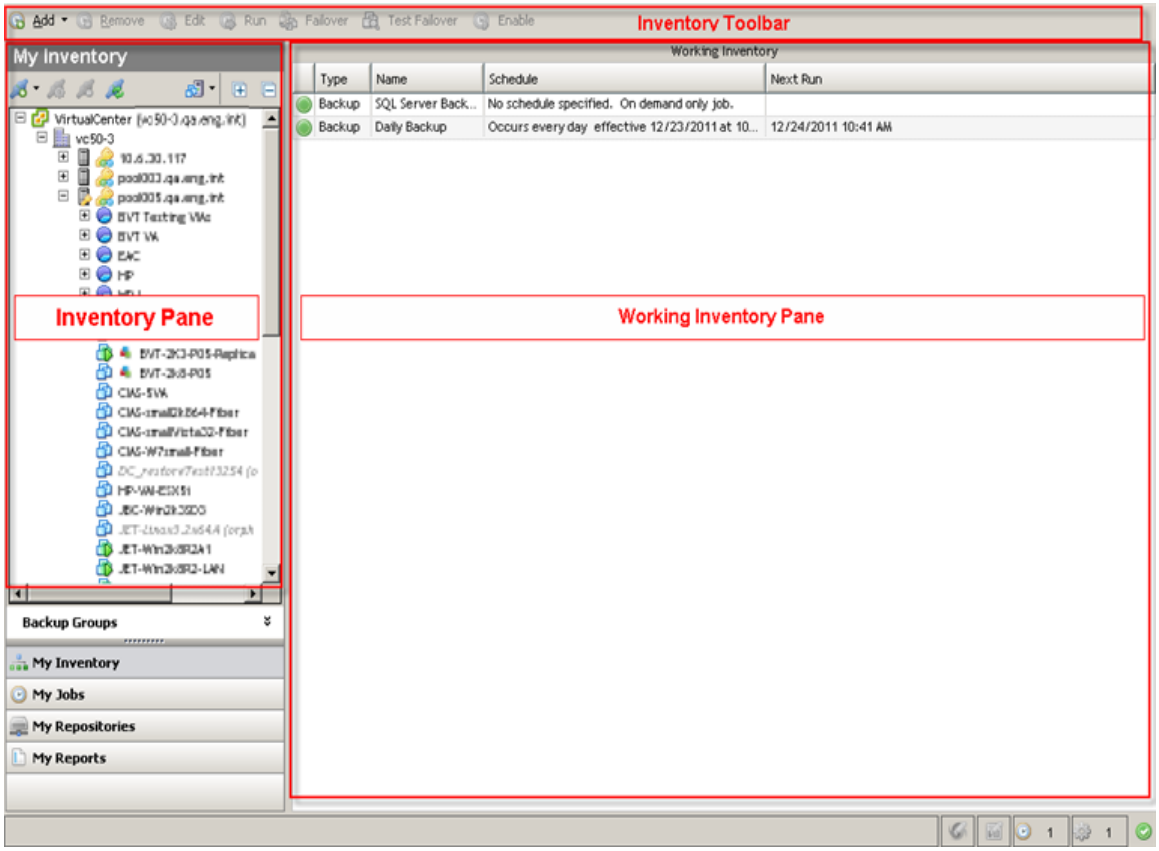
In every view, several icons display in the **Status Bar at the bottom of the vRanger UI**. Regardless of the view displayed, the **Status Bar** always gives you a glimpse of active job and task count. Using the example from the table below, the **Active Jobs Count** (4) corresponds to the first number in the **My Jobs: Current Jobs** node. The **Active Tasks Count** (2) corresponds to the second number at the same node level. That is, these counts display in this format in **My Jobs: Current Jobs** (4,2).

	**Service Support Logging**: Right-click to enable log files to assist in troubleshooting Windows vRanger service.

	**Client Support Logging**: Right-click to enable log files to assist in troubleshooting the vRanger client (GUI).
	**Active Jobs Count**: This indicator tells you the total number of active backup, restore, and replication jobs running in real-time.
	**Active Tasks Count**: This indicator tells you the total number of active backup, restore, and replication tasks running in real-time.
	**New Failed Jobs**: Indicates that there are new failed jobs.
	When you click the **New Failed Jobs** icon, this action resets the data in the **Status Bar**. In place of the new failed jobs icon, this icon displays and the active job and task count fields are reset, populating with zeroes (0) initially.

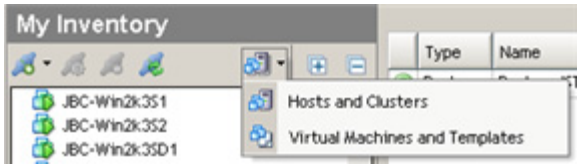
My Inventory View

The **My Inventory** view displays the virtual infrastructure, configured jobs, and job details. It consists of the Inventory pane on the left and the **Working Inventory** pane on the right. This is the view from which most backup and replication jobs are added.



My Inventory Pane

The **My Inventory** pane offers two views, depending on which value you select from the Inventory dropdown--Hosts and Clusters or VMs and Templates.



From this pane, you can navigate and refresh the tree; add or remove connections to VCs or ESX Servers; and add credentials to ESX Servers. You can also add a backup or restore job.

Inventory Pane Toolbar and Icons

The toolbar is located under the General Toolbar (File, Tools, Help). Options on this toolbar will be enabled or disabled based on which object is selected.

	**Add**: Click this command to add a backup, restore, or replication job.
	**Cancel**: To cancel a job that is running, click this command.
	**Edit**: Click this icon to open the **Backup Wizard, Restore Wizard,** or **Replication Wizard** (as appropriate) and make updates to the configuration for the job.
	**Disable**: When you click this command, the job that is selected in the **Working Inventory** pane will be disabled. A pause icon displays adjacent to the listing.
	**Enable**: When a disabled job is selected in the **Working Inventory** pane, click this command to enable the job. A run icon will display next to the listing.
	**Remove**: When you click to highlight a job or savepoint in the **Working Inventory** pane and then click this icon, the job is removed.
	**Restore**: When you click to highlight a backup job or savepoint in the **Working Inventory** pane, click this icon to open the **Restore Wizard.**
	**Run**: Click this command to run the job that is selected. This will reload the window and display the job in **My Jobs: Current Jobs**.

The icons below appear in the My Inventory Pane, and are enabled based on what you select in the **My Inventory** and **Working Inventory** panes



.

	**Add**: The action this command triggers depends upon the node that you have highlighted in the **My Inventory** pane. Click this icon to access one of these dialogs--**Add VirtualCenter Credentials**, **Add Host Credentials**, **Backup Wizard**, or **Restore Wizard**.
	**Properties**: When a VC is selected, click Properties to view the credentials used to connect to it.
	**Remove**: This command is available to delete any object that has been added to the view. **Note** If you remove an ESX host, all login and backup information is deleted. If there is a job scheduled to use this host, the job will fail.
	**Refresh**: Click this icon to refresh the My Inventory pane. Initially, only the highest-level node displays (VC). A Loading Inventory message displays while the pane refreshes.
	**Run**: Click this icon to run an associated job.
 	**Inventory**: Click one of these icons to display the **Hosts and Clusters** or **Virtual Machines and Templates** view.
	**Expand All**: This will expand the tree view. After refreshing the pane, click this icon to reload the tree structure under the VC node.
	**Collapse All**: Click this icon to collapse the tree view so that only the VC node displays in the pane.

When the **Hosts and Clusters** inventory option is selected, the icons below appear in the inventory list.

	**Folder**: Click this icon to expand the view.
	**Resource Pool**: Click this icon to expand the view

	**VirtualCenter**: Right-click this node to access these commands: • Add Credential • Credential Properties • Remove VirtualCenter • Backup VirtualCenter • Restore • View History • Refresh Inventory
	**Cluster**: Right-click this node to access these commands: • Add Credential • Credential Properties • Remove Credentials • Backup • Restore • View History • Refresh History
	**ESX(i) Host** (not credentialed): Right-click this node to access these commands: • Add Credentials • Credential Properties • Remove Credentials • Backup • Restore • Configure Change Tracking (choose any or all VMs in host) • View History • Refresh Inventory
	**ESX(i) Host** (Credentialed): Right-click the node to access these commands: • Add Credentials • Credential Properties • Remove Credentials • Backup • Restore • Configure Change Tracking (choose any or all VMs in host) • View History • Refresh Inventory
	**ESX(i) Host Add**: Click this icon to add an ESX host.
	**Host Disconnected** Right click this icon to connect to the host.
	**Host Error**: Right click on this icon to determine the error.

	**Host Maintenance Mode**: Indicates the host is in maintenance mode, and may not be available.
	**Hosts and Clusters**
	**vApp**: vApps can comprise of any applications running on any OS, and provide a mechanism for customers to move their applications between internal clouds or external clouds with still the same service levels.
	**Change Block Tracking**: This icon indicates that a job has Change Block Tracking enabled. Left click the icon to disable.
	**Virtual Machine Powered Off**: Click the icon to power on the VM.
	**Virtual Machine Powered On**: Click the icon to power off the VM.
	**Virtual Machine Suspended (Paused)**: Click the icon to start the VM.

When the **Virtual Machines and Templates** inventory option is selected, the icons below appear in the inventory list.

	**VirtualCenter**: Right-click at this node level to access these commands: • Add Credential • Credential Properties • Remove VirtualCenter • Backup VirtualCenter • Restore • View History • Refresh Inventory
	**Virtual Center Add**: Click this icon to Add a Virtual Center.
	**Cluster**: Right-click this node to access these commands: • Add Credential • Credential Properties • Remove Credentials • Backup Demo • Restore • View History • Refresh History
	**Virtual Machine**: Right-click the node to access these commands: • Add Credential • Credential Properties • Remove Credentials • Backup • Restore • Replication • Enable (Disable) Change Tracking • View History • Refresh Inventory
	**Virtual Machine Folder**: Click on this icon to display the virtual machines in this folder.
	**Virtual Machine Folder Open**: Clicking this icon will collapse the view and close the folder.
	**Virtual Machine Powered Off**: Click the icon to power on the VM.
	**Virtual Machine Powered On**: Click the icon to power off the VM.
	**Restore Virtual Machine**: Clicking this icon will start a restore job for this VM.

	**Virtual Service**: Click this icon to expand the list of services.
	**Backup Job**: Right-click the node to access these commands: • Add Credential • Credential Properties • Remove Credentials • Backup • Restore • Replicate • Enable Change Tracking • View History • Refresh Inventory
	**Running Backup Job:** Right-click the node to access these commands: • Add Credential • Credential Properties • Remove Credentials • Backup • Restore • Replicate • Enable Change Tracking • View History • Refresh Inventory
	**Backup Virtual Machine**: Right-click on the desired VM, then click this icon to start a backup job for this VM.
	**Replicate Virtual Machine:** Right-click on the desired VM, then click this icon to start a replication job for this VM.
	**Change Block Tracking**: This icon indicates that a job has Change Block Tracking enabled. Left click the icon to disable.
	**Connected**: This icon shows that the node is connected.
	**Connecting**: This icon indicates that a node is in the process or connecting.
	**Credentialed**: This icon indicates that VM or Host is Credentialed.
	**Data Center**: Selecting this icon will expand the data center and display all objects in it.
	**Disconnected**: This icon indicates the Node is disconnected.
	**Error**: This icon indicates that an error has occurred, or that a field is lacking required data.

	**Hierarchy**: This icon indicates the organization of the inventory displayed.
	Inventory Tree Wait: This icon is displayed while vRanger searches for all the components in the inventory.
	**Credentialed/Licensed**: This icon shows that the VM or host is licensed.
	**Invalid Licensed Credentials**: This icon shows that the credentials are invalid.
	**Resource**: This icon indicates the presence of a resource in the inventory.
	**Save**: Clicking this icon will save any changes made.
	**Server**: This icon indicates a server in the inventory.
	**Virtual Machine Template**: copy and paste this template as needed to reproduce VMs.
	**vApp**: vApps can comprise of any applications running on any OS, and provide a mechanism for customers to move their applications between internal clouds or external clouds with still the same service levels.
	**BackupReplication License:** Indicates the VC is licensed for both backup AND replication.
	**Backup** or **Replication License:** Indicates the VC is licensed for either backup OR replication.

Working Inventory Pane

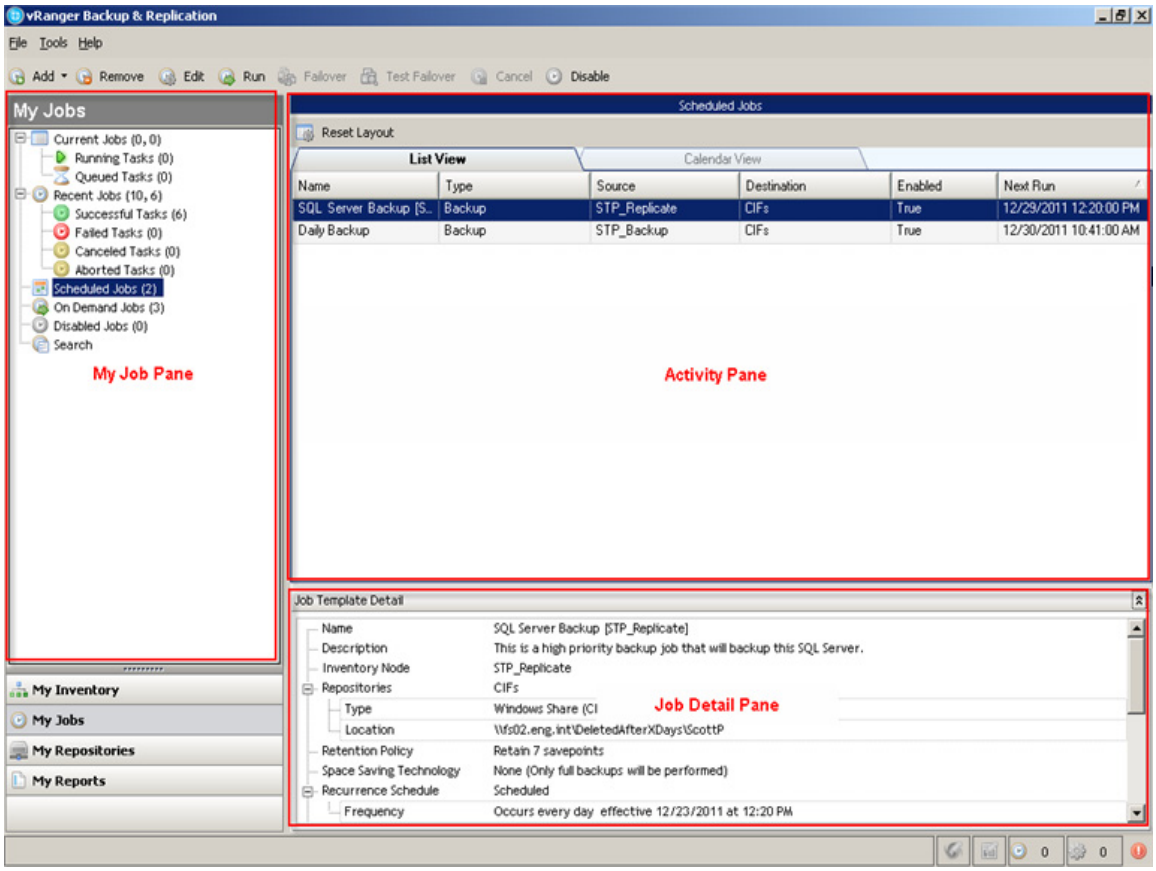
The **Working Inventory** pane lists any configured jobs for the node selected in the Inventory pane.

The data presented here--including Type, Name, and Schedule--can be sorted by dragging a column heading. When you click, an arrow will display in the column heading, indicating the sort order (ascending or descending). Click the column heading again to change the sort order. The same data can be grouped by dragging a column heading to the target area above all of the headings. Doing this will group current jobs by that data type.

If you click to highlight a job in the Working Inventory pane and then click the Run icon, the job is moved to the **My Jobs: Scheduled Jobs**. From this location, you can start the job. Once the backup has been completed, it will be moved to **My Jobs: Recent Jobs**.

My Jobs View

The My Jobs view consists of the **My Jobs** pane on the left, the **Activity** pane on the upper right, and the **Job Detail** pane on the lower right. Use this view to add, run, schedule, disable, or cancel a backup job. From this view you must have either the Scheduled Jobs or On Demand Jobs node (in the My Jobs pane) highlighted to add a job.



A job can be made up of many tasks. Even if some of its tasks fail, the job itself can complete successfully. In this view, you can also confirm configuration settings for a job. **My Jobs** makes monitoring jobs and tasks easy. Use the **Job Detail** pane to track job progress. To view task progress, see the **Task Detail** area. Use the tool to view job history and check status.

My Jobs Pane

The **My Jobs** pane displays task and job information for each of the following statuses: Current; Recent; Scheduled; On Demand; and Disabled. For Current and Recent jobs, associated tasks are listed in separate categories. In these nodes, jobs and tasks are distinguished in this format--(x, y)--where *x* is the number of jobs and *y* refers to the number of tasks that make up the job. The numbers in the **Current Jobs** node corresponds to the values in the **Status Bar's Active Jobs Count** and **Active Tasks Count fields**, respectively.

My Jobs Pane Toolbar and Icons

	**Add**: Click this command to add a backup, restore, or replication job.
	**Remove**: Use this icon to delete a job highlighted in the **Activity** pane.
	**Edit**: With a job selected, click the this icon to access the **Backup Wizard, Restore Wizard,** or **Replication Wizard**.
	**Enable**: When a disabled job is selected in the **Working Inventory** pane, click this command to enable the job. A run icon will display next to the listing.
	**Disable**: When you click this command, the job that is selected in the Working Inventory pane will be disabled. A pause icon displays adjacent to the listing.
	**Run**: Click this command to run the job that is selected. This will reload the window and display the job in **My Jobs: Current Jobs**.
	**Cancel**: To cancel a job that is running, click this command.
	**Current Jobs:** Click this icon to display Current Jobs. Two levels can be displayed: • Running Tasks • Queued Tasks

	**Running Tasks**: Refers to the number of active tasks associated with current jobs.
	**Queued Tasks**: For all current jobs, this number indicates the number of tasks waiting to run.

	**Recent Jobs**: This node lists the number of recent jobs and tasks, respectively, that have been run. The format (x, y) is used, indicating *x* recent jobs and *y* recent tasks. Under this node, there are several objects: • Successful Tasks • Failed Tasks • Canceled Tasks • Aborted Tasks
	**Successful Tasks**: This category indicates the number of tasks that have completed successfully for recent jobs.
	**Failed Tasks**: The number in this category notes the failed tasks associated with recent jobs.
	**Canceled Tasks** or **Aborted Tasks**: These tasks use the same icon but refer to different statuses for recent jobs. A canceled task results from a user action. An aborted task refers to a change initiated by the system.
	**Scheduled Jobs**: When this node is selected, the **Activity** pane is divided into two tabs--**List View** and **Calendar View**. The jobs and tasks listed on both views are those that have been scheduled to run. The **Job Template Detail** area features configuration options selected for each job.
	**On Demand Jobs**: These jobs are configured to run at any time; that is, they are not scheduled. The **Job Template Detail** area features configuration settings for each job.
	**Disabled Jobs**: Listed here are jobs that have been configured, have been scheduled and perhaps even run, but are currently disabled. From this view, any disabled job can be enabled.
	Selecting this node displays **Job** and **Results** areas in the **Activity** pane. A **Job Template Detail** area displays as well. Enter criteria in the Job area and view the data returned in the Results area. If you click a listing in the results, you can view additional detail about it in the **Job Template Detail** area.
	**LAN Failover:** If the Failover option (along with the Fiber Backups option) was selected, this icon indicates the Fiber connect failed, and the job failed over to a LAN connection.

Activity Pane

- The data that displays in the Activity pane depends upon the category that you select in the **My Jobs** pane. You can expand a job listing to display data on tasks associated with that job.

For the **Recent Jobs, On Demand Jobs,** and **Disabled Jobs,** the **Activity** pane features data organized in these categories:

- **Name**--This is the name of the job.
- **Type**--This refers to job type.
- **Source**--Indicates the host or cluster in which the source VM resides.
- **Destination**--For backup jobs, this indicates the repository selected in the backup job. For restore and replication jobs, this indicates the target host.
- **Status**--Lists the status of the backup or restore job or task: Completed, Failed, Aborted, or None.
- **Start Time**--Lists when the job or task was initiated.
- **Duration**--The length of time that the job or task took to complete.
- **Message**--In this column, information about a task or job displays. For example, "Host is not licensed" or "Please check host credentials."

If you select the **Current Jobs** node, all of the categories listed above will display in the **Activity** pane. In addition, data for these categories will be included: T**hroughput** and **Progress**.

In the interface, you can clearly distinguish the running job (parent) and the running task (child). To view details for either job or task, click to highlight the object. Either the **Job Detail** or **Task Detail** area will display at the bottom of the **Activity** pane.

You can sort the columns in the **Activity** pane by clicking them. When you click, an arrow will display in the column heading, indicating the sort order (ascending or descending). Click the column heading again to change the sort order.

If you select any sub-category under the **Recent Jobs** node and then right-click on a job or task in the **Activity** pane, a **Go To** menu will display. The available commands list depends upon the activities previously associated with that job or task. For example:

- **Go To Repository Savepoint**
- **Go To Scheduled Activity**
- **Go To Disabled Activity**

If you select a category under **Current Jobs**, **Scheduled Jobs**, or **On Demand Jobs** and then right-click on a job or task in the **Activity** pane, these navigation options will display:

- **Go To Inventory View**
- **Find Previous Jobs**

Go To Current Jobs

For **Scheduled Jobs,** the **Activity** pane features two tabbed areas--**List View** and **Calendar View. List View** displays scheduled jobs by data type--N**ame, Type, Source, Destination, Enabled,** and **Next Run.** The **Calendar View** offers **monthly, weekly, daily options** for viewing scheduled jobs. Below this, the **Job Template Detai**l area features basic configuration information about the job selected in the **List View**.

Click the node in the **My Jobs** pane to display **Job** and **Results** areas in the **Activity** pane. After the results display, you can click to highlight any entry and additional information about the job will display in the **Job Template Detail** area below the results, which you can save.

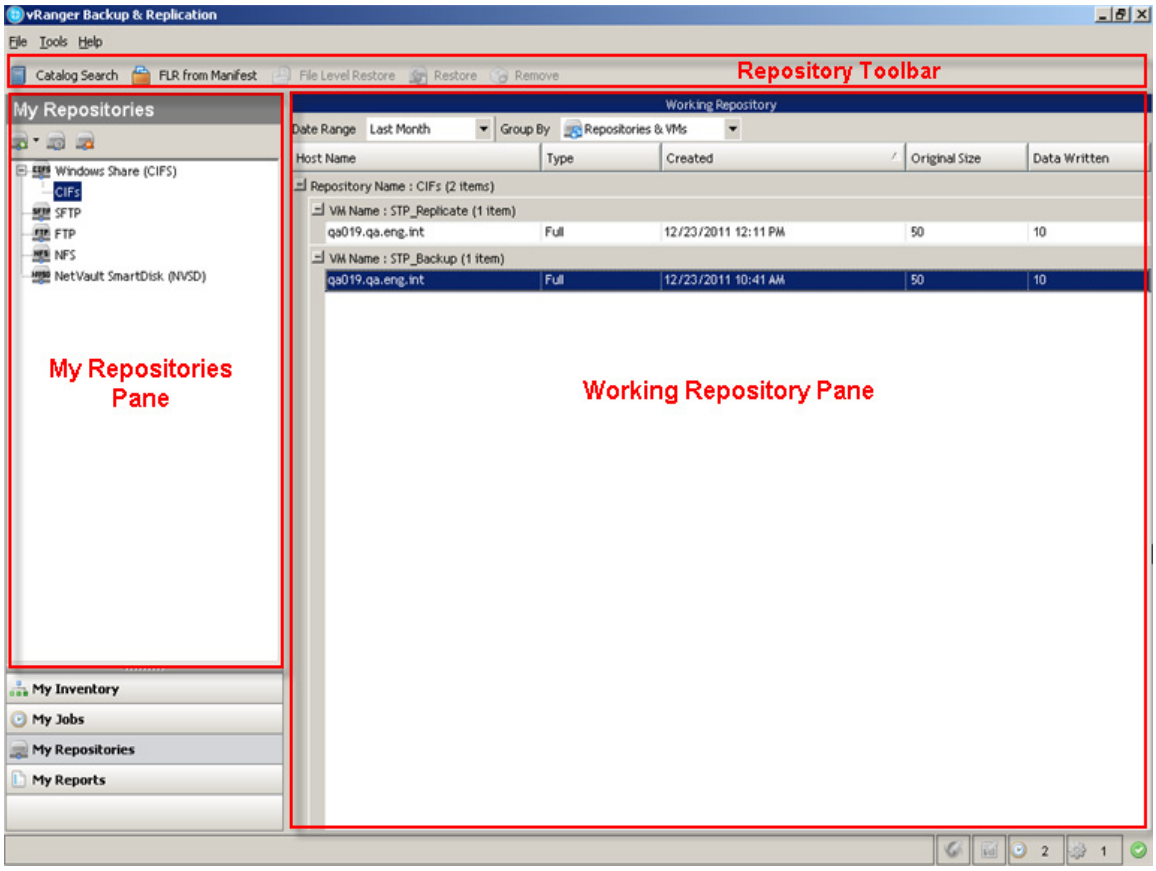
Type	• Historic • On Demand • Scheduled • Disabled
Task Status	• All • Success • Failed • Aborted • Canceled
Repository	The values included in this field depend on the repositories that were added previously.
Job Type	• All • Backup • Restore • Replication
VM Name	To search by VM name, populate this field.
Job Status	• All • Success • Failed • Aborted • Canceled
To Date	The default value is (none). When you click the dropdown, a calendar view displays.
Host or VC	The default value for this field is (All). This field is populated with the hosts and VCs that were added previously.

Job Detail Area

The **Job Detail** area contains job configuration data as well as the progress and status of currently running jobs and tasks. This area features two tabs--**Detail** and **Log**. For example, if you select **Recent Jobs** in the **My Jobs** pane, the **Job Detail** area will display this type of data on the **Detail** tab. The data in the corresponding **Log** tab will indicate which VMs were retrieved, which disks were skipped, and which tasks were completed. It will also indicate that the job was saved.

My Repositories View

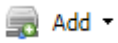
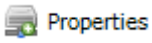
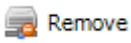
This view displays the configured repositories and available savepoints. This is the view from which most restore jobs are configured. It consists of the **My Repositories** pane on the left and the **Working Repository** pane on the right.



My Repositories Pane

This pane displays the configured repositories in a tree structure, grouped by repository type. Repositories can be added and removed through the pane toolbar.

My Repositories Pane Toolbar and Icons

	**Catalog**: After selecting a repository, click this icon to search (or browse) for a specific Savepoint. **Note** Cataloging must have been enabled at the time the backup job ran.
	**File-Level Restore**: When you want to restore an individual file from a savepoint, click this icon to access the **File Level Recovery** dialog.
	**FLR From Manifest**: When you wish to restore an individual file from the Manifest, click this icon.
	**Restore**: When you click to highlight a backup job or savepoint in the **Working Repository** pane, click this icon to open the **Restore Wizard**.
	**Remove**: To delete a savepoint that is selected in the **Working Repository** pane, click this command.
	**Add**: Use this command to add either a CIFS or SFTP repository.
	**Properties**: This icon is the gateway to the connection information and repository credentials used to connect to the share. You can update credentials at any time.
	**Remove**: Click this icon to open the **Repository Removal Confirmation** dialog. Before removing a repository, confirm that it is not needed to store current jobs.

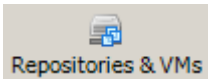
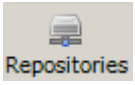
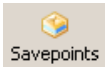
Working Repository Pane

The **Working Repository** pane shows the configured repositories and the savepoints stored there. The data displayed in this pane can be grouped and sorted using the icons in the **Group By** dropdown. The data in this pane is organized under these columns:

- Repository Name
- VM Name
- Host Name
- Type (Full, Differential, Incremental)
- Start Time
- Original Size (MB)
- Stored Size (MB)

Grouping

There are two methods for grouping savepoints. You can select an option from the Group By toolbar, or you can create customized groupings using the GroupBox command.

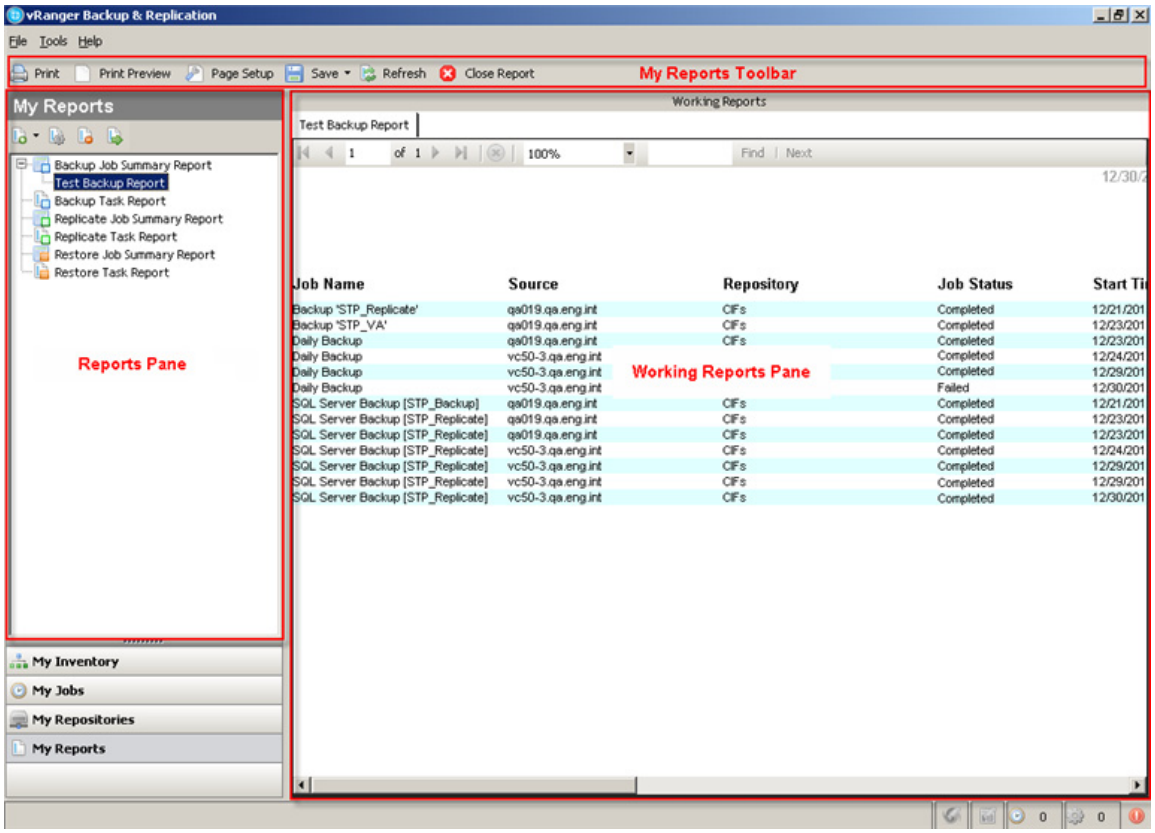
	**Repositories & VMs**: Groups the savepoints first by repository, then by VM.
	**Repositories**: Click this icon to group savepoints by repository.
	**VMs**: This command groups the savepoints by VM.
	**Savepoints**: This groups the savepoints first by repository, then VM, then by root savepoint ID.
	**Hosts**: This will group the savepoints by host.
	**Date Range**: This dropdown allows you to select a date range for the jobs listed in the Working Repository pane. The values include: • All • Last Day • Last 3 Days • Last Week • Last 2 Weeks • Last Month • Last 3 Months • Custom • Define Custom. If you select Define Custom, the Savepoints Date Ranger dialog will display. Use it to set a date range for the jobs you want to view.

Sorting

The columns in the **Working Repository** pane can be re-ordered by dragging a heading and clicking the column. When you click, an arrow will display in the column heading, indicating the sort order (ascending or descending). Click the column heading again to change the sort order.

My Reports View

From the **My Reports** view you can create and manage reports as well as email them. The left-hand **Reports Pane** displays a list of all reports for backups and restores. The right-hand pane, called **Working Reports**, displays the report itself.



My Reports Toolbar

The toolbar is located under the General Toolbar (File, Tools, Help). The options on this menu are only enabled if a report is showing in the Working Reports Pane, and act only on that report

.Reports Pane

	**Print:** Use this icon to print the selected report.
	**Print Preview:** Use this icon to preview the report before printing.
	**Page Setup:** Use this icon to configure the page settings used to print your report.
	Save: Use this icon to save your report as a PDF or Excel file.
	**Refresh:** Use this icon to refresh the selected report.
	**Close**: Use this icon to close the selected report.

The Reports Pane includes nodes for the types of reports listed below.

- Backup Job Summary
- Backup Task Summary
- Backup Task
- Replicate Job Summary
- Replicate Task
- Restore Job Summary
- Restore Task

Simply right click on the node for a desired report type, and select Add <report type>, You may also create a new report or work with existing reports via the Reports Pane toolbar icons.

Reports Pane Toolbar

	**Add**: Use this icon to start the process of creating a backup or restore report.
	**Edit**: Click this icon to open the **Report Wizard** to make changes to the selected report's configuration. Enabled only when a job is selected in My Jobs.
	**Remove**: Click this icon to delete the selected report. Enabled only when a job is selected in **My Jobs**.
	**Run**: Click this icon to run the selected report. Enabled only when a job is selected in **My Jobs**.

Report Wizard

Use the **Report Wizard** to configure a report about backup, restore, or replication jobs or tasks. After naming and describing the report, you select the VMs to include. Then, you identify the contents of the report.

You then select options for filtering data based on date range and status. You can assign styles for headers, footers, and body text. Finally, you assign a schedule for generating the report and you select settings for the email notification sent when a report is created. A summary screen will display, listing the options that you have selected for the report. After reviewing the summary, the configuration is complete.

The report listing will display in the **My Reports** pane. The text of the reports displays in **Working Reports** pane.

Backup

This chapter includes general information about backup functionality as well as instructions for adding a job through the Backup Wizard and My Jobs view.

This chapter contains the following sections:This section contains the following topics:

Backup Overview 87
Backup Transport Options 87
Transport Selection Order 89
Configuring A Backup Job 91

Backup Overview

vRanger uses VMware snapshot technology to temporarily store incoming write requests while the source VM is being backed up. After the backup completes, the snapshot is deleted, which commits those pending writes to disk. vRanger can backup a VM that already has an open snapshot and will backup the open snapshot, but any secondary consolidated helper snapshots will be closed prior to running the backup.

With Inventory Node Selection, you can browse the VirtualCenter (vCenter) inventory and select which VMs or groups (folders) you wish to backup. You can select a VM, ESX host, folder, resource pool, datacenter, or vCenter and backup all of the VMs located under that node in the tree.

Backup Limitations

vRanger cannot backup physical RDMs (raw device map) partitions. You should see the warning: “incompatible drive detected”.

Backup Transport Options

vRanger offers different data transport options for backup tasks. When configuring a backup job, vRanger will automatically choose the fastest option available, but you can also override this selection by configuring a custom transport selection. vRanger will choose from the following transport options, based on your configuration.

HotAdd

vRanger includes support for VMware's HotAdd disk transport functionality which allows LAN-Free backups from a virtual machine. vRanger can use HotAdd for two backup methods:

- **VA-based HotAdd** - will mount the source VM’s disk to the vRanger virtual appliance deployed on the source host (or cluster). This allows vRanger to have direct access to the VM data through VMware’s I/O stack rather than the network.
- **Machine based HotAdd** - if vRanger is installed in a virtual machine, this method will mount the source VM’s disk to the vRanger virtual machine. This allows vRanger to have direct access to the VM data through VMware’s I/O stack rather than the network.With this method, the backup processing activity occurs on the vRanger server.

In order to use vRanger with HotAdd, the virtual machine (either the vRanger VA, or the vRanger virtual machine) must be able to access the target VM's datastore(s). In addition, all hosts that the vRanger VM or VA could be vMotioned to must be able to see the storage for all VMs that vRanger will be configured to back up.

Note When using HotAdd, plan on 1 concurrent backup per CPU core. To calculate the maximum number of concurrent backup tasks per proxy server, simply identify the number of CPU cores on that server - that is the maximum number of concurrent backups. For example, a Dual-Socket, Quad-Core system will be able to perform up to 8 concurrent backup jobs.

Configuring the vRanger machine for HotAdd

When using HotAdd, make sure to disable automount on the vRanger machine. This will prevent Windows on the vRanger VM from assigning a drive letter to the target VMDK. To disable automount:

- From the start menu, select "run" and enter `diskpart`.
- Run the `automount disable` command to disable automatic drive letter assignment.
- Run the `automount scrub` command to clean any registry entries pertaining to previously mounted volumes.

HotAdd Limitations

The use of HotAdd for backups is subject to the following limitations:

- **VMFS Block Size** - To use HotAdd, the VMFS containing the source VM and the VMFS containing the vRanger VM must use the same block size. For example, if you back up a virtual disk on a datastore with 1MB blocks, the proxy must also be on a datastore with 1MB blocks. For more information, refer to VMware's *VDDK 1.2.1 Release Notes*: *http://www.vmware.com/support/developer/vddk/VDDK-1.2.1-Relnotes.html#knownissues*
- **HotAdd and the vRanger VM** - You cannot use HotAdd to backup the vRanger virtual machine. This may result in the vRanger volume being removed from the VM along with the source VMDK. For more information, refer to VMware's *VDDK 1.2.1 Release Notes*: *http://www.vmware.com/support/developer/vddk/VDDK-1.2.1-Relnotes.html#knownissues*
- IDE disks are not supported.
- Mixed mode clusters are not supported for HotAdd. For example, a VM on an ESXi 4.0 host cannot HotAdd a disk from an ESX 3.5 host.

SAN Backups

The LAN-Free (SAN) configuration is identical whether ESX or ESXi is used, providing the best mix of performance and compatibility for protecting your data, especially if your environment has a mix of VMware hypervisors.

In order to perform SAN backups, vRanger must be installed on a physical system attached to your SAN environment. This is a high performance configuration that requires vRanger to be connected to your fibre or iSCSI network. In addition, the VMFS volumes containing the VMs to be protected must also be properly zoned/mapped to the vRanger proxy server.

Configuring vRanger for SAN Backups

With vRanger installed on a physical proxy server, the following configurations must be made:

- Disable automount on the vRanger machine:
  - From the start menu, select "run" and enter `diskpart`.
  - Run the `automount disable` command to disable automatic drive letter assignment.
- Run the **`automount scrub`** command to clean any registry entries pertaining to previously mounted volumes.

- On your storage device, zone your LUNs so that the vRanger HBA (or iSCSI initiator) can see and read them.
- Only one proxy should see a set of VMFS LUN's at one time. For backups only, The proxy server should only have **read-only** access to the LUNs. In order to perform LAN-Free restores, please ensure that the vRanger server has Read + Write access to any zoned VMFS LUNs to which you wish to restore.

LAN

If no LAN-Free option is available, backups can be processed over the network. LAN backups are supported in all configurations and require no additional set-up to work, but are generally the slowest option. Network backups can be processed two ways, depending on the source host configuration.

- **VA-based LAN** - will transfer the source VM's data from the source disk to the vRanger virtual appliance over the network. With this method, the backup processing activity occurs on the vRanger virtual appliance.
- **Machine-based LAN** - If there is no vRanger VA deployed, vRanger will transfer the source VM's data from the source disk to the vRanger server over the network. With this method, the backup processing activity occurs on the vRanger server. For ESXi servers (which do not have the service console), data will be sent via VMware's VDDK transport. For ESX servers that are credentialed at the host level, Machine-based LAN backups will be processed in the host Service Console.

Note Generally speaking, network backups and restores will be slower than LAN-Free. If your environment supports LAN-Free backups, use that option whenever possible.

Transport Selection Order

The vRanger Backup Wizard includes an **Automatic** option which allows vRanger to select the best available method for your configuration. When determining the best transport, vRanger will use two key criteria:

- whether virtual appliances are deployed
- where vRanger is installed

The logic used to select a transport differs slightly based on whether vRanger is installed in a VM or on a physical server. The sections below describe the steps taken to select an automatic transport.

vRanger - Virtual Machine

When installed in a virtual machine, vRanger will check the possible transports in the order shown below, selecting the first option available. The preferred transports are checked first, ensuring the best performance possible.

1. **Virtual Appliance on source host** - vRanger will first check the source host for a virtual appliance. If available, that VA will be used.
   a. **Is VA-based HotAdd available?** - vRanger will check the local VA for HotAdd support. If the host is not properly licensed, or the VA cannot access the storage for the source VM, HotAdd will not be available.

      Important If a virtual appliance is configured and HotAdd is not available, a network backup will be performed from the virtual appliance. No further transport checks will be performed.
2. **Virtual Appliance on source cluster** - if the source host does not have a virtual appliance, vRanger will check if the host is a member of a cluster, If so, vRanger will check for a shared virtual appliance on that cluster.
   a. **Is VA-based HotAdd available?** -vRanger will check the cluster VA for HotAdd support. If the host is not properly licensed, or the VA cannot access the storage for the source VM, HotAdd will not be available.

      Important If a virtual appliance is configured and HotAdd is not available, a network backup will be performed from the virtual appliance. No further transport checks will be performed.
3. **Machine-based-HotAdd** - If there is no virtual appliance configured, vRanger will perform a check to see if HotAdd is available for the vRanger VM. If the vRanger VM is on a host which is not properly licensed, or if the VM cannot access the storage for the source VM, HotAdd will not be available.
4. **Machine-based-LAN** - If a virtual appliance is not available, and HotAdd is not supported on the vRanger VM, a network backup will be performed from the vRanger machine.

vRanger - Physical Machine

When installed on a physical machine, vRanger will check the possible transports in the order shown below, selecting the first option available. The preferred transports are checked first, ensuring the best performance possible.

1. **Virtual Appliance on source host** - vRanger will first check the source host for a virtual appliance. If available, that VA will be used.
   a. **Is HotAdd available?** - vRanger will check the local VA for HotAdd support. If the host is not properly licensed, or the VA cannot access the storage for the source VM, HotAdd will not be available.

      Important If a virtual appliance is configured and HotAdd is not available, a network backup will be performed from the virtual appliance. No further transport checks will be performed.
2. **Virtual Appliance on source cluster** - if the source host does not have a virtual appliance, vRanger will check if the host is a member of a cluster, If so, vRanger will check for a shared virtual appliance on that cluster.
   a. **Is HotAdd available?** -vRanger will check the cluster VA for HotAdd support. If the host is not properly licensed, or the VA cannot access the storage for the source VM, HotAdd will not be available.

      Important If a virtual appliance is configured and HotAdd is not available, a network backup will be performed from the virtual appliance. No further transport checks will be performed.
3. **Machine-based SAN** - If there is no virtual appliance configured, vRanger will perform a check to see the vRanger server is configured for SAN backups.

4 **Machine-based LAN** - If a virtual appliance is not available, and SAN backups are not supported on the vRanger server, a network backup will be performed from the vRanger machine.

Configuring A Backup Job

vRanger uses a simple wizard to guide you through the process of creating a backup job. At a high level, the process includes the following steps. Each of the steps below is explained in more detail in the sections that follow.

- Step 1: Selecting An Inventory Node
- Step 2: Exclusions and Inclusions
- Step 3: Repository Selection
- Step 4: Transport Selection
- Step 5: Configure Backup Options
- Step 6: Retention Policies & Space Saving Technologies
- Step 7: Recurrence Schedule Selection
- Step 8: Email Notification Selection
- Step 9: Summary

Step 1: Optional Configurations

vRanger includes some optional configurations that you may want to make before created a backup job for a given object.

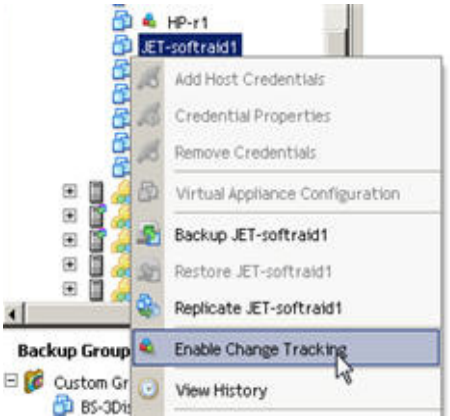
Enabling Changed Block Tracking (CBT)

Changed Block Tracking reduces the time needed for incremental and differential backups by only backing up the portions of a disk that have changed since the last backup. By determining which blocks changed within the VMDK file, vRanger only backs up the portions of a disk that have changed since the last backup. This often results in shorter backup operations, and reduced resource consumption on network and storage elements.

vSphere supports CBT, and most VMs running in this environment can utilize it. The VMs must be Hardware Version 7, and have been created and hosted in ESX4 or 4i hosts. VMs that are created in ESX3.5 or below must be migrated to Hardware Version 7 for CBT to be supported. CBT must be enabled for each VM with which CBT is to be used.

To enable or disable CBT:

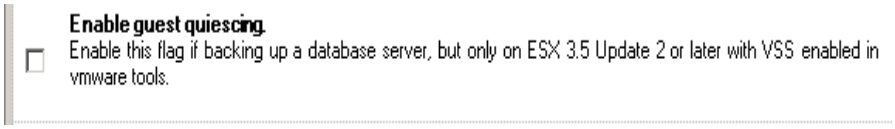
1 On the **My Inventory Screen,** right click a VM -- you can then **Enable** or **Disable Tracking** as appropriate..



2 The **Tracking Icon**  appears next to any VM on which CBT is enabled.

Determining Application Consistency

vRanger, by default, provides no quiescing during backups. When enabled, quiescing in vRanger is provided by leveraging VMware Tools installed in the VM.



This can provide three different levels of backup consistency, as described below:

Crash-consistent - A crash consistent backup is analogous to pulling the plug on a server and then backing up the data. The state of the data that is being backed up with respect to the users of the data is indeterminate. Restoring a crash-consistent image is equivalent to rebooting a server after a hard shut-down.

File System Consistent - File system consistency is achieved through standard quiescing (via the VMware Sync Driver.) which ensures that no file system writes are pending when the snapshot is taken. For normal VMs, file-system consistency is adequate, although it can cause corruption in database applications.

Application consistent - Consistency of VSS compatible applications is achieved by freezing application I/O just prior to creating the VM snapshots. This ensures that all application writes requests in the machines memory are committed to disk before the snapshot is taken.

The level of consistency provided by the **Enable Guest Quiescing** option is dependent upon the version of VMware ESX(i) (and the corresponding VMware tools) and the guest operating system. The tables below provide more detail on what is needed to achieve various levels of consistency:

File-level Quiescing		
ESX Version	**Windows Server 2003**	**Windows Server 2008 (includes R2)**
ESX(i) 4.0	VMware VSS	VMware VSS
ESX(i) 4.1	VMware VSS	VMware VSS
Application-level Quiescing		
ESX Version	**Windows Server 2003**	**Windows Server 2008 (includes R2)**
ESX(i) 4.0	VMware VSS	vzShadow.exe
ESX(i) 4.1	VMware VSS	vzShadow.exe

As shown in the table above, application consistency is not always available with the basic quiescing options. In these situations, you may use vRanger's VSS Tools (vzShadow.exe) for application-level consistency.

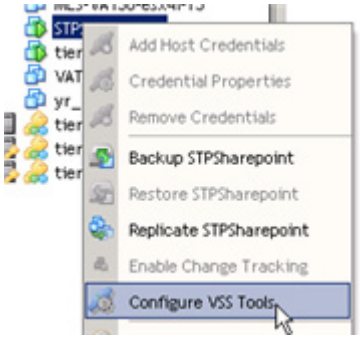
Configuring VSS Tools - Automated Method

In addition to the standard quiescing implementation using VMware Tools, vRanger provides an optional method for application quiescing, which uses a small executable tool (*vzShadow.exe*) deployed to the source VM and activated by VMware Tools during quiescing. vRanger's VSS Tools must be deployed to each VM for which you need application-consistent backups.

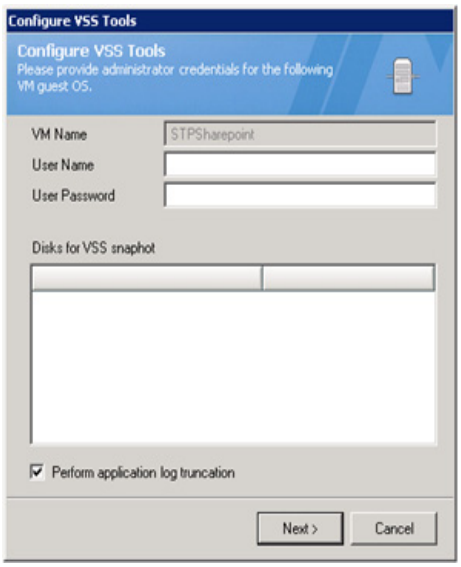
Note This method is available only for powered- on VMs running WIndows 2008 (or newer). For VMs running older Windows operating systems, you will need to install *vzShadow.exe* manually, as described in Configuring VSS Tools - Manual Method.

To configure VSS Tools:

1 When viewing the My Inventory View, right-click the desired VM and select **Configure VSS Tools**.



2 The **Configure VSS Tools** dialog appears.Enter the name and password for an account with administrative privileges on the VM.**Click Next**.



3 If the selected VM has more than one disk, the available disks will be shown in the **Disk for VSS snapshot section**. Select the disks to quiesce.

4 If desired, select **Perform application log truncation**.

5 Click **Configure**.

6 Click **Next**.

vRanger will perform the following actions:

- create the directory *C:\Program Files\quest\vzShadow*
  - populate the directory with the *vzShadow* executable and supporting files
- Create the directory *C:\Program Files\VMware\VMware Tools\backupscripts.d*
  - create a freeze.bat file in that directory that contains the appropriate contents based on your selections.

Removing VSS Tools

To remove the vRanger VSS Tools and batch file from the source VM, perform the following actions.

1. When viewing the My Inventory View, right-click the desired VM and select **Configure VSS Tools**.
2. The **Configure VSS Tools** dialog appears.Enter the name and password for an account with administrative privileges on the VM.**Click Next**.
3. Click **Remove**. The files will be removed from the VM, but the directories will remain.

Configuring VSS Tools - Manual Method

For Windows 2003 VMs, the vRanger VSS Tools must be manually deployed to the target VM, using the procedures below.

Note *vzShadow.exe* works in conjunction with VMware Tools' VSS function. VMware Tools VSS must be enabled when using *vzShadow.exe*.

1. On the vRanger machine, browse to the "Client\VSS\VSSfor2k3" sub-directory in the vRanger installation location. If the default location is accepted during the installation, the path is: *C:\Program Files\Quest Software\vRanger\Client\VSS\VSSfor2k3*
2. In this directory, find the appropriate C++ Redistributable package and *vzShadow.exe* for the bit level of the source VM (the one you wish to backup).

   For 64-bit (x64) machines, choose

   - *vcredist_x64.exe (Microsoft C++ Redistributable)*
   - *vzshadow_x64.exe*

   For 32-bit (x86) machines, , choose

   - *vcredist_x86.exe*
   - *vzshadow_x86.exe*
3. On the source VM, launch the *vzcredist* file to install the C++ Redistributable.
4. Depending on your system, move the appropriate file to the machine's root directory (typically the C:\ directory).

   For 64-bit OS, copy *vzshadow_x64.exe*

   or

   For 32-bit OS, copy *vzshadow_x86.exe*
5. Create a directory on the VM:

   C:\Program Files\VMware\VMware Tools\backupscripts.d
6. In this directory, create a batch file called `freeze.bat`.

   A sample batch file is shown below;

   ```
   c:\vzshadow.exe x:
   ```

   where X equals the drive that hosts the application -- if there are multiple volumes on this VM, list all volumes, separated by colon space (:), as in:

   ```
   c:\vzshadow.exe c: d: e:
   ```

Note When using vRanger with Quest Recovery Manager for Exchange 5.1 (or later), use the text below for your batch file. This will create an XML file that RME uses to identify vRanger savepoints with Exchange databases.

- `c:\vzshadow_64.exe-bc=x:\backup.xml x:`
- `c:\vzshadow_86.exe-bc=x:\backup.xml x:`

where X equals the drive that hosts the Exchange Server -- if there are multiple volumes on this VM, list all volumes, separated by colon and a space (:), as in:

- `c:\vzshadow_x86.exe -bc=c:\backup.xml c: d: e:`

When VSS is triggered by VMware tools, the pre-freeze script will execute *vzShadow.exe*.

Step 1: Selecting An Inventory Node

The first step to creating a vRanger backup job is to select the inventory node (vCenter, host, Backup Group, etc) that you want to back up. After selecting the node, you may launch the Backup Wizard as described in the procedures below.

You may also create a custom grouping of existing objects from multiple hosts or vCenters. See Adding a Custom Backup Group for more information. Custom Backup Groups are treated the same as any other inventory node when it comes to creating backup jobs.

Note You may select an inventory node and start the backup wizard from both the My Inventory View and the My Jobs view. The procedure is slightly different for each, as detailed below.

Adding a Custom Backup Group

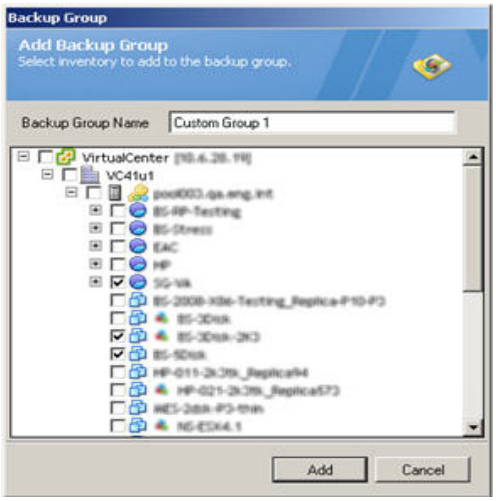
This feature allows a user to group different VMs, hosts, and folders together, across virtual centers and hosts for the purpose of creating a backup job for only the selected nodes. This provides additional flexibility for creating backup jobs.

To add a custom backup group:

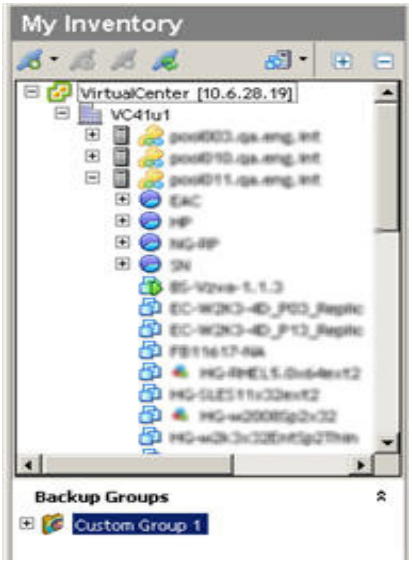
1. On the **My Inventory Screen**, click the **Add** Icon and select **Backup Group**.

   or

2. Expand the Backup Group panel in the Inventory pane, right-click on an empty spot, and click **Add Backup Group**.
3. Fill in the desired name of the **Custom Backup Group**, and click to select any node that you wish to include.



4 Clicking **Add** closes the **Add Backup Group** screen -- your new Custom Backup Group is listed in the **Backup Groups** pane.



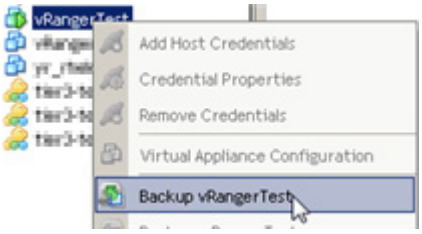
You can edit or remove a group by right-clicking it and selecting **Edit** or **Remove**.

Selecting a Node in the My Inventory View

When working in the My Inventory View, launching the Backup Wizard will create a job to backup the highlighted inventory node or backup group. If a node s not highlighted, the **Add** icon in the toolbar is disabled.

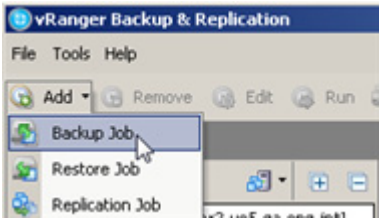
1 When viewing the My Inventory View, select the desired node in the Inventory Pane to highlight the object or backup group.

2 Right-click the highlighted node and select **Backup <object name>**.



or

With the desired inventory node selected, click the **Add** icon and select **Backup Job**.



3 The **Backup Wizard** will launch. Enter a name and description for the backup job, and click **Next**.

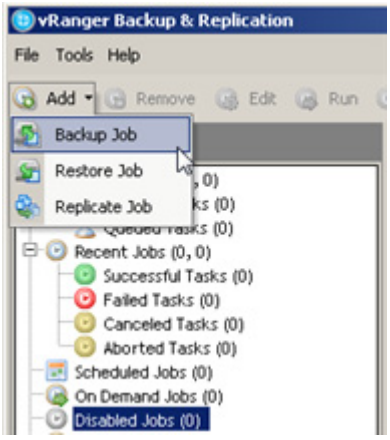
Selecting a Node in the My Jobs View

You may also launch the Backup Wizard from the Scheduled Jobs, On Demand Jobs, and Disabled Jobs node of the My Jobs view. As there will be no inventory object selected, the process is slightly different than when using the My Inventory View.

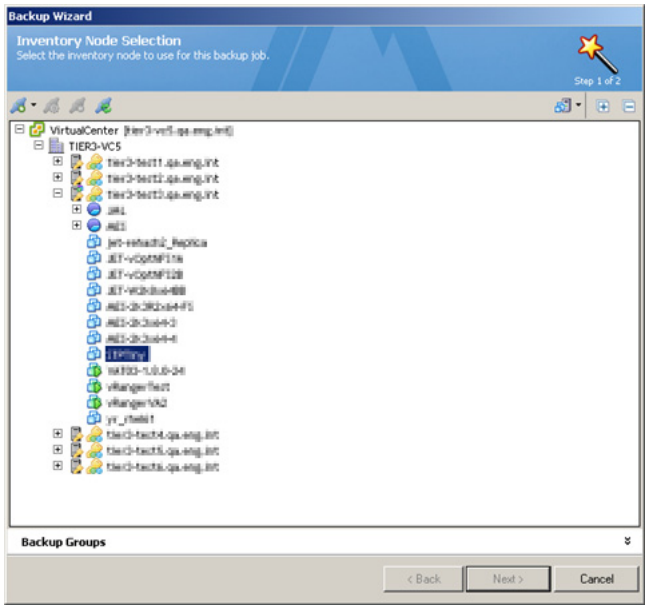
1 From the My Jobs view, click either the Scheduled Jobs, On Demand Jobs, and Disabled Jobs node. This will cause the Add icon in the toolbar to become enabled.

Note The node you select has no bearing on the status of the created job.For example, selecting the Disabled Jobs node will not cause the job to be created in a disabled state.

2 Click the **Add** icon and select **Backup Job**.



3 The Backup Wizard will start with the Inventory Node Selection, as shown below.



4 From the Inventory Pane displayed in the wizard, select the node to backup.

Note If you want to back up more than one node, you will need to create a Custom Backup Group and select that for the backup job. See Adding a Custom Backup Group for more information.

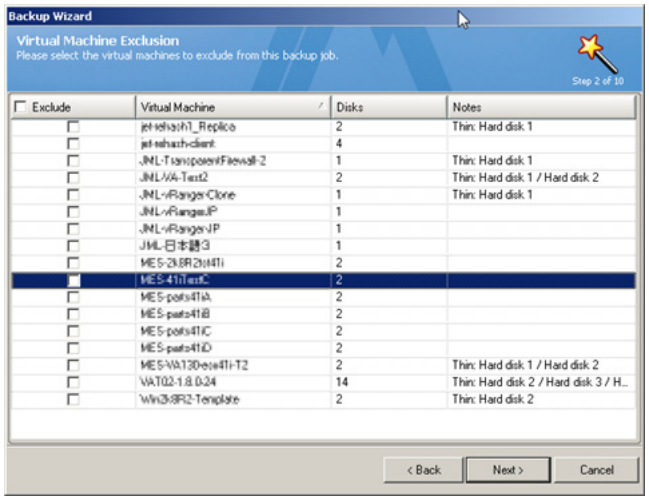
5 Click **Next**.

Step 2: Exclusions and Inclusions

If you selected a node that includes multiple VMs (such as a host or backup group), then next screen in the Backup Wizard will be the **Virtual Machine Exclusion** dialogue, followed by the **Hard Disk Inclusion** dialogue. If you selected a single VM, you will see only the **Hard Disk Inclusion** dialogue.

Virtual Machine Exclusion [Optional]

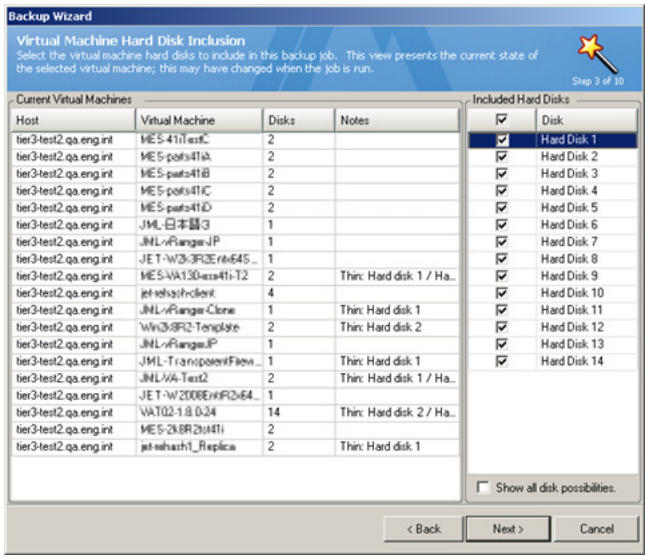
This dialogue will only be shown if you created a job for a node that includes multiple virtual machines, such as a host or backup group. On this screen, you may select individual VMs to **exclude** from the backup job. For example, if you want to backup all of the VMs on a host except 2, you would configure the backup job for the host node, and exclude the two VMs on this screen.



1. In the **Exclude** column on the left, select each VM that you want to exclude from the backup job.
2. Click **Next**.

Virtual Machine Hard Disk Inclusion

The Virtual Machine Hard Disk Inclusion dialogue creates a rule for vRanger to use when determining which disks to back up.



When you are creating a backup job for one VM, the rule translated directly to selecting which disks to back up. When you are creating a job for multiple VMs, however, the rule determines which disks will be backed up *for every VM*.

The **Include Hard Disks** pane on the right side of the dialogue represents the maximum number of disks configured for any VM in the job. For example, if you have a backup job with some number of single-disk VMs, and one VM with 6 disks, the Include Hard Disks column will show 6 entries.

Selecting “Hard Disk 1” through “Hard Disk 6 “will tell vRanger to back up every disk for each VM.

If, in the example above, the 6-disk VM gets an additional disk, it will not be backed up as only 6 disks are included. In order to allow for future growth, the **Show all disk possibilities** option will allow you to select all *possible* disks to ensure that any disk added to a VM in the future is included in the backup job.

To include hard disks in the backup job:

1. In the **Include Hard Disks** pane, configure a disk inclusion rule that vRanger should apply to each VM.

   Important Remember that this selection is not VM-specific. It is global for all VMs in this backup job.

2. Optionally, if you want to allow for additional disks in the future, select **Show all disk possibilities**, and select additional disks. You can select up to 64 disks, which is the VMware limit on a VM.
3. Click **Next**.

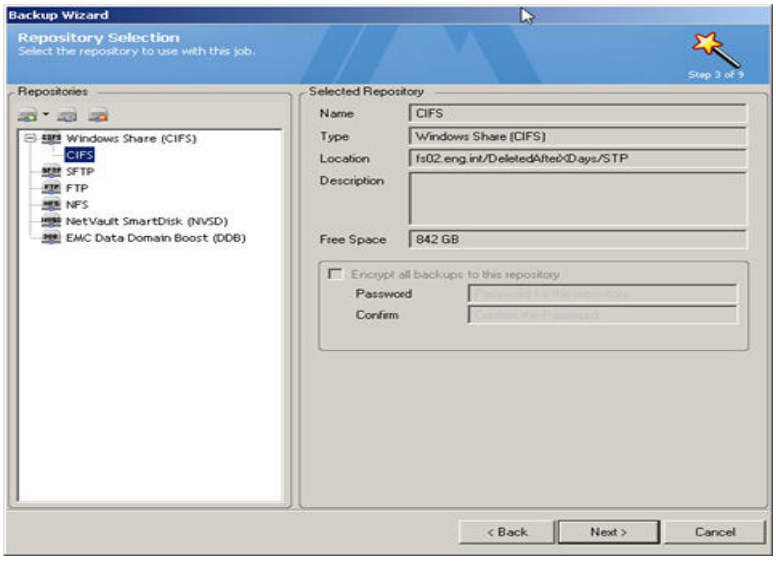
Step 3: Repository Selection

vRanger uses repositories to store backup archives. Repositories can be either a CIFS, NFS (version 3), FTP, or SFTP share. In addition, you may also deduplicate your backups by using a NetVault SmartDisk (v1.5 or later) instance or Data Domain system with DD Boost as a repository.

Note For more information on **NetVault SmartDisk**, see the SmartDisk page of the Quest Software website: *http://www.quest.com/netvault-smartdisk/*. For more information on Data Domain’s **DD Boost** technology, see the Data Domain website: *http://www.datadomain.com/products/dd-boost.htAdminiml*

On the Repository Selection dialogue, you can select the repository to which the job should send the backup data. You may also add a new repository from this dialogue. See Add Repositories for more information.

Note Only one repository can be configured per backup job.



1 In the Repositories pane, select the repository for this backup job.

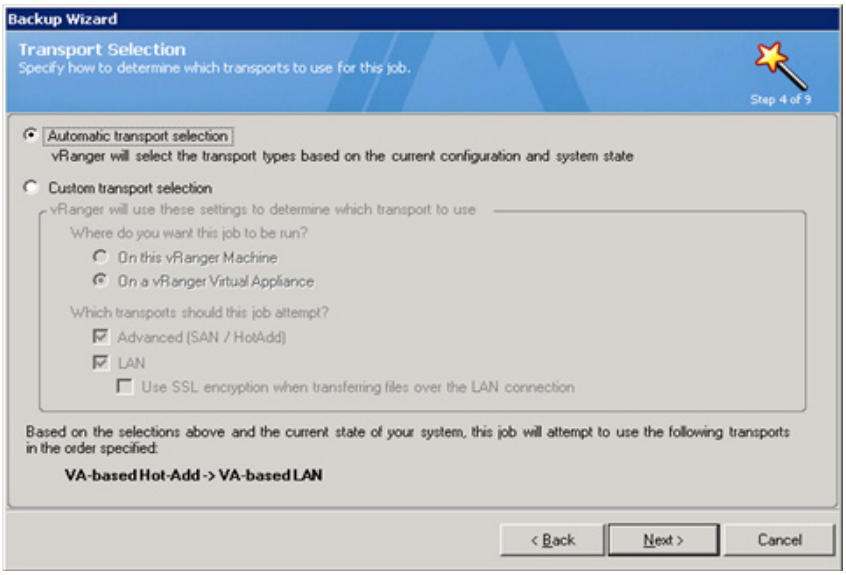
or

Add a new repository as needed. See Add Repositories for more information.

2 Click **Next**.

Step 4: Transport Selection

In vRanger, the transport determines how backup data is sent, and where the backup processing activity occurs (see Backup Transport Options for more information). The Backup Wizard, by default, will use the Automatic Transport Selection option to automatically select the best transport at run-time, but you can also configure the transport manually as well (via the Custom Transport Selection option).



Automatic Transport Selection

The Backup Wizard includes an **Automatic Transport Selection** option which allows vRanger to select the best available method for your configuration. When determining the best transport, vRanger will use two key criteria:

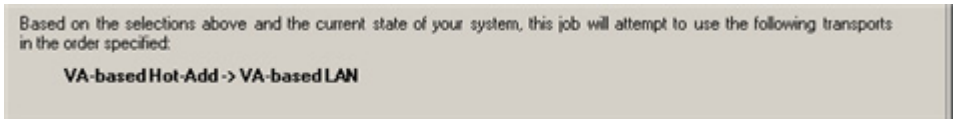
- whether virtual appliances are deployed
- where vRanger is installed

The logic used to select a transport differs slightly based on whether vRanger is installed in a VM or on a physical server. See Transport Selection Order for more information on the order in which vRanger will check for transport availability.

Using Automatic Transport Selection

1 On the Transport Selection dialog, select **Automatic Transport Selection**.

The selected transport method will be shown at the bottom of the dialogue:



Important Each time the job is run, vRanger will detect your configuration and select the best transport method available at run-time. That method may not always be what is shown on the dialogue during the job setup.

2 vRanger will check if any virtual appliances are configured on the source host. If the host belongs to a cluster, and no host VA is found, vRanger will check for shared virtual appliances on the cluster as well. If no VA is detected, the message below will be shown, as well as the **Configure Virtual Appliance** button, which will allow you to add a virtual appliance to vRanger.

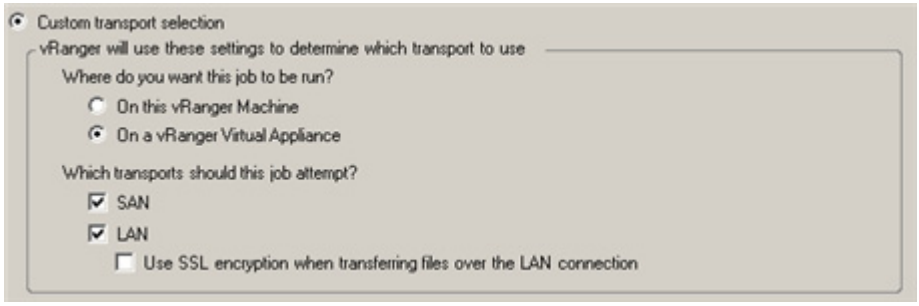


See Adding the Virtual Appliance to vRanger for procedures on the Virtual Appliance Configuration dialogue.

3 Click **Next**.

Custom Transport Selection

The Custom Transport Selection Option allows you to specify both where the job will be run (where the actual backup processing activity occurs), and which transport to use. See Backup Transport Options for more information on transports.



To configure a Custom Transport Selection:

1. In the Custom Transport Section, first select where you want the job to run - on the vRanger machine, or on a virtual appliance.

   vRanger will check if any virtual appliances are configured on the source host. If the host belongs to a cluster, and no host VA is found, vRanger will check for shared virtual appliances on the cluster as well. If no VA is detected, the message below will be shown, as well as the **Configure Virtual Appliance** button, which will allow you to add a virtual appliance to vRanger.

   See Adding the Virtual Appliance to vRanger for procedures on the Virtual Appliance Configuration dialogue.

   Note Using the vRanger virtual appliance is recommended for scalability and performance.

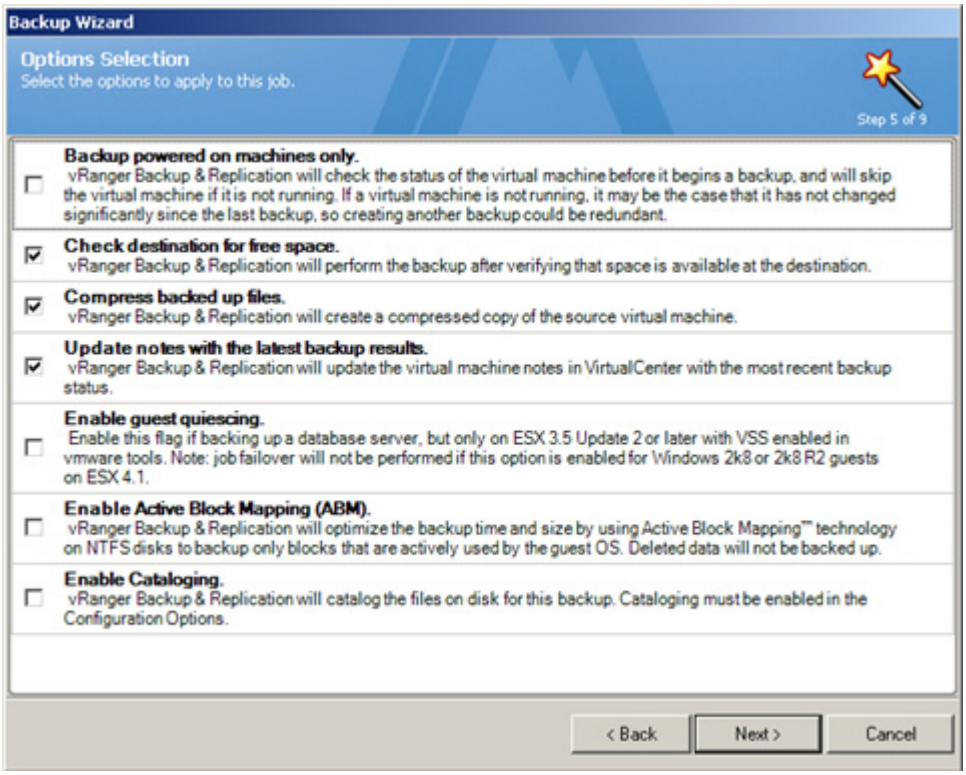
2. Select which transports the job should attempt. See Backup Transport Options for more information on the transport options.
3. [Optional] If you want your LAN backups to be encrypted, select **Use SSL encryption when transferring files over the LAN connection**.

   Caution This option will cause reduced performance.

4. Click **Next**.

Step 5: Configure Backup Options

The Backup Options dialogue allows you to configure additional operations performed during the backup job.



See the table below for more guidance on configuring backup options:

Option	Description
Backup powered on machines only.	If this flag is selected, vRanger will check the status of the VM before it begins a backup. If the VM is not running, it may not have changed significantly since the last backup, making another backup redundant.
Check destination for free space.	This option prompts vRanger to check the destination location to ensure that there is enough space available for the backup file to be written. Because it is not possible to determine how much space will be saved by compression, vRanger will generate an error if the free space at the destination is less than the combined size of the files belonging to the VM.
Compress backed up files.	This option prompts vRanger to compress the archive. **Note** Compression is not supported for NetVault Smartdisk and Data Domain DD Boost repositories.

Update notes with latest backup results.	If this flag is selected, vRanger will update the VM Notes in VC with the most recent backup status. **Note** vRanger maintains the existing VM notes in vCenter, appending the backup status to the end.
Enable guest quiescing.	This option enables the VMware Tools when creating a backup. Use this flag if you are backing up a database server. During this process, vRanger freezes or pauses writes on supported applications to provide a consistent image of VSS-enabled databases. See Determining Application Consistency for more information.
Enable Active Black Mapping	Checking this box enables Active Block Mapping, which allows vRanger to skip zeroed out blocks and junk data. **Note** With this option enabled, deleted data on the VM will not be backed up.
Enable Cataloging	Cataloging allows you to browse, search, and restore file level content in a backup image. See Cataloging Options for more information.

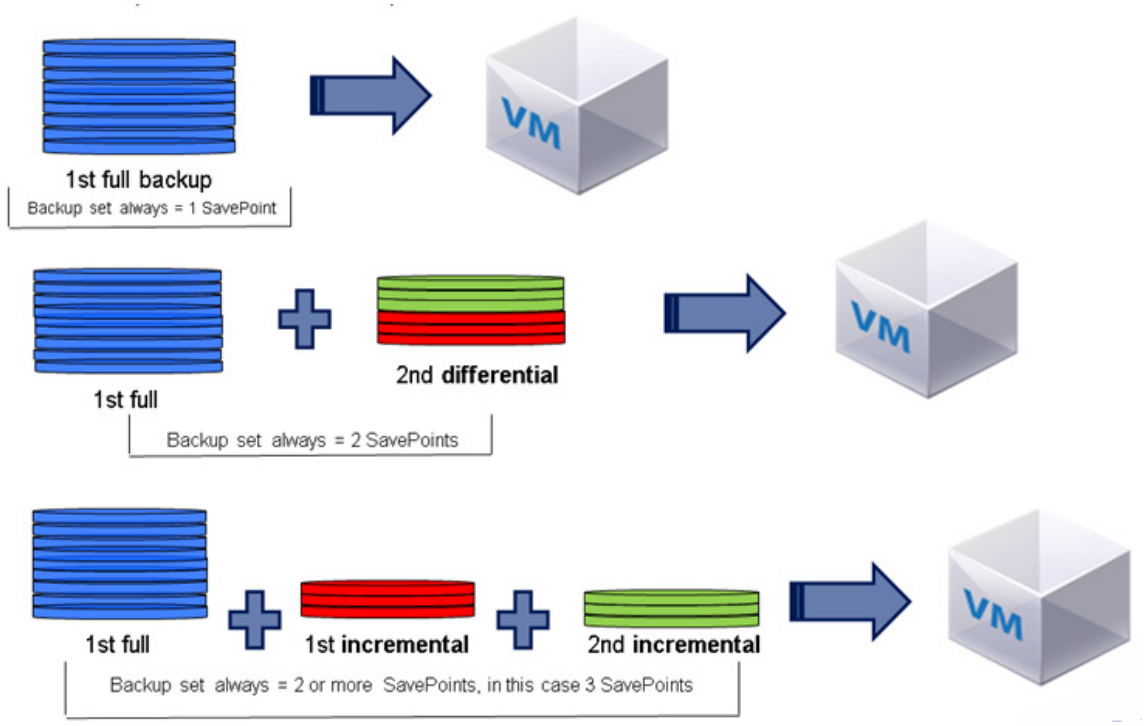
Step 6: Retention Policies & Space Saving Technologies

The Retention Policy Selection dialogue allows you to configure the type of backup to perform (full, incremental, or differential) and the minimum number of savepoints to keep for each VM.

Retention Policies

vRanger retention policies define the minimum number of restorable savepoints to keep in a repository for a backup job. The actual number of savepoints that will be retained at any one time depends on the type of backups being performed (full, differential, or incremental) and where in the full/differential or full/incremental cycle the retention check is being performed. vRanger will not retire an expired savepoint if that savepoint is required to restore a differential or incremental savepoint still within the retention period. See the sections below for more information on retention polices for each backup type.

A key concept in discussing retention policies is the notion of a “backup set”.A backup set is a number of savepoints that are grouped together for two reasons. The first is to implement the retention policy and the second is to maintain the parent-child relationships that exist when we choose to use differential or incremental backups. The backup set for each type of backup is different, as described below.



Full Backups

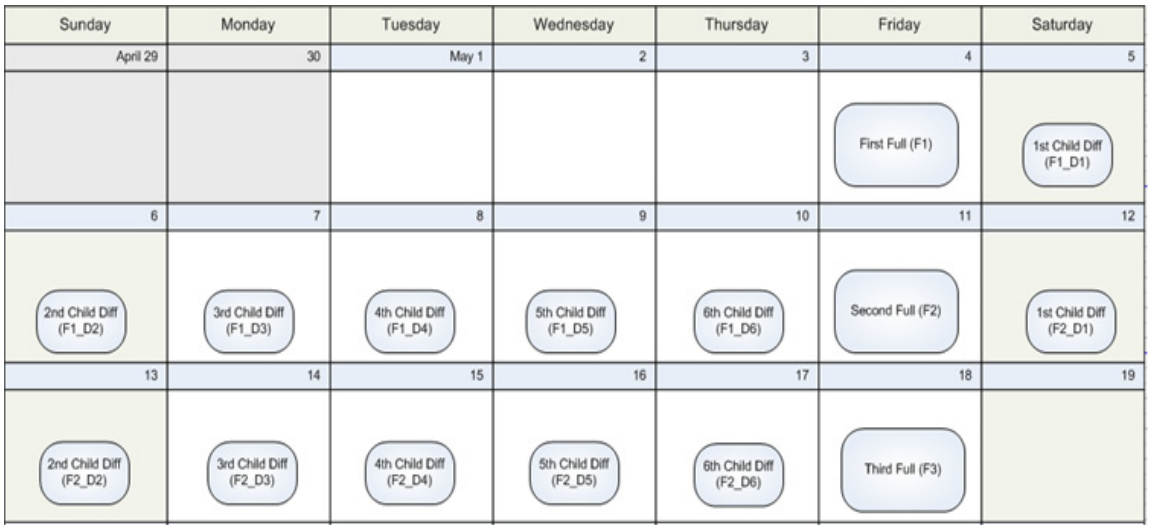
A backup job consisting only of full backups is the simplest case for retention policy configuration. Each savepoint is independent of the other savepoints, and can be retired without affecting any other savepoints. In other words, a backup set for full backups is equal to 1 savepoint - only one savepoint is required for a restore.

Assume a Savepoint Count of 7 is set. This means that vRanger will store 7 savepoints before it begins to retire savepoints from the repository. As the retention policy is checked after each successful backup, only after the 8th full backup is successful with the retention policy execute, removing the first savepoint from the repository and the vRanger database. With each subsequent successful backup, the oldest savepoint will be removed.

Differential Backups

Differential savepoints require the parent full backup and the selected differential savepoint in order to restore.Each differential backup includes the contents of the previous differential, which means, in terms of retention, that only the parent full and the most recent differential are required for a restore.

Assume, as an example, that you have a differential backup scheduled with a Threshold Count of 6 - meaning that 6 differential backups will be performed after the parent full.This schedule will produce a full backup on the same day each week, as shown in the image below.



Using the example schedule above, vRanger will process retention as described below.

- **Friday the 4th**- The first parent full backup (F1) is taken.
  - This makes 1 savepoint in the repository, which is below the minimum value. No retention action is taken.
- **Saturday the 5th**-The first child differential backup (F1_D1) is taken.
  - This makes 2 savepoints in the repository, which is below the minimum value. No retention action is taken.
- **Sunday the 6th**-The second child differential backup (F1_D2) is taken.
  - This makes 3 savepoints in the repository, which is below the minimum value. No retention action is taken.
- **Monday the 7th-** The third child differential backup (F1_D3) is taken.
  - This makes 4 savepoints in the repository, which is below the minimum value. No retention action is taken.
- **Tuesday the 8th**- The fourth child differential backup (F1_D4) is taken.
  - This makes 5 savepoints in the repository, which is below the minimum value. No retention action is taken.
- **Wednesday the 9th**- The fifth child differential backup (F1_D5) is taken.
  - This makes 6 savepoints in the repository, which is below the minimum value. No retention action is taken.
- **Thursday the 10th**- The sixth (and last) child differential backup (F1_D6) is taken.
  - This makes 7 savepoints in the repository, which satisfies the minimum value, but retiring a savepoint will bring the total below the minimum. No retention action is taken.
- **Friday the 11th**- The second parent full backup (F2) is taken.

- This makes 8 savepoints in the repository. vRanger will check if it can delete the oldest savepoint F1, but it cannot. F1 is part of a backup set and the parent of the first differential (D1). Retiring F1 would break the backup set, which is not allowed.

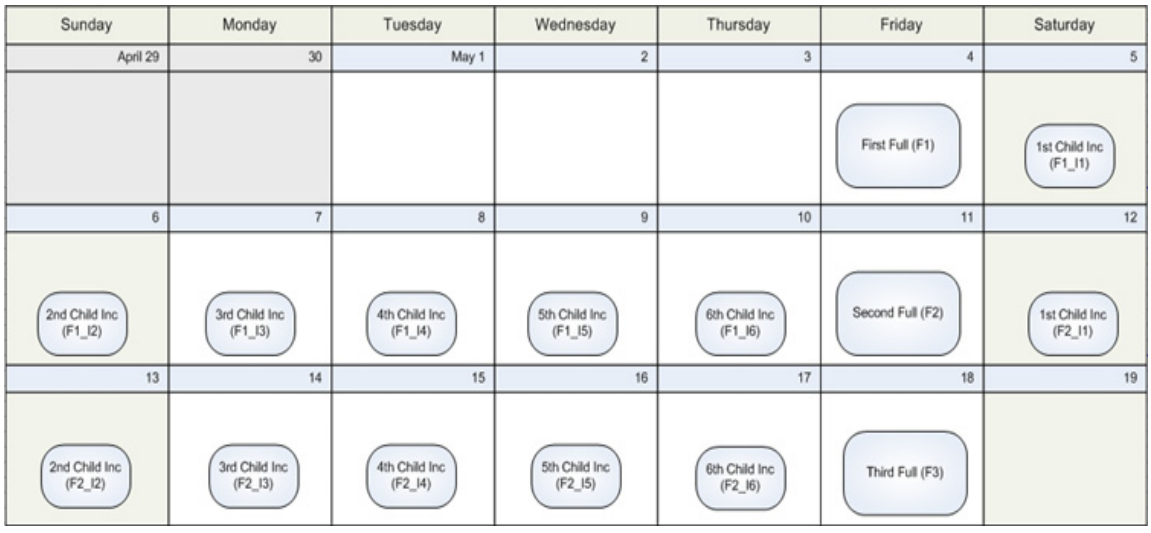
- **Saturday the 12th**- The first child differential of the new backup set (F2_D1) is taken.
  - This makes 9 savepoints on disk, which is more than the seven required from the savepoint Count setting. vRanger cannot delete the first full (F1) since the full is needed for all of the differentials from the first week, but it can remove the first differential (F1-D1)from Saturday the 5th. So F1D, the first differential from Saturday the 5th, is retired. After the first differential is retired, 8 savepoints remain on disk, since vRanger cannot remove the first full (F1).

On Sunday, the process continues, the second differential, F2_D2 is made, and so F1_D2 can be retired. The process continues: on Monday, Tuesday, and Wednesday the 16th. On Thursday the 17th, F2_D6 is made and F1_D6 is removed. This leaves 8 savepoints remaining. When vRanger checks to see if the first full (F1) can be removed, it determines that there are no dependencies to keep it and so it is removed as well. With differentials, vRanger can retire individual differentials but not the full until all of the children of the full are removed. Only when F1-D6 is removed will F1 be retired.

Incremental Backups

Incremental backups jobs backup only the blocks that have changed since the last backup - full or incremental. Incremental savepoints require the parent full *and* every incremental between the full and the selected incremental.

As an example, if you create an incremental backup job with a Threshold Count of 6, your backup set (all of the savepoints required for recovery) can reach 7 savepoints since the backups form a sequential chain starting with the full and continuing through each incremental backup.



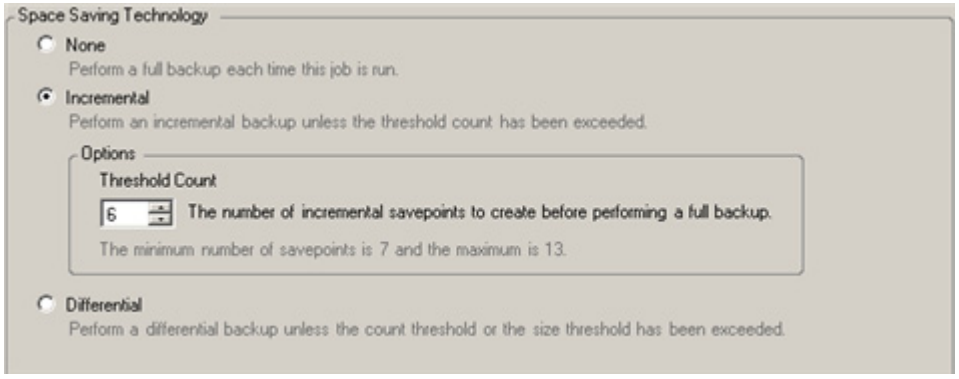
As in the differential backup example, assume a schedule with Full backups on Fridays, and incremental backups each day in between. vRanger will process retention as described below.

- **Friday the 4th**- The first parent full backup (F1) is taken.

  - This makes 1 savepoint in the repository, which is below the minimum value. No retention action is taken.
- **Saturday the 5th**-The first child incremental backup (F1_I1) is taken.
  - This makes 2 savepoints in the repository, which is below the minimum value. No retention action is taken.
- **Sunday the 6th**-The second child incremental backup (F1_I2) is taken.
  - This makes 3 savepoints in the repository, which is below the minimum value. No retention action is taken.
- **Monday the 7th-** The third child incremental backup (F1_I3) is taken.
  - This makes 4 savepoints in the repository, which is below the minimum value. No retention action is taken.
- **Tuesday the 8th**- The fourth child incremental backup (F1_I4) is taken.
  - This makes 5 savepoints in the repository, which is below the minimum value. No retention action is taken.
- **Wednesday the 9th**- The fifth child incremental backup (F1_I5) is taken.
  - This makes 6 savepoints in the repository, which is below the minimum value. No retention action is taken.
- **Thursday the 10th**- The sixth (and last) child incremental backup (F1_I6) is taken.
  - This makes 7 savepoints in the repository, which satisfies the minimum value, but retiring a savepoint will bring the total below the minimum. No retention action is taken.
- **Friday the 11th**- The second parent full backup (F2) is taken since we've reached the configured Threshold Count.
  - After the backup labelled F2 completes, vRanger checks the number of savepoints present and the savepoint Count for this job and determines that the 8 savepoints exceeds the minimum savepoint Count of 7. Next, vRanger looks to see if it can remove the oldest savepoint and determines that the first full cannot be removed, since it is part of a backup set with each of the incrementals F1-i1 to i6. So no old savepoints are purged and the job ends with 8 savepoints present. Retiring F1 would break the backup set, which is not allowed.
- **Saturday the 12th**- The first child incremental of the new backup set (F2_I1) is taken.
  - Again at the end of the task vRanger checks and determines that it cannot remove any old SavePoints since they are all part of the same set. So now the SavePoint Count is 9.
- **Sunday the 13th - Wednesday the 16th** - the same process continues, with the third through fifth incremental backups being performed, and the savepoint total exceeding the configured Savepoint Count. No savepoints are retired as that would break the F1 backup set.
- **Thursday the 17th** - the sixth and final incremental backup of the F2 backup set is performed.After this backup completes, there are 14 savepoints in the repository, with 7 savepoints in the F2 backup set. This means that the first F1 backup set can be retired and maintain the minimum savepoint value.

Space Saving Technologies

vRanger offers the following options to reduce the storage footprint and network load of VM backups. Each of the options in the Space Saving Technology section is described below.



None (Full Backups)

For full backups, vRanger performs a full scan and copy of the source VM. Full backup jobs can be modified with the following operations:

- **Active Block Mapping (ABM):** ABM will scan the disks on a virtual machine, and detect the blocks being actively used by the disk, as opposed to blocks that have been deleted by the Windows operating system. With ABM enabled, vRanger will only backup that part of a virtual disk that has active data on it.

  Caution ABM does not backup deleted data. If a VM backed up using ABM is restored, undelete operations will not be possible.

Incremental Backups

For incremental backups, vRanger will perform a full scan of the source VM, but copy only blocks that have changed since the last backup.

Incremental backups are typically the fastest backup option and consume less storage space per archive. It is important to note, however, that in order to restore from an incremental backup, each incremental archive between the full backup and the desired restore point must be available. When creating incremental backup schedules, use caution to minimize the risk introduced by long incremental chains.

Restoring an incremental backup can take longer than restoring a full or differential backup, because each previous incremental archive needs to be restored as well.

For incremental backups, you must configure a Threshold Count value. The **Threshold Count** refers to the number of incremental jobs that will be performed until a full backup is done.

Incremental backups can be modified with the following operations:

- **Active Block Mapping (ABM):** ABM will scan the disks on a virtual machine, and detect the blocks being actively used by the disk, as opposed to blocks that have been deleted by the Windows operating

system. With ABM enabled, vRanger will only backup that part of a virtual disk that has active data on it.

Caution ABM does not backup deleted data. If a VM backed up using ABM is restored, undelete operations will not be possible.

- **Changed Block Tracking (CBT):** Changed Block Tracking reduces the time needed for incremental and differential backups by only backing up the portions of a disk that have changed since the last backup. By determining which blocks changed within the VMDK file, vRanger only backs up the portions of a disk that have changed since the last backup. This will often result in shorter durations for backup operations, and reduce resource consumption on network and storage elements.

  Note CBT will copy deleted blocks if ABM is not also enabled.

Differential Backups

For differential backups, vRanger will perform a full scan of the source VM, but copy only blocks that have changed since the last Full backup.

Differential backups are typically slower than incrementals and can consume more storage space. They are usually faster to restore, however, and require only one differential backup and the parent full backup to restore.

For differential backups, you can configure a Threshold Count or a Threshold Size:

- The **Threshold Count** refers to the number of differential jobs that are performed until a full backup is done. A Threshold Count of 6 will cause a full backup to occur on the same day each week.
- The **Threshold Size** refers to the percentage of a VM's hard disk that must be captured in differentials before performing a full backup.

Using the Threshold Size value may cause your Full backups to be taken on a on unexpected days. Differential backups can be modified with the following operations:

- **Active Block Mapping (ABM):** ABM will scan the disks on a virtual machine, and detect the blocks being actively used by the disk, as opposed to blocks that have been deleted by the Windows operating system. With ABM enabled, vRanger will only backup that part of a virtual disk that has active data on it.

  Caution ABM does not backup deleted data. If a VM backed up using ABM is restored, undelete operations will not be possible.

- **Changed Block Tracking (CBT):** Changed Block Tracking reduces the time needed for incremental and differential backups by only backing up the portions of a disk that have changed since the last backup. By determining which blocks changed within the VMDK file, vRanger only backs up the portions of a disk that have changed since the last backup. This will often result in shorter durations for backup operations, and reduce resource consumption on network and storage elements.

  Note CBT will copy deleted blocks if ABM is not also enabled.

To configure a Space Saving Technology:

1. Select **None**, **Incremental**, or **Differential**.
2. For incremental backups configure a **Threshold Count** value.

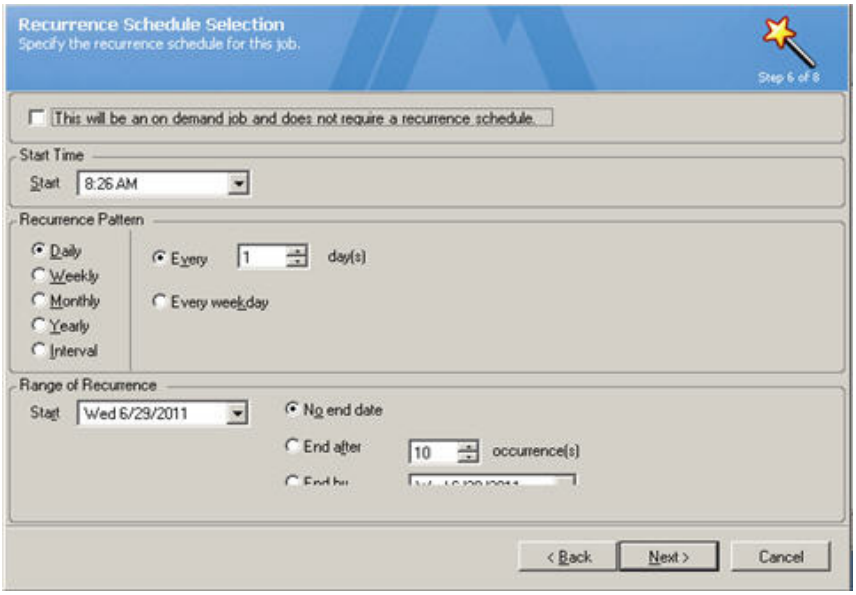
   or

   For differential backups, configure a **Threshold Count** value or a **Threshold Size** value.

3 Click **Next.**

Step 7: Recurrence Schedule Selection

On the Recurrence Schedule Selection screen, you indicate whether the job should be done on demand. If not, you must define start time, recurrence pattern, and range of recurrence.



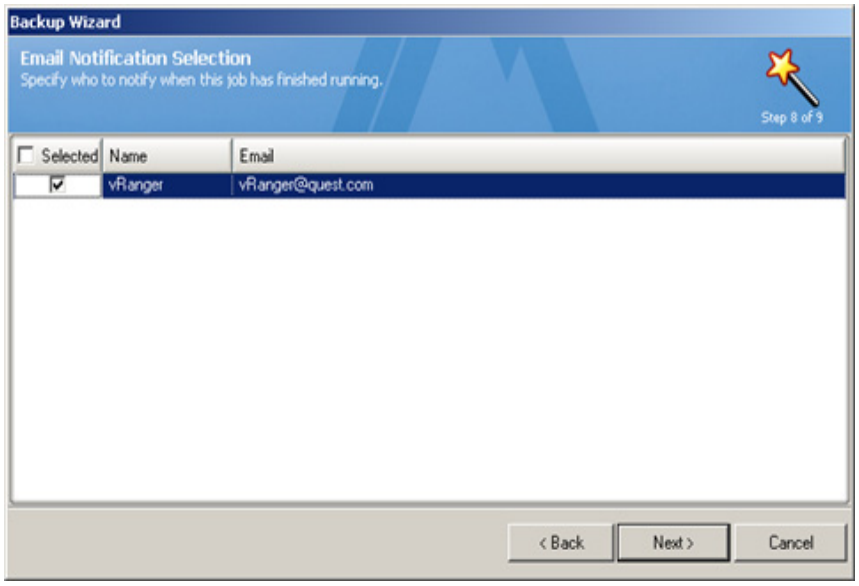
This will be an On Demand job . . .	There is no need to define a schedule. The job will be run on demand.
Start Time	The time of day at which to run the job.
Daily	The daily option can be scheduled to run every weekday or every x number of days.
Weekly	vRanger can be configured to run on weekly intervals, from every week to every 99 weeks. The day of the week on which to run jobs can also be configured.

Monthly	The monthly option offers the following configurations: • **Day** *[x]* **of every** *[y]* **month:** •*x* can be any value from 1 to 31. This determines the day of the month that the job will run. •*y* can be any value from 1 to 99. This value determines the monthly interval--for example, every 2 months sets the job to run every other month. • **The** *[f]* *[d]* **of every** *[y]* **month(s)**: •*f* can be either: first, second, third, fourth or last. •*d* can be: weekday, weekend day, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday. •*y* can be any value from 1 to 99. This value determines the monthly interval--for example, every 2 months sets the job to run every other month.
Yearly	The yearly option offers the following configurations: • **Every** *[m][x]*: • *m* is any month of the year. This determines the month of the year in which the job will run. • *x* can be any value from 1 to 31. This determines the day of the month that the job will run. • **The** *[f][d]* **of** *[m]*: • *f* can be either: first, second, third, fourth, or last. •*d* can be: day, weekday, weekend day, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday. • *m* is any month of the year. This determines the month of the year in which the job will run.
Range of Recurrence	Select the starting date and ending date. • **No end date**: The job will recur until stopped. • **End after**: Select the number of occurrences. • **End by**: Select the date by which to end the recurrence.

1. Configure the schedule options.
2. Click **Next.**

Step 8: Email Notification Selection

Through the **Email Notification Selection** screen, you select the email addresses to which notifications should be sent for this job. You can also request that a report be sent to the recipients identified once the job has completed. To configure email notifications, see Set up Mail Server Information/Add Email Addresses.

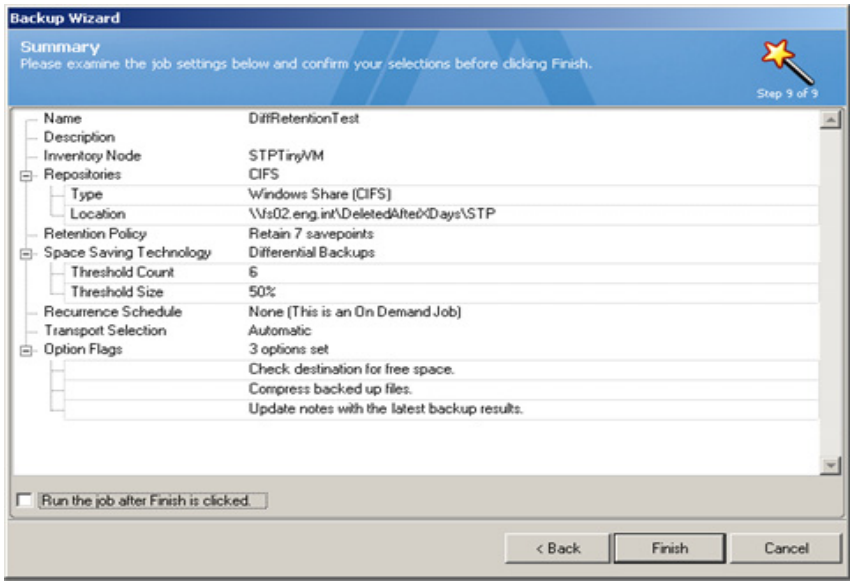


To select an email address:

1. In the **Selected** column, select the address to which notifications should be sent. To select all email addresses, select the check box in the Selected column header.
2. Click **Next**.

Step 9: Summary

The final screen in the **Backup Wizard** displays a summary of the selections you made during the process.



1 After confirming the summary, you can select the **Run the Job after Finish is clicked** check box to trigger the job to start immediately.

Note Selecting the **Run the Job after Finish is clicked** check box moves the job to the **Working Inventory** pane. To initiate running the job, you must then click run. This will move the job to **My Jobs** and start the backup.

2 Click **Finish** to close the Backup Wizard and save the job.

Restore

This chapter describes the restore options available in vRanger.

This chapter contains the following sections:This section contains the following topics:

The vRanger Restore Process 118
Performing a Full Restore 118
Performing a File-Level Restore - Windows 130
Performing a File-Level Recovery - Linux 132
Restoring from a Manifest 133

The vRanger Restore Process

vRanger offers three main options for restoring from backup: restoring the full image from a repository; restoring a file or files from a repository; or restoring an image or file from a manifest. See the sections below for more information

- Full Restore
- Restoring a File
- Restore From Manifest

Full Restore

Using vRanger, recovery of an entire VM, or multiple VMs, is a simple process. A full restore will return the VM(s) to its state as of the selected savepoint, or the point in time at which the backup was performed. If the savepoint is for a differential backup, the restore process will first restore the matched full backup, and then merge the data from the selected differential archive. For incremental savepoints, the full backup will be restored along with each incremental archive taken between the full and the selected savepoint.

During the normal restore process, information is pulled from the vRanger database to complete the restore job. In the event that the vRanger server is lost, and the database cannot be recovered (or is otherwise unavailable), you can reinstall a fresh version of vRanger and import the repository to regain access to your backups.

Restoring a File

Unlike traditional backup solutions that require file level agents, vRanger can recover a file or files directly from the image-level backup, without mounting the image. Incremental backups are combined with the parent full image as needed.

One of the key problems with file-level recovery is finding the appropriate files. vRanger includes a cataloging feature that indexes backups of Windows VMs on the fly, and records the metadata to enable faster searching. See Linux FLR Limitations for more information.

Restore From Manifest

Each savepoint contains a manifest file that can be used to restore the savepoint without access to the vRanger database. In addition, Restore from Manifest can be used to recover savepoints not part of a repository (those that have been recovered from tape, for example).

To restore a savepoint using the manifest file, see Performing a File-Level Restore - Windows.

Performing a Full Restore

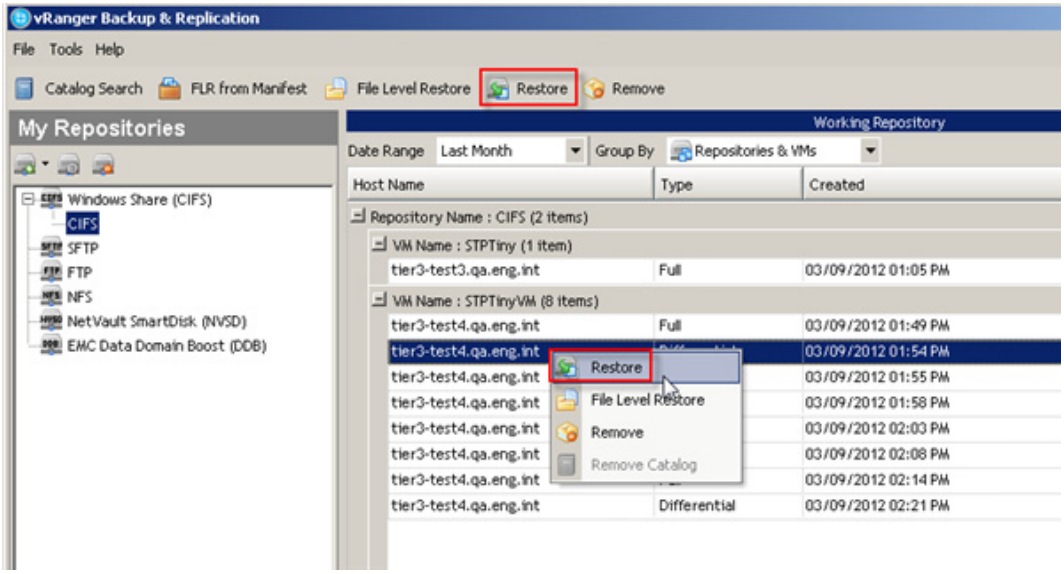
The restore process uses the vRanger Job Wizard, following the steps described below. For information on performing a full restore from a savepoint manifest, see Performing a File-Level Restore - Windows.

Step 1: Select Savepoint

1 Click **My Repositories.**

2 With **Repositories & VMs** selected, in the **Working Repository** pane click to select a savepoint to restore.

3 Click the **Restore icon** on the toolbar, or right-click the savepoint and click **Restore**.



Step 2: Restore Wizard

The vRanger Restore Wizard launches.

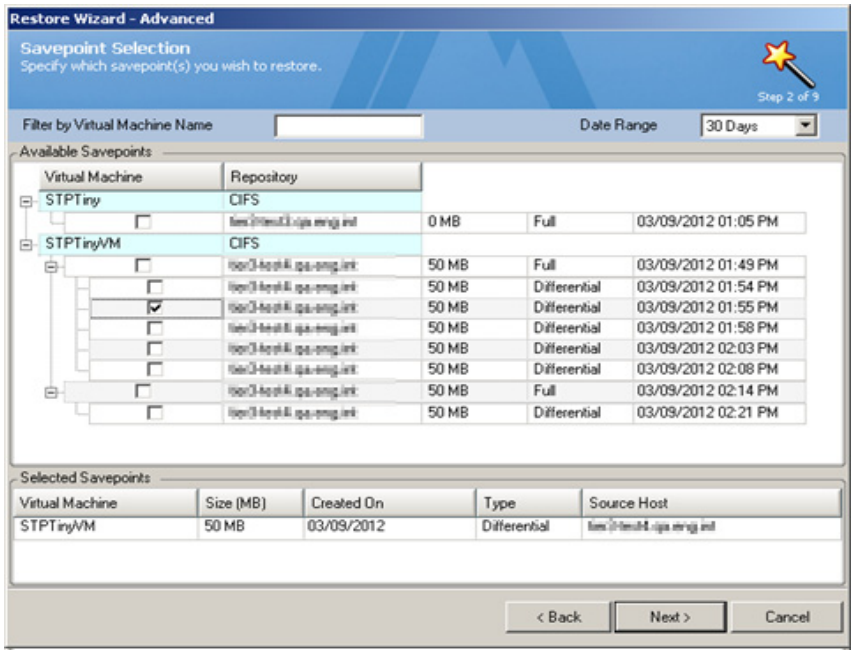
1 In the drop-down menu, select **Restore from Savepoint.**

Note For the procedures on restoring from a manifest file, see Full Restore from Manifest.

2 Populate the **Job Name** and **Job Description** fields. Click **Next**.

Step 3: Savepoint Selection

The **Savepoint Selection** dialog displays. All available savepoints are listed here, including their respective VMs and repositories.



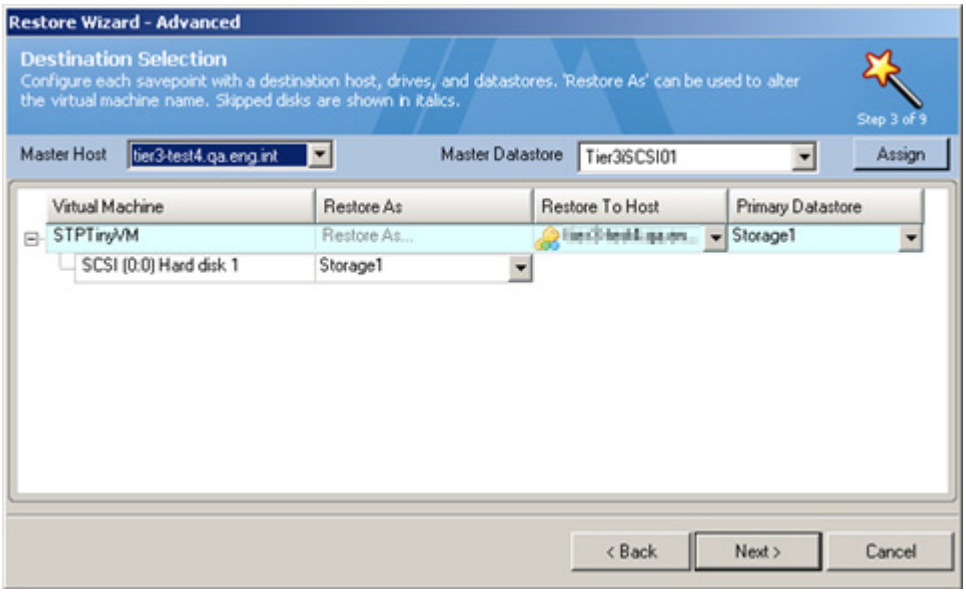
1 The savepoint you selected is checked. You can clear this and choose another savepoint, or choose an additional savepoint (for incremental backups).

Note If you select two or more savepoints with the same name on the same host, an error message will display and the Next button will be disabled until you chose your selection.

2 Click **Next.**

Step 4: Destination Selection

The **Destination Selection** dialog displays, which allows you to configure the host, drives, and datastores to which the VMs will be restored.



A datastore can be configured for each VMDK. You can also configure a Master Host and Master Datastore, which will use assigned masters for all VMs and VMDKs displayed. Skipped disks appear in italics on this dialog. Use the **Restore As** field to rename the VM you are restoring.

1. For each VM: In the **Restore To Host** column, select the host to which the VM should be restored.

   or

   From the **Master Host** dropdown, select the host to which all VMs should be restored. Click **Assign**.

2. For each VM: In the **Primary Datastore** field, select a value.

   or

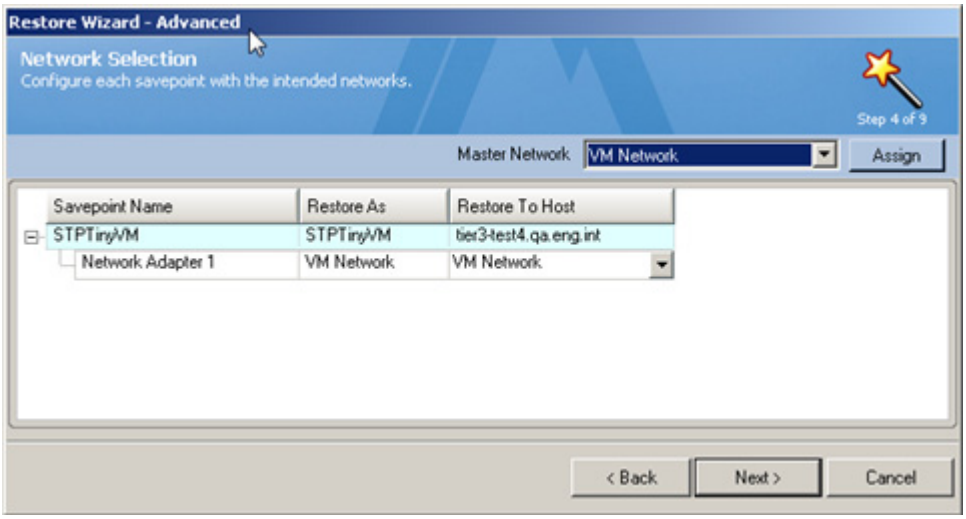
   In the **Master Datastore** field, select the datastore to be used for all VMs.

3. For each VMDK: Select the datastore to which each VMDK should be restored.

4. Click **Next**.

Step 5: Network Selection

The Network Selection dialog displays, which allows you to configure the virtual network to which each vNIC will be connected upon restore. You can also configure a **Master Network,** which will use the selected network for all vNICs.



1. For each VM, the connected network adapters will be listed: In the **Restore To Host** column, select the virtual network to which the vNIC should be connected.

   or

2. In the **Master Network** field, select the host to which all VMs should be restored.

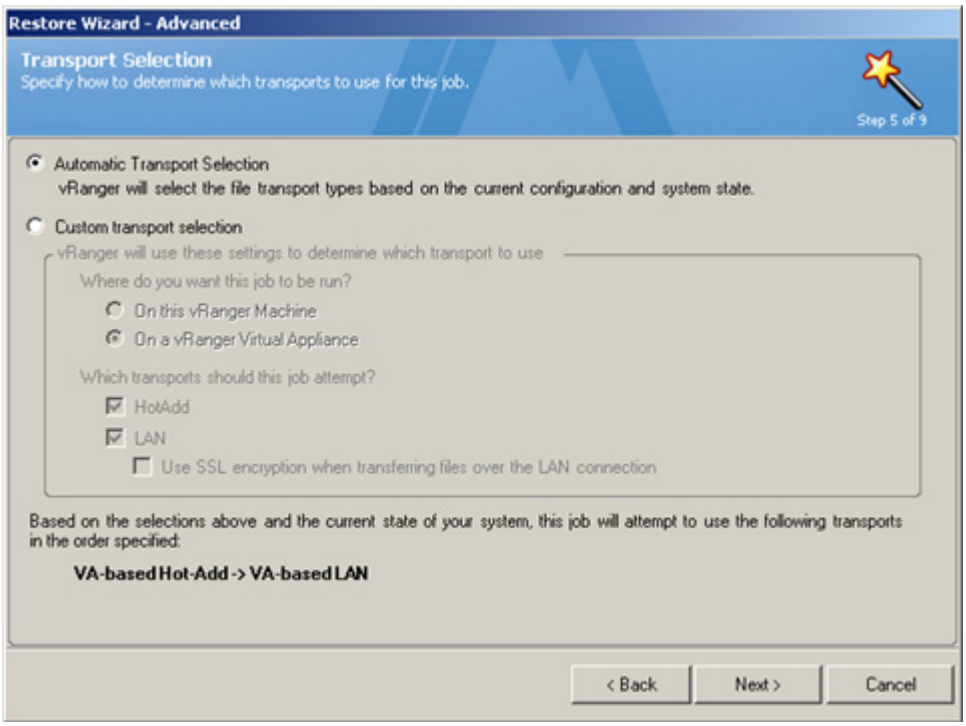
3. Click **Assign**.

4. Click **Next**.

Step 6: Transport Selection

In vRanger, the transport determines how backup data is sent, and where the restore processing activity occurs. See Backup Transport Options for more information on vRanger transports.

The Restore Wizard, by default, will use the Automatic Transport Selection option to automatically select the best transport at run-time, but you can also configure the transport manually as well (via the Custom Transport Selection option).



Automatic Transport Selection

The Restore Wizard includes an **Automatic Transport Selection** option which allows vRanger to select the best available method for your configuration. When determining the best transport, vRanger will use two key criteria:

- whether virtual appliances are deployed
- where vRanger is installed

The logic used to select a transport differs slightly based on whether vRanger is installed in a VM or on a physical server. The sections below describe the steps taken to select an automatic transport.

vRanger - Virtual Machine

When installed in a virtual machine, vRanger will check the possible transports in the order shown below, selecting the first option available. The preferred transports are checked first, ensuring the best performance possible.

1. **Virtual Appliance on source host** - vRanger will first check the source host for a virtual appliance. If available, that VA will be used.
   a. **Is VA-based HotAdd available?** - vRanger will check the local VA for HotAdd support. If the host is not properly licensed, or the VA cannot access the storage for the target host (the host to which the VM will be restored), HotAdd will not be available.

      Important If a virtual appliance is configured and HotAdd is not available, a network restore will be performed from the virtual appliance. No further transport checks will be performed.
2. **Virtual Appliance on source cluster** - if the target host does not have a virtual appliance, vRanger will check if the host is a member of a cluster, If so, vRanger will check for a shared virtual appliance on that cluster.
   a. **Is VA-based HotAdd available?** -vRanger will check the local VA for HotAdd support. If the host is not properly licensed, or the VA cannot access the storage for the target host, HotAdd will not be available.

      Important If a virtual appliance is configured and HotAdd is not available, a network restore will be performed from the virtual appliance. No further transport checks will be performed.
3. **Local- HotAdd** - If there is no virtual appliance configured, vRanger will perform a check to see if HotAdd is available for the vRanger VM. If the vRanger VM is on a host which is not properly licensed, or if the VM cannot access the storage for the target host, HotAdd will not be available.
4. **Local-LAN** - If a virtual appliance is not available, and HotAdd is not supported on the vRanger VM, a network restore will be performed from the vRanger machine.

vRanger - Physical Machine

When installed on a physical machine, vRanger will check the possible transports in the order shown below, selecting the first option available. The preferred transports are checked first, ensuring the best performance possible.

1. **Virtual Appliance on source host** - vRanger will first check the source host for a virtual appliance. If available, that VA will be used.
   a. **Is HotAdd available?** - vRanger will check the local VA for HotAdd support. If the host is not properly licensed, or the VA cannot access the storage for the target host, HotAdd will not be available.

      Important If a virtual appliance is configured and HotAdd is not available, a network restore will be performed from the virtual appliance. No further transport checks will be performed.
2. **Virtual Appliance on source cluster** - if the source host does not have a virtual appliance, vRanger will check if the host is a member of a cluster, If so, vRanger will check for a shared virtual appliance on that cluster.
   a. **Is HotAdd available?** -vRanger will check the local VA for HotAdd support. If the host is not properly licensed, or the VA cannot access the storage for the target host, HotAdd will not be available.

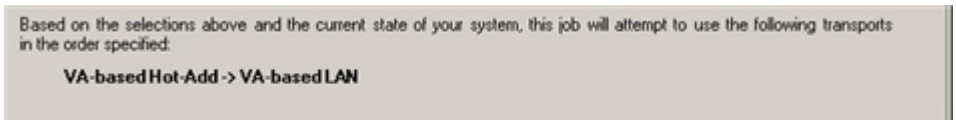
      Important If a virtual appliance is configured and HotAdd is not available, a network restore will be performed from the virtual appliance. No further transport checks will be performed.
3. **Machine-based SAN** - If there is no virtual appliance configured, vRanger will perform a check to see the vRanger server is configured for SAN restores.

4 **Machine-based LAN** - If a virtual appliance is not available, and SAN restores are not supported on the vRanger server,a network restore will be performed from the vRanger machine.

Using Automatic Transport Selection

1 On the Transport Selection dialog, select **Automatic Transport Selection**.

The selected transport method will be shown at the bottom of the dialogue:



Important Each time the job is run, vRanger will detect your configuration and select the best transport method available at run-time. That method may not always be what is shown on the dialogue during the job setup.

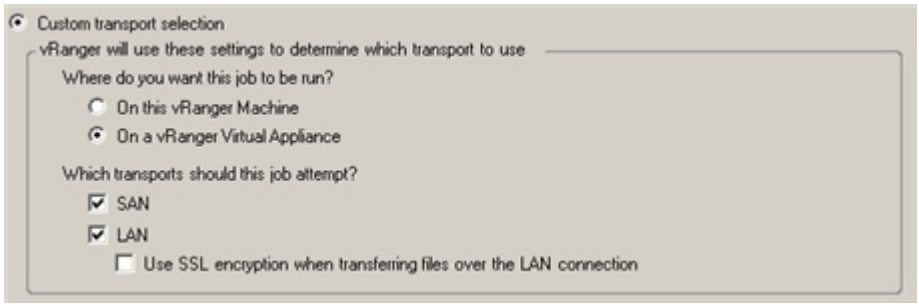
2 vRanger will check if any virtual appliances are configured on the target host. If the host belongs to a cluster, vRanger will check for shared VAs as well. If no VA is detected, the message below will be shown, as well as the **Configure Virtual Appliance** button, which will allow you to add a virtual appliance to vRanger. See Adding the Virtual Appliance to vRanger for procedures on the Virtual Appliance Configuration dialogue.



3 Click **Next**.

Custom Transport Selection

The Custom Transport Selection Option allows you to specify both where the job will be run (where the actual restore processing activity occurs), and which transport to use. See Backup Transport Options in the “Backup” chapter for more information on transports.



To configure a Custom Transport Selection:

1 In the Custom Transport Section, first select where you want the job to run - on the vRanger machine, or on a virtual appliance.

vRanger will check if any virtual appliances are configured on the target host (the host to which the VM will be restored). If the host belongs to a cluster, vRanger will check for shared VAs as well. If no VA is detected, the message below will be shown, as well as the **Configure Virtual Appliance** button, which will allow you to add a virtual appliance to

vRanger. See Adding the Virtual Appliance to vRanger for procedures on the Virtual Appliance Configuration dialogue.

Note Using the vRanger virtual appliance is recommended for scalability and performance.

2 Select which transports the job should attempt. See Backup Transport Options in the "Backup" chapter for more information on the transport options.

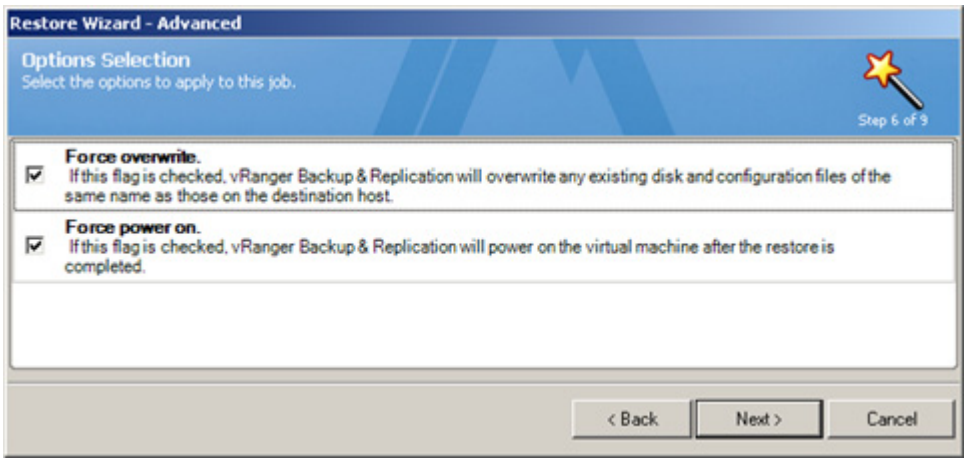
3 **[Optional]** If you want your LAN restores to be encrypted, select **Use SSL encryption when transferring files over the LAN connection**.

Caution This option will cause reduced performance.

4 Click **Next**.

Step 8: Options Selection

The **Options Selection** dialog displays.



1 Select restore options:

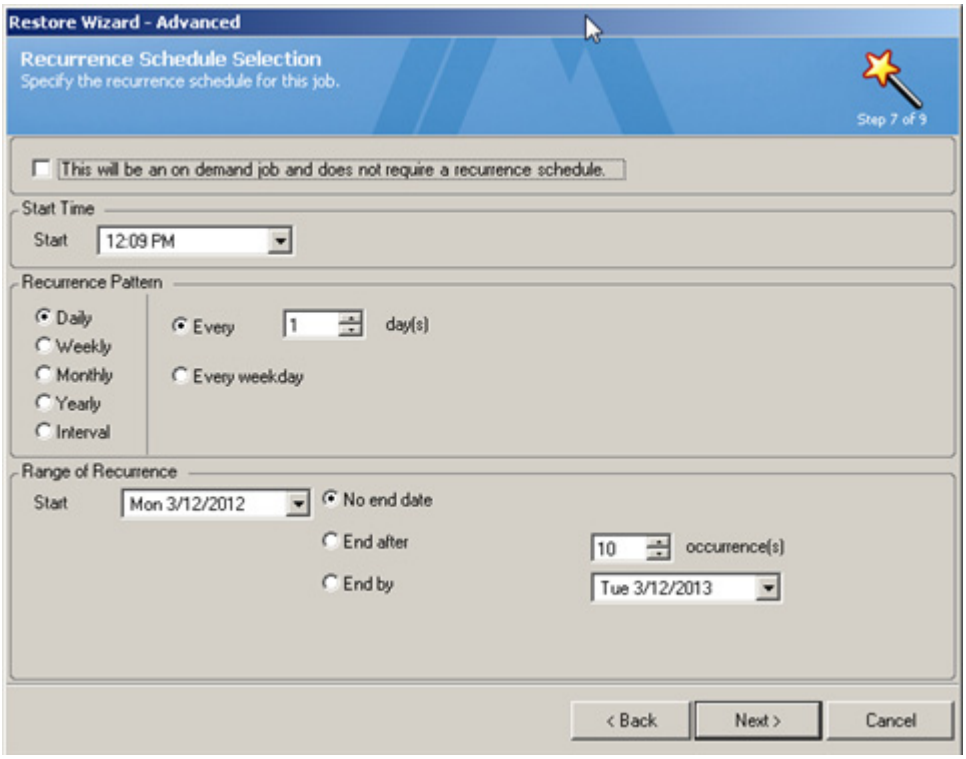
- **Force Overwrite:** This will overwrite any existing disk/config files of the same name as those in the destination host.
- **Force Power On**: This will power on the VM after the restore.

Note **Force overwrite** and **Force power on** are selected as default values.

2 Click **Next**.

Step 9: Recurrence Schedule

The **Recurrence Schedule Selection** dialog displays. On the **Recurrence Schedule Selection** screen, you indicate whether the job should be done on demand. If not, you must define start time, recurrence pattern, and range of recurrence.



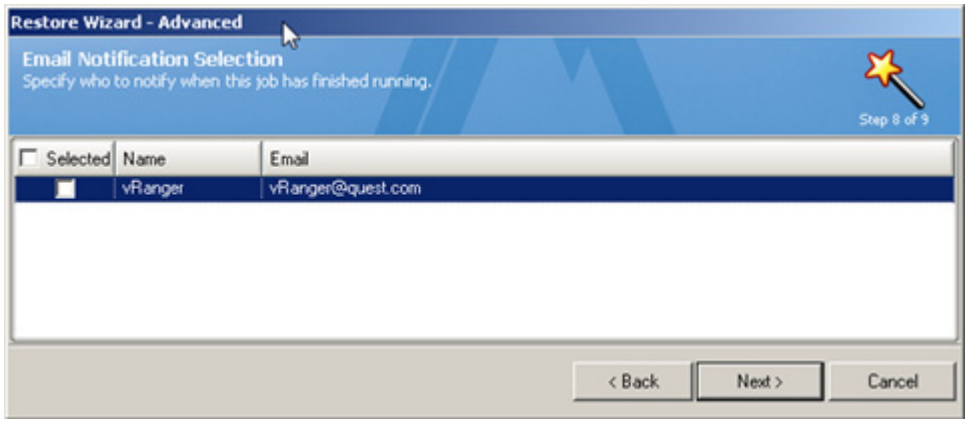
This will be an On Demand job . . .	There is no need to define a schedule. The job will be run on demand.
Start Time	The time of day at which to run the job.
Daily	The daily option can be scheduled to run every weekday or every x number of days.
Weekly	vRanger can be configured to run on weekly intervals, from every week to every 99 weeks. The day of the week on which to run jobs can also be configured.

Monthly	The monthly option offers the following configurations: • **Day** *[x]* **of every** *[y]* **month:** •*x* can be any value from 1 to 31. This determines the day of the month that the job will run. •*y* can be any value from 1 to 99. This value determines the monthly interval--for example, every 2 months sets the job to run every other month. • **The** *[f]* *[d]* **of every** *[y]* **month(s)**: •*f* can be either: first, second, third, fourth or last. •*d* can be: weekday, weekend day, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday. •*y* can be any value from 1 to 99. This value determines the monthly interval--for example, every 2 months sets the job to run every other month.
Yearly	The yearly option offers the following configurations: • **Every** *[m][x]*: • *m* is any month of the year. This determines the month of the year in which the job will run. • *x* can be any value from 1 to 31. This determines the day of the month that the job will run. • **The** *[f][d]* **of** *[m]*: • *f* can be either: first, second, third, fourth, or last. •*d* can be: day, weekday, weekend day, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday. • *m* is any month of the year. This determines the month of the year in which the job will run.
Range of Recurrence	Select the starting date and ending date. • **No end date**: The job will recur until stopped. • **End after**: Select the number of occurrences. • **End by**: Select the date by which to end the recurrence.

1. Configure the schedule options.
2. Click **Next.**

Step 9: Email Notification Selection

The **Email Notification Selection** dialog displays.Through the **Email Notification Selection** screen, you select the email addresses to which notifications should be sent for this job. You can also request that a report be sent to the recipients identified once the job has completed. To configure email notifications, see Set up Mail Server Information/Add Email Addresses.

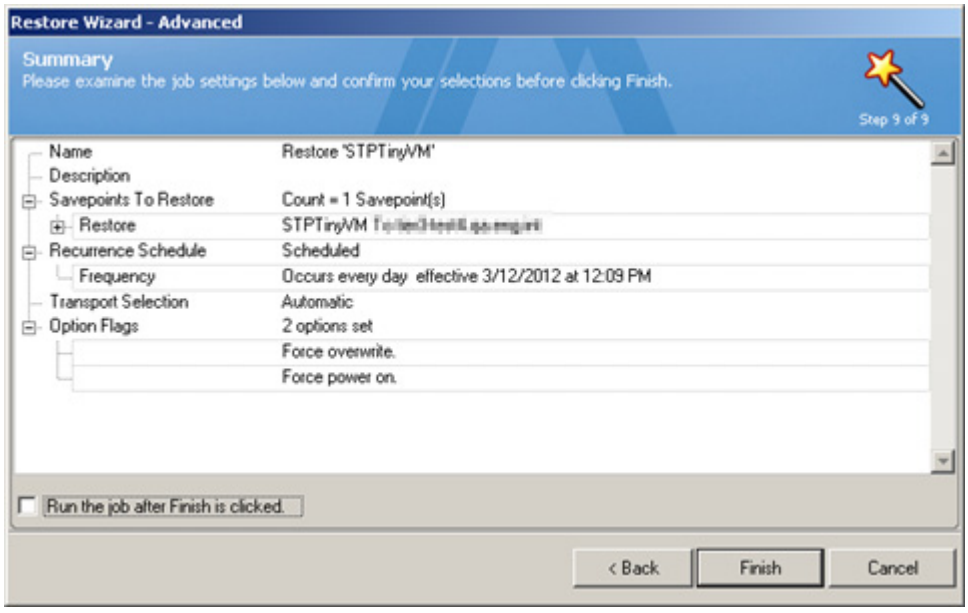


To select an email address:

1. In the **Selected** column, select the address to which notifications should be sent. To select all email addresses, select the check box in the Selected column header.
2. Click **Next**.

Step 9: Summary

The Summary dialog displays. The **Summary Screen** displays a summary of the selections you made during the process.



1. To run the restore immediately, select **Run Restore on Finish**.
2. Click **Finish**.

Step 10: Running the Restore

1. If you did not choose to run the restore from the wizard, access the **Working Inventory** pane. Select a job to restore.
2. Click **Run.**
3. The restore will begin immediately.
4. To check progress or confirm completion, access **My Jobs-->Recent Jobs**.

Performing a File-Level Restore - Windows

You can restore a file from a VM by accessing the FLR command in the My Repositories view. FLR is accessible regardless of how the savepoints are sorted. You can right-click the savepoint in the Working Repository pane to select the command or you can click to highlight the savepoint and then click the FLR icon in the toolbar.

There is no need to copy or recreate the contents of the disk. The data remains compressed. When you make an FLR request, only the relevant files are moved from the repository to the designated computer.

A key component of the vRanger file-level recovery is the vRanger catalog functionality. While you can use FLR without cataloging enabled, it is much easier to find a desired file via a catalog search than through manual browsing of savepoints.

FLR- Limitations and Requirements

There are several limitations and requirements that apply to file-level restores.

Note These are summarized briefly below, see the *vRanger System Requirements Guide* for complete information.

- **Disk types**- VMs with dynamic disks are not supported for FLR.
- **GPT Partitions** - VMs with GPT partitions are not supported for FLR

Linux FLR limitations are summarized in Linux FLR Limitations.

File Level Restore - Catalog Search

Cataloging allows searching and browsing of savepoints for files inside the VM images. Cataloging must be enabled at the start of the backup job to function, and also enabled globally in the Configuration Options dialog.

Catalog Search has the following parameters:

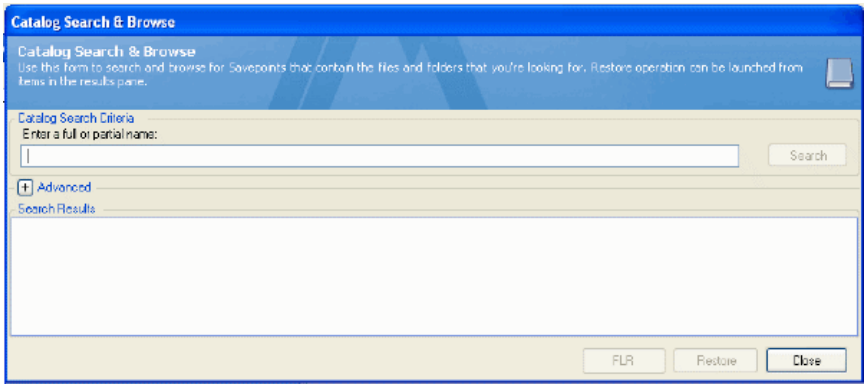
- when cataloging a full backup, all files in the system are stored in the DB.
- when cataloging an incremental backup, only the files that were modified since the last backup are stored in the DB.
- When cataloging a differential backup, only the files that were modified since the last full backup are stored in the DB.

Note When it comes to reporting the status of a backup task, cataloging is not considered a critical function. The cataloging portion of the task can fail, and the task will still be considered successful.

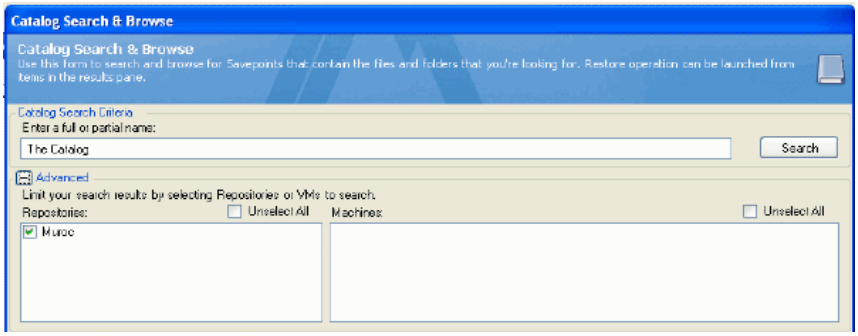
To use the Catalog function:

1 Go to the **My Repositories Pane**, and select the repository you wish to search.

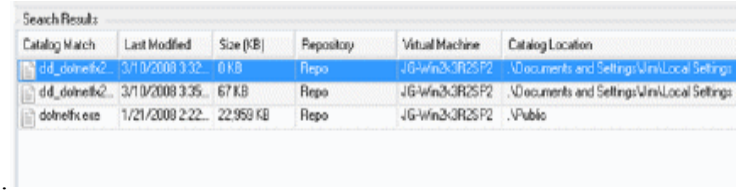
2 Click the **Catalog Search Icon** , or, on the **Menu Bar,** select **Tools-- > Catalog Search.** The **Catalog Search & Browse** screen displays.



Entering a search string (in the **Catalog Search Criteria** box) will search all repositories and savepoints for the string. To further refine the scope of the search, click **Advanced**



3 The **Advanced** option allows you to limit the search to a repository or VM (savepoint). Select the repository or VM you wish to search, enter the search string, and click **Search**



4 Highlight the desired savepoint, and click **FLR** for File Level Restore, or **Restore** for the Full Restore.

Note Catalog searching supports the traditional wildcard character (*) in any position. The search string can be as short, or as long as you desire, however, the shorter the string, the longer the search will take.

SQL Express is limited to one CPU and 1 GB of RAM; an extremely short search string (for instance, “dot”) could result in very slow searching, and SQL express could run our of memory.To minimize performance issues during Catalog Search, make the search string as specific as possible.

File Level Restore - Manual

This dialog is made up of three panes:

- **Savepoint Pane**--In this area, all of the files or folders connected to a savepoint are nested. When you click the file or folder that you want to restore, the path is listed above it.
- **Select Destination Pane**--This pane includes a tree structure of all possible destinations for the restored file or folder you select.
- **Restore Status Pane**--This area of the dialog contains data once the restore process has started. It includes a column that indicates the percentage of the job that has completed. The dialog also features a *Stop* link that, when clicked, ends the restore process. When a job is stopped or it has completed, a *View in Explorer* link displays.

Step 1: Select Savepoint

1. Click **My Repositories.**
2. In the **Working Repository Pane**, right-click the savepoint for which you want an FLR and select **File Level Restore.**

Step 2: Select Files

1. In the Sel**ect the Files/Folders to Restore** pane, expand the archive to view the file structure.
2. Select the file or files to recover.

Step 3: Select Destination

1. In the **Select Destination** pane, select the destination to which the selected files should be recovered.

   The **Select Destination** pane shows the Windows Explorer view for the machine on which vRanger is installed.

   Caution vRanger will not check for previous versions of the files you are restoring. Files restored using FLR will overwrite any previous files (of the same name) in the restore destination.
2. Click **Restore**.
3. In the **Restore Status** pane, view the status of the restore activity and the value in the Percent Finished column. Status should be Finished and Percent Finished should be 100%. To view the restored file or folder, click the ***View in Explorer*** link.

The user needs the ability restore a virtual machine whose backup files are not in a repository. This allows the user to move the files that make up a backup to wherever they wish and still perform a restore.

Performing a File-Level Recovery - Linux

File Level Restores from Linux servers require a virtual appliance be installed and setup. See the *vRanger Installation and Setup Guide* for instructions on deploying and configuring the vRanger virtual appliance.

Performing a Linux File-level Restore

Once the vRanger virtual appliance is configured, the FLR process for a linux VM is the same as the manual FLR process (meaning the process not using a catalog search) for a windows VM. File Level Restore - Manual

Linux FLR Limitations

There are several limitations and requirements that apply to file-level restores from linux VMs.

Note These are summarized briefly below. See the *vRanger System Requirements Guide* for more information.

- **Linux** - File-level recovery from Linux servers requires the use of the vRanger virtual appliance. See Performing a File-Level Recovery - Linux for more information.
- **Linux** - Not all Linux file systems are supported for file-level recovery. See the *vRanger Systems Requirement Guide* for a list of supported Linux platforms and file systems.
- **Permissions** - vRanger requires you to recover the linux files to an intermediate Windows machine. When you recover linux files to a Windows machine, you will lose the file permissions.
- **Source VM Configuration** - the source VM properties needs to show the OS type as Linux. If this is not configured properly, vRanger will not identify the savepoint as a Linux VM.

Restoring from a Manifest

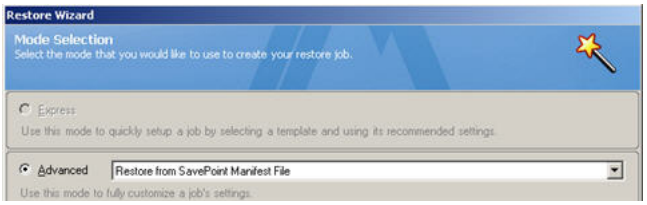
vRanger offers the ability to restore a savepoint after it has been moved from a repository. This allows for greater flexibility when restoring a file or VM after it has been moved to long-term tape backups. You may also use the restore from manifest feature to restore a VM after transporting the savepoint to a secondary location.

Full Restore from Manifest

Each savepoint contains a manifest file that can be used to restore the savepoint without access to the vRanger database. If the vRanger machine is lost, you may install a fresh installation of vRanger, and restore VMs using this manifest file.

Step 1: Create Restore Job

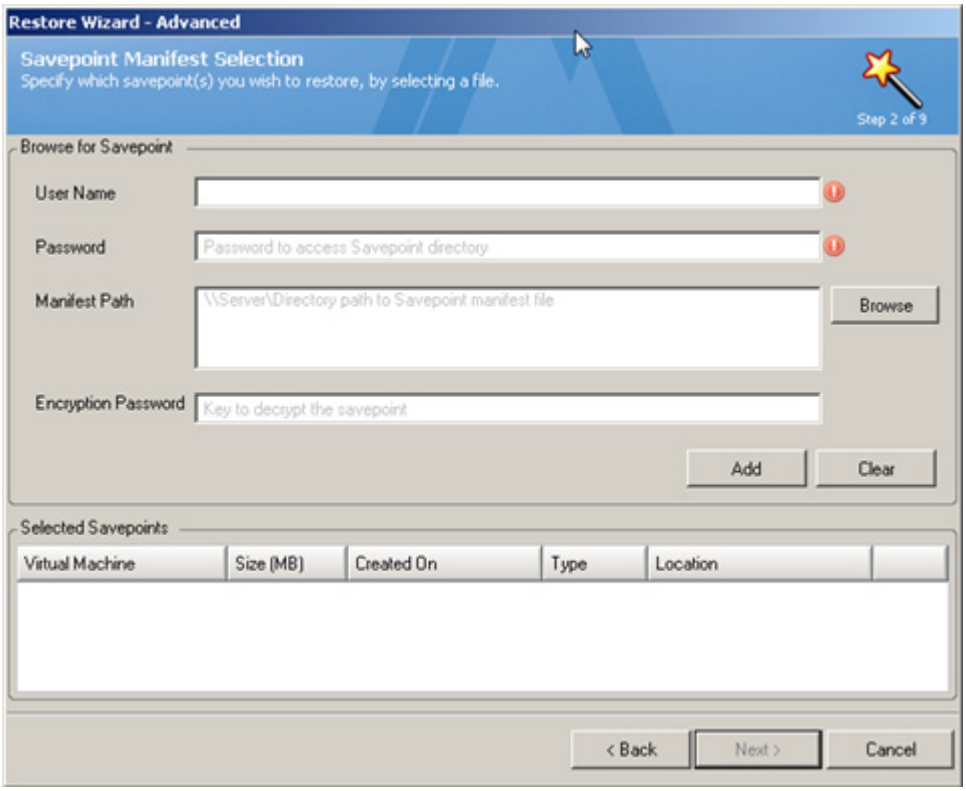
1. From the vRanger toolbar, click **Add->Restore Job**.
2. Select **Restore from Savepoint Manifest.**



3 Populate the **Job Name** and **Job Description fields**. Click **Next.**

Step 2: Select Savepoint

The Savepoint Manifest Selection dialog appears. Use this screen to browse to the location of the manifest on your network, and configure the appropriate authentication credentials for the file.



1 Browse to the location of the savepoint manifest**.**

2 Enter the username and password with access to the savepoint manifest location.

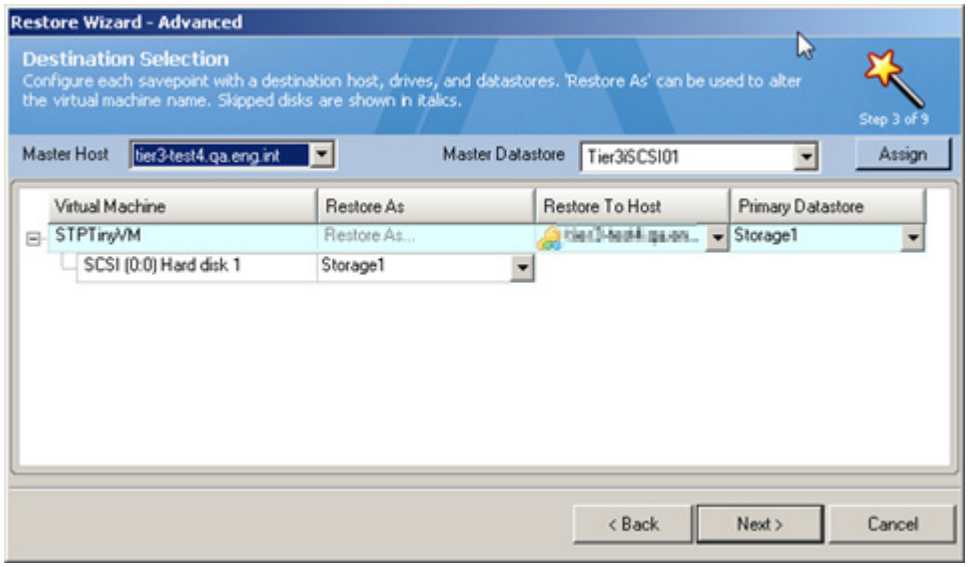
3 **(Optional)** If the savepoint is encrypted, enter the encryption password.

4 Click Add.

5 Repeat steps 1-4 as needed for multiple savepoints. Click **Next**.

Step 3: Destination Selection

The **Destination Selection** dialog displays, which allows you to configure the host, drives, and datastores to which the VMs will be restored.



A datastore can be configured for each VMDK. You can also configure a Master Host and Master Datastore, which will use assigned masters for all VMs and VMDKs displayed. Skipped disks appear in italics on this dialog. Use the **Restore As** field to rename the VM you are restoring.

1. For each VM: In the **Restore To Host** column, select the host to which the VM should be restored.

   or

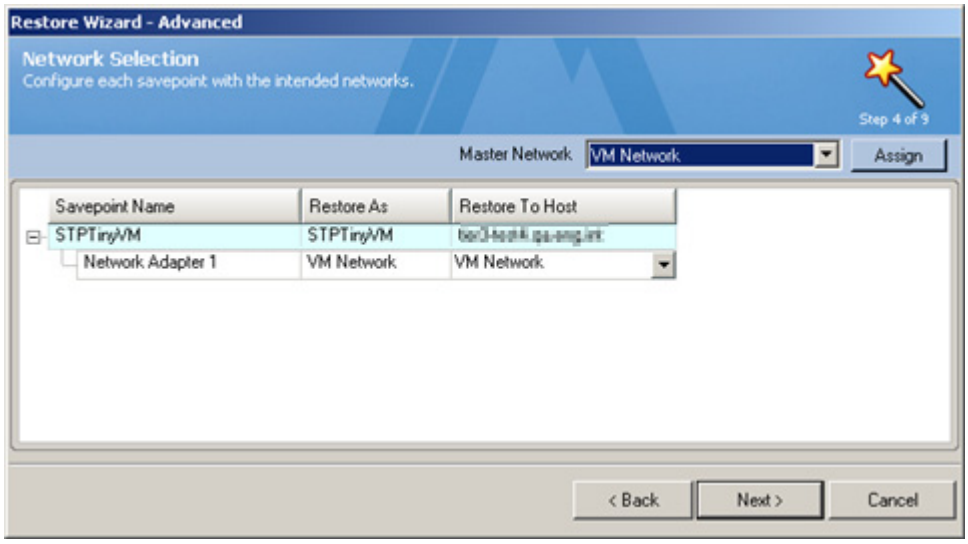
   From the **Master Host** dropdown, select the host to which all VMs should be restored. Click **Assign**.
2. For each VM: In the **Primary Datastore** field, select a value.

   or

   In the **Master Datastore** field, select the datastore to be used for all VMs.
3. For each VMDK: Select the datastore to which each VMDK should be restored.
4. Click **Next**.

Step 4: Network Selection

The Network Selection dialog displays, which allows you to configure the virtual network to which each vNIC will be connected upon restore. You can also configure a Master Network, which will use the selected network for all vNICs.



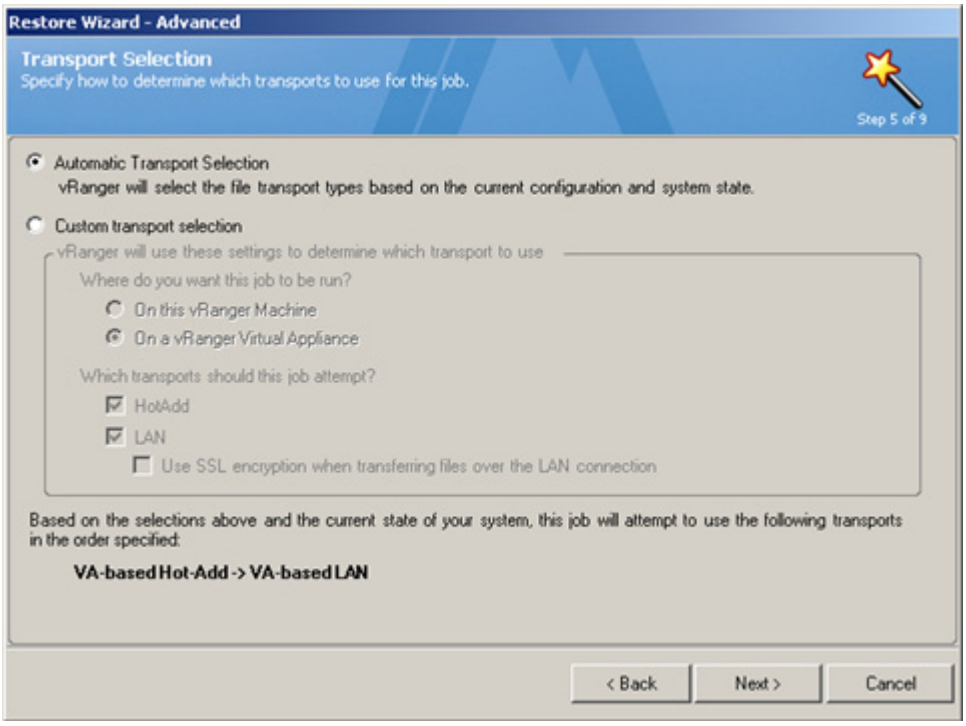
1. For each VM, the connected network adapters will be listed: In the **Restore To Host** column, select the virtual network to which the vNIC should be connected.

   or

2. In the **Master Network** field, select the host to which all VMs should be restored.
3. Click **Assign**.
4. Click **Next**.

Step 5: Transport Selection

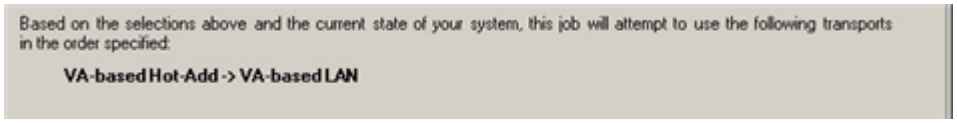
In vRanger, the transport determines how backup data is sent, and where the restore processing activity occurs. See Step 6: Transport Selection for more information on vRanger transports.



Using Automatic Transport Selection

1 On the Transport Selection dialog, select **Automatic Transport Selection**.

The selected transport method will be shown at the bottom of the dialogue:



Important Each time the job is run, vRanger will detect your configuration and select the best transport method available at run-time. That method may not always be what is shown on the dialogue during the job setup.

2 vRanger will check if any virtual appliances are configured on the target host. If the host belongs to a cluster, vRanger will check for shared VAs as well. If no VA is detected, the message below will be shown, as well as the **Configure Virtual Appliance** button, which will allow you to add a virtual appliance to vRanger. See Adding the Virtual Appliance to vRanger for procedures on the Virtual Appliance Configuration dialogue.



3 Click **Next**.

To configure a Custom Transport Selection:

1 In the Custom Transport Section, first select where you want the job to run - on the vRanger machine, or on a virtual appliance.

vRanger will check if any virtual appliances are configured on the target host (the host to which the VM will be restored). If the host belongs to a cluster, vRanger will check for shared VAs as well. If no VA is detected, the message below will be shown, as well as the **Configure Virtual Appliance** button, which will allow you to add a virtual appliance to vRanger. See Adding the Virtual Appliance to vRanger for procedures on the Virtual Appliance Configuration dialogue.

Note Using the vRanger virtual appliance is recommended for scalability and performance.

2 Select which transports the job should attempt. See Backup Transport Options in the "Backup" chapter for more information on the transport options.

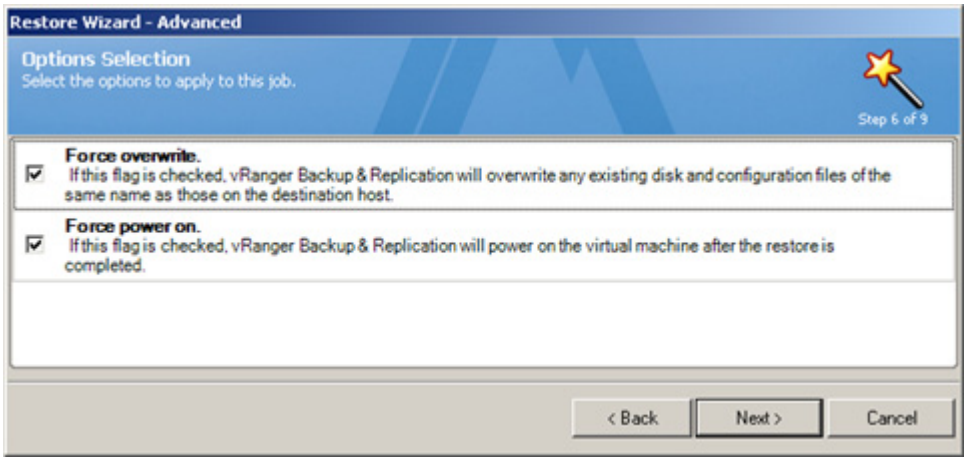
3 [Optional] If you want your LAN restores to be encrypted, select **Use SSL encryption when transferring files over the LAN connection**.

Caution This option will cause reduced performance.

4 Click **Next**.

Step 6: Options Selection

The **Options Selection** dialog displays.



1 Select restore options:

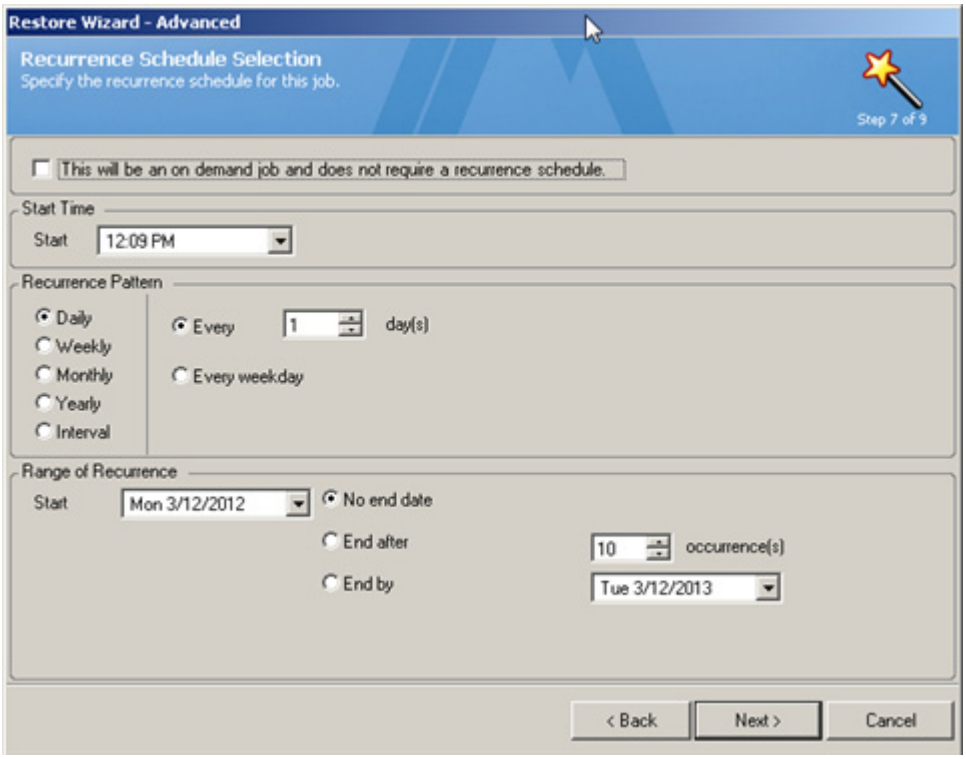
- **Force Overwrite:** This will overwrite any existing disk/config files of the same name as those in the destination host.
- **Force Power On**: This will power on the VM after the restore.

Note **Force overwrite** and **Force power on** are selected as default values.

2 Click **Next**.

Step 7: Recurrence Schedule

The **Recurrence Schedule Selection** dialog displays. On the **Recurrence Schedule Selection** screen, you indicate whether the job should be done on demand. If not, you must define start time, recurrence pattern, and range of recurrence.



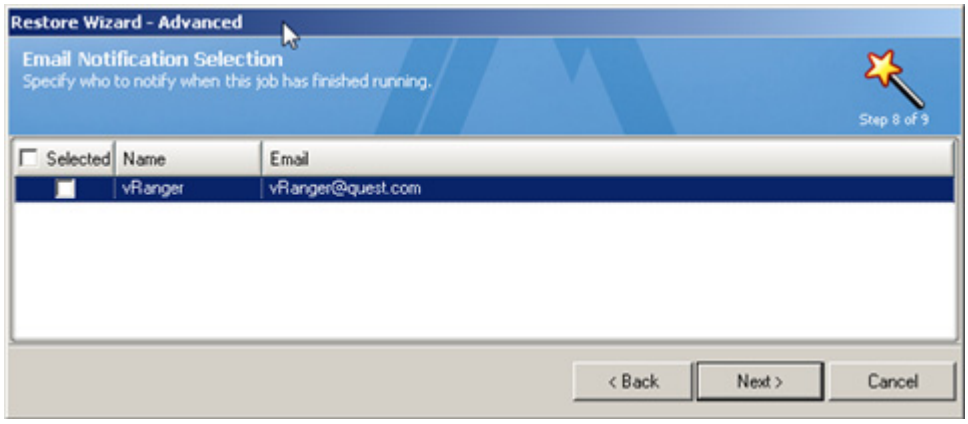
This will be an On Demand job . . .	There is no need to define a schedule. The job will be run on demand.
Start Time	The time of day at which to run the job.
Daily	The daily option can be scheduled to run every weekday or every x number of days.
Weekly	vRanger can be configured to run on weekly intervals, from every week to every 99 weeks. The day of the week on which to run jobs can also be configured.

Monthly	The monthly option offers the following configurations: • **Day** *[x]* **of every** *[y]* **month:** •*x* can be any value from 1 to 31. This determines the day of the month that the job will run. •*y* can be any value from 1 to 99. This value determines the monthly interval--for example, every 2 months sets the job to run every other month. • **The** *[f]* *[d]* **of every** *[y]* **month(s)**: •*f* can be either: first, second, third, fourth or last. •*d* can be: weekday, weekend day, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday. •*y* can be any value from 1 to 99. This value determines the monthly interval--for example, every 2 months sets the job to run every other month.
Yearly	The yearly option offers the following configurations: • **Every** *[m]* *[x]*: • *m* is any month of the year. This determines the month of the year in which the job will run. • *x* can be any value from 1 to 31. This determines the day of the month that the job will run. • **The** *[f]* *[d]* **of** *[m]*: • *f* can be either: first, second, third, fourth, or last. •*d* can be: day, weekday, weekend day, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday. • *m* is any month of the year. This determines the month of the year in which the job will run.
Range of Recurrence	Select the starting date and ending date. • **No end date**: The job will recur until stopped. • **End after**: Select the number of occurrences. • **End by**: Select the date by which to end the recurrence.

1. Configure the schedule options.
2. Click **Next.**

Step 8: Email Notification Selection

The **Email Notification Selection** dialog displays.Through the **Email Notification Selection** screen, you select the email addresses to which notifications should be sent for this job. You can also request that a report be sent to the recipients identified once the job has completed. To configure email notifications, see Set up Mail Server Information/Add Email Addresses.

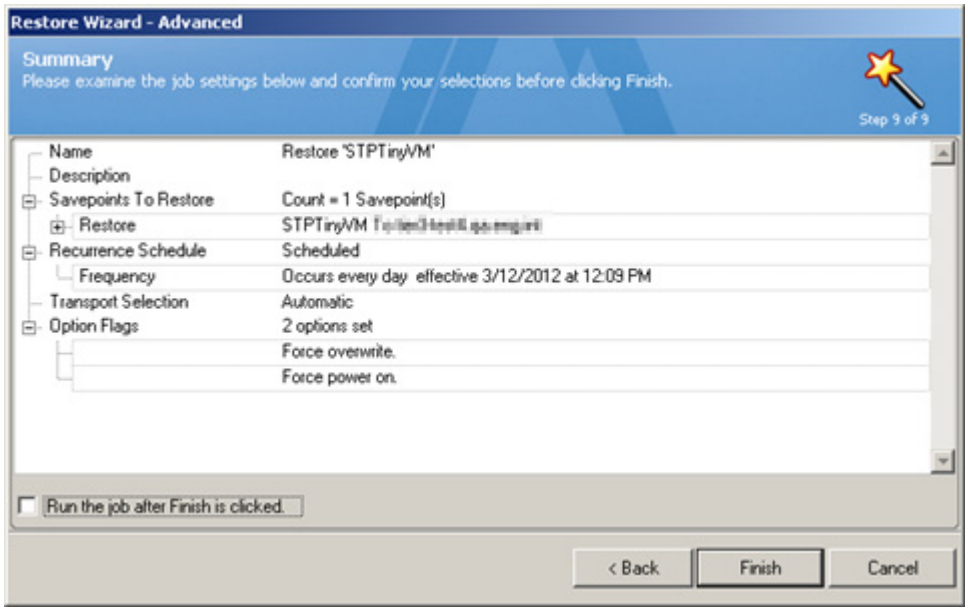


To select an email address:

1. In the **Selected** column, select the address to which notifications should be sent. To select all email addresses, select the check box in the Selected column header.
2. Click **Next**.

Step 9: Summary

The Summary dialog displays. The **Summary Screen** displays a summary of the selections you made during the process.



1. To run the restore immediately, select **Run Restore on Finish**.
2. Click **Finish**.

Step 10: Running the Restore

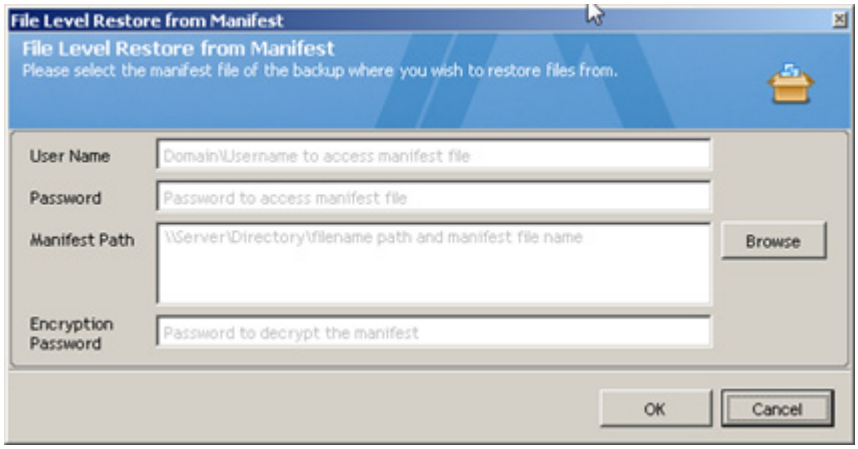
1. If you did not choose to run the restore from the wizard, access the **Working Inventory** pane. Select a job to restore.
2. Click **Run.**
3. The restore will begin immediately.
4. To check progress or confirm completion, access **My Jobs-->Recent Jobs**.

FLR from Manifest

To perform a file-level recovery from a manifest, use the procedure below.

Step 1: Select Manifest File

1. Click **My Repositories.**
2. In the **My Repositories** toolbar, click the **FLR from Manifest Icon** . **The File Level Restore from Manifest** dialogue appears.



3. Enter the user name, password and path for the Manifest file of the backup that contains the files you wish to restore.
4. Enter the **Encryption Password,** if the backup was encrypted.
5. Click **OK.**

Step 2: Select Files

1. In the **Select the Files/Folders to Restor**e pane, expand the archive to view the file structure.
2. Select the file or files to recover.

Step 4: Select Destination

1. In the **Select Destination** pane, select the destination to which the selected files should be recovered.

The **Select Destination** pane shows the Windows Explorer view for the machine on which vRanger is installed.

Caution vRanger will not check for previous versions of the files you are restoring. Files restored using FLR will overwrite any previous files (of the same name) in the restore destination.

2 Click **Restore**.

3 In the **Restore Status** pane, view the status of the restore activity and the value in the Percent Finished column. Status should be Finished and Percent Finished should be 100%. To view the restored file or folder, click the ***View in Explorer*** link.

The user needs the ability restore a virtual machine whose backup files are not in a repository. This allows the user to move the files that make up a backup to wherever they wish and still perform a restore.

Replicate

This chapter describes the Replication options available in vRanger, and gives step by step instructions on configuring replication jobs.

This chapter contains the following sections:This section contains the following topics:

About Replication....145
Replication Modes....148
Creating a Replication Job....153
Failover....162
Failback....165

About Replication

vRanger includes an integrated replication based on the proven technology of vReplicator. There are two options for replication: standard replication using the service console; or replication via virtual appliance (required for replication to or from VMware ESXi). The replication process is the same for each method, except as noted.

A VM is made up of a set of files, which means that replicating a VM is, in essence, replicating the set of files that make up the VM, with changes to these files that reflect user specified settings for the source VM.

The set of files replicated by vReplicator is listed below:

File Extension	Description
.vmx	The VM config file, one per VM
.vmxf	The extended VM config file, one per VM
.nvram	The VM BIO file, one per VM
.vmdk	The VM hard disk file, one per hard disk or snapshot
-flat.vmdk	The VM hard disk data file, one per hard disk or snapshot
-delta.vmdk	The VM snapshot data file, one per snapshot
-ctk.vmdk	The VM hard disk change tracking file when CB is enabled on the disk

Replication Requirements

A continuous connection between source and target sites is required when replication is taking place. Excessive network packet loss could result in replication failure.

Replication will work with links having average packet loss of less than 2%. Replication is not designed to work in replication environments where packet loss can exceed commercially accepted limits.

Networks having 99% Uptime/Availability will generally provide for good Replication performance.

How Replication Works

During the replication process, the configuration files will be created and modified on the target server via the VMware API.

A set of working files is also created and used during the replication process. These files and their purposes are listed below:

File Extension	Description
.vzmap	Records data block offset and hash of files on the target VM. A vzmap file is created for each of the files replicated at the end of the replication. The vzmap file is used by the next replication pass to detect any data changes since the previous pass. It stays on the target VM as long as the job is still configured to run. While relatively small, the size of the vzmap file is directly proportional to the size of the VMDK it is based on. During normal replication, the vzmap file is stored with the replica VM files. For VA replication, the vzmap file is stored on the target VA.
vzundo-script	A script created by the replication process to roll back changes on the target VM in case of a replication failure. This file created in the target VM folder at the beginning of the replication process and removed at the end.
.vzundo	This is a temporary file that records original data of changed blocks since the replication started. One for each VMDK replicated. In case of a replication failure such as network failure, the vzundo-script can be executed to restore files to their original state. These files are created by the replication process and removed when the job is completed. **Note** The vzundo file will be as large as the amount of changed data replicated during a given pass. For example, if a replication pass sends 20 GB of changed data to the target host, the vzundo file will be 20 GB.
.vmdk-abbt.vztemp	Active block filter file. One for each hard disk data file when ABT is enabled. It records active data block offsets for source VM disks. This file is used against the .vzmap file to figure out data blocks that need to be streamed to the target. It is created at the beginning of the replication process and removed when disk replication is completed.
.vmdk-abfg.vztemp	Change block filter file. One for each hard disk when CBT is enabled. It records changed block offsets for source VM disks. This file is only generated when CBT and ABT are both enabled. It will later be combined with the .vmdk-abbt.vztemp file into -flat-map.vztemp and removed.

-flat-map.vztemp	Disk data filter file. One for each hard disk when one of two situations are true: CB is enabled, or both AB and CB are enabled. It contains active/changed data block offsets that need to be compared to the .vzmap file at the target in order to figure out data blocks that need to be streamed to the target. It is created right before file replication starts and removed when file replication is completed
.vzcid	Records target VM disk CIDs at the end of the replication pass.
.vzmutex	Lock file. Indicate ownership of the VM by the current Quest application.

When replication is executed through the virtual appliance, handling of these files will change:

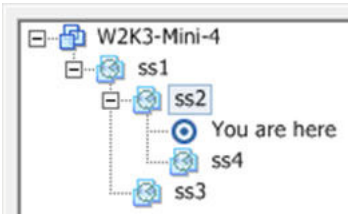
- Working files that only exist during the replication process shall be written to a temporary directory of the VA, since their existence is relevant to the BET on the VA
- Working files that exists across replication processes shall be written to their original location, since these are not truly temporary files and should be reference-able even when the job is modified to not use a VA
- The vzmutex file shall no longer be checked similar to how it is handled in ESXi backup

Replication with User Snapshots

Note This section applies only to replication using the virtual appliance. Service Console replication tasks cannot process user snapshots. If you need to replicate a VM with user snapshots, please use a virtual appliance.

When replicating a VM that contains user snapshots, vRanger will replicate all of the snaphots in the chain from the current snapshot to the base disk. At the target side (the VM to which changes are replicated), the snapshots will be merged into a single disk.

Snapshots not in the chain of the current snapshot will not be replicated. In the image below, ss2 is the current snapshot. Only ss1 and ss2 will be replicated. Snapshot 3 (ss3) is a lateral snapshot to ss2, while ss4 is a child to ss2.



Replication With the Virtual Appliance

vRanger supports ESXi replication via the vRanger virtual appliance. Although the primary purpose of the VA is ESXi replication, you can also use it for ESX replication as well.

Once the virtual appliances are configured and deployed, the use of the virtual appliance is automatic and transparent. Some key points about replicating with the virtual appliance are listed below:

- See the *vRanger Installation and Configuration Guide* for instructions on deploying and configuring the virtual appliances.
- You will need to deploy, at minimum, one virtual appliance on every ESXi server that will be used as a replication source or target. For hosts in a cluster, you may deploy the VA to just one host in the cluster; the VA will be shared among the cluster's other hosts.

  Virtual appliances must be used in pairs. For example, if you are replicating to a target host or cluster that is using a virtual appliance, you must use a virtual appliance on the source host/cluster as well.
- Due to VMware limitations on how snapshots are handled, hardware changes to the source VM occurring after a job has been configured will not be replicated to the target VM. See Replication Modes below.
- Communication between the virtual appliances occurs via an SSH tunnel using AES-256 encryption. See Encryption for more information.

Capturing Hardware Changes with VA Replication

When using the virtual appliances (which rely on HotAdd) for replication, it is not possible to replicate hardware changes that occur on the source VM after the job has been configured. This is due to a limitation in how snapshots are processed by HotAdd.

If you make hardware changes on the source VM, you will need to manually configure the target VM in the same way.

Caution If you add one or more disks to the source VM, you will need to add the disk(s) to the target VM *before* the next replication pass. You will also need to edit the job to include the new disks.

Replication Modes

vRanger offers two different replication modes: Hybrid, and Differential. Each replication mode works in a particular way. Regardless of which replication method is chosen, only the changed data will be replicated on each replication pass.

- Differential -See Differential Replication
- Hybrid - Hybrid Replication. Hybrid replication is not compatible with the vRanger virtual appliance.

Each of these replication modes has the option of Active Block Mapping (ABM). VM replication in general starts with replicating the source VM to the target host. Changes are applied to the target VM at user designated intervals to keep the target in sync with the source. Thus the key areas of difference among the replication modes are:

- Base target VM creation
- Detection and application of changes to the base target VM

- The rest of the section describes each replication mode in more detail.

Replication With Changed Block Tracking (CBT)

Replication with CBT is the recommended method of replication whenever possible.

When enabled, CBT will track the disk block changes made by the source VM. CBT, once enabled on the source host, records the blocks that have changed since the last replication pass and transfers them to the target host without scanning the VMDK. CBT is only available when replicating from ESX 4 host to ESX 4 host.

Note CBT must be enabled for each VM that you want to replicate. vSphere supports CBT, and most VMs running in this environment can utilize it. The VMs must be Hardware Version 7, and must be hosted in ESX(i) 4.0 with updated VMware tools.

Differential Replication

In differential replication, the source VMDK is scanned. A data map is created during the first replication pass. When the first snapshot is taken, disk writes are suspended temporarily. The snapshot remains open. An empty virtual disk-called the delta file is added. As data is replicated, this file grows. These changes are merged back into the source VMDK and the snapshot is closed. During the second replication pass, a snapshot is opened. The source VM's current data is compared to the changed data stored in the data map. These changes are replicated, block by block, and committed to the destination machine. Then, the snapshot is closed and the VMDK is opened. The entire source VM image is replicated to the destination machine on the first pass, with only changed data replicated on subsequent passes.

Note The scanning of the VMDK takes approximately 1 minute per GB. When configuring differential jobs for large VMs, take scanning time into account.

Hybrid Replication

Using the hybrid approach, replication is done as a function of change over time rather than change to data. Snapshots can be taken at shorter intervals with no loss of data and no need to scan during each replication pass. The differentials engine re-synchronizes the source and destination VMs selectively. Therefore, there is no need to replicate the entire source image again, just to capture the data that changed between snapshots.

Note Hybrid replication is not supported for ESXi, nor for ESX when using the Virtual Appliance.

During hybrid replication, two snapshots are created. One snapshot remains open continuously. As time passes, the second snapshot fills with data. Ultimately, this data is copied to the destination VM and then merged into the first snapshot.

This method allows for a very fast process, in part because no scanning is required. Nonetheless, the continuously open snapshot can decrease I/O temporarily. Because this method replicates as a function of change over time, snapshots can grow rather large and cumbersome. For this reason, it is important to confirm sufficient free disk space in advance of initiating a hybrid replication.

During hybrid replication, a snapshot remains open continuously. This can compromise vRanger's ability to complete differential backups of a VM being replicated. When you initiate a hybrid replication, you should

not use vRanger to perform a differential backup of that VM. In this case, it is best to complete a full backup of the machine.

Hybrid Limitations

Hybrid replication is intended for those that can't take advantage of replication with CBT (pre-ESX 4.x environments, for example), and for workloads that require shorter replication intervals that differential replication will allow. If your VMware environment supports replication with CBT, that is the recommended option.

Some limitations of hybrid replication are:

- Hybrid replication is not supported with the vRanger virtual appliance. ESXi replication requires the VA, which means that Hybrid replication is not compatible with ESXi. If you choose to replicate from an ESX host using the virtual appliance, hybrid replication is not an available option.
- Hybrid replication, as an alternative to CBT, is not compatible with CBT.
- During hybrid replication, ABM is only used on the first pass and any re-synchronization. During hybrid replication, snaphots containing only changed data are sent from the source to the target, making ABM unnecessary.

Active Block Mapping (ABM)

ABM filters deleted data blocks so that only active blocks are scanned and streamed to the target. White space detection eliminates the need to compress, stream, and write zero blocks during the replication process.

Choosing a Replication Method

While the choice of Hybrid or Differential replication should be made for each VM based on the RPO, RTO, IO characteristics of the VM and other requirements, there are some general rules of thumb that assist with determining which method of replication to use for each VM:

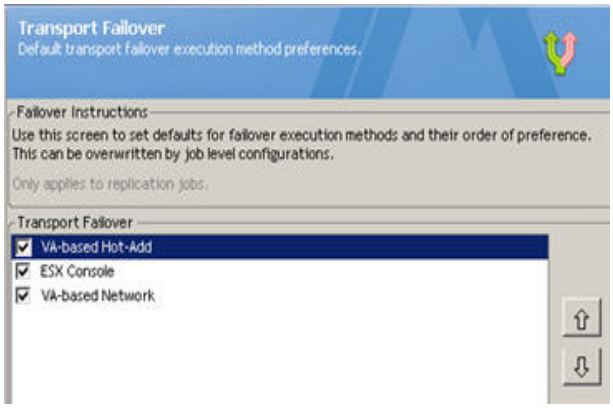
VM Size	Replication Interval	Recommended Replication Settings
Small (< 20 GB)	Any	Use CBT if your virtualization platform supports it. If not, consider using Differential. Small VMs are replicated very efficiently using Differential replication.
Medium- Large (20-150 GB)	Frequent (e.g., once per hour)	Use CBT if your virtualization platform supports it. If not, consider using Hybrid. By using Hybrid replication, replication intervals can be very close due to the elimination of scan time.

Medium- Large (20-150 GB)	Infrequent (e.g., once per day)	Use CBT if your virtualization platform supports it. If not, use Differential. Using differential replication ensures that snapshots will not grow throughout the day, as they would if using Hybrid.
Very Large (500 GB+)	Any	Use CBT if your virtualization platform supports it. If not, use Hybrid, and replicate frequently. Hybrid ensures the elimination of scan time, and replicating frequently ensures that snapshots do not grow to an unmanageable size. While large VMs can be replicated less frequently, ensure that you have an adequate replication interval to replicate the entire VM.

Configuring Transport Failover Options

Note vRanger 5.4.1 uses the custom replication transport option only. The transport failover configuration will have no effect on vRanger 5.4.1. See Enabling/Disabling The Custom Transport Option for more information.

Replication transport actions can occur either in the Service Console or via HotAdd (Lan-Free or Network). If one transport method is not available, vRanger will attempt to perform the replication task using one of the other transport methods. You may configure the order in which vRanger will attempt the transport options with the Transport Failover node of the Configuration Options dialogue.



The order in which you should sort the transport options depends on what type of host you have. The VA-based Network option is the slowest of the three, and should be configured last.

- ESX - The recommended transport failover order for ESX hosts is **VA-based Hot-Add> ESX Console> VA-based Network**.

- ESXi/ESX 5 - For hosts without a service console, configure the transport order to **VA-based HotAdd> VA-based Network**.

Enabling/Disabling The Custom Transport Option

During the standard replication process, vRanger passes vCenter credentials to the virtual appliance, which then uses VMware's VDDK transport to communicate with vCenter in order to perform operations such as mounting, reads, and writes of disks. In environments where the virtual appliances do not have the required access to vCenter, a custom transport must be used.

The custom transport in vRanger 5.4.1 mounts target disks to the virtual appliance through vRanger via the vSphere API. Read and write operations are then performed on the VA as a device instead of by using the VDDK transport.

The Custom Transport option is configured via the configuration file for the vRanger service - *Vizioncore.vRanger.Service.exe.config*. This file is located in the Service sub-directory of the vRanger installation directory. By default this is :

C:\Program Files\Quest Software\vRanger\Service

Caution Change only the values listed below unless directed otherwise by Quest Support.

To configure custom transports:

Step 1: Opening the Configuration File

1. Open the *Vizioncore.vRanger.Service.exe.config* with your text editor of choice.
2. Search for "UseCustomTransport:". This is set to "true" by default, which enables the custom transport.

```
<add key="UseCustomTransport" value="true"/>
```

3. To disable custom transports, set the value to "false".
4. Save the file and exit.
5. Restart the vRanger service in order for the changes to take effect.

Pre-seeding Replication Jobs

Ranger replication is intended to replicate changes from a source VM to a remote target. It is often not practical to perform the first replication pass (which sends the full VM) to a remote site over a WAN link. You may use vRanger to "seed" a replication job locally in order to reduce the amount of data sent over the WAN.

Backing up the Source VM

1. If vRanger is not already installed please do so.
2. Select the VM you wish to get to the DR site and create a Full backup job.
3. Backup the VM locally.

4 Copy the backup savepoint, and move it to the DR site. This is usually accomplished by shipping a USB drive or some other media containing the backup archive.

   Savepoints are named using a *VMname_date_time_UUID* format. Copy the entire directory (containing a .var file, and two .metadata files) for the savepoint in question. The example below shows this directory highlighted in red.



5 Using the Restore from Manifest procedure documented in Performing a File-Level Restore - Windows, restore the backup to the target ESX Server at the DR site. Make sure the name of the restored VM is the same as the source VM.

Replicating the Changes

1 Launch vRanger and expand the host where the source VM is located. Select the VM.

2 Click **Add Replication Job**.

3 Configure a replication job with the desired options.

4 When selecting the target host, select the host on the target side to which the source VM has been restored.

   Note When selecting the target VM name, ensure that you are using the same name as the source VM. vRanger will append "_VzReplicate" to the VM name on the target side.

5 In the **Preseed VM UUID** field of the **Select Target Host** dialog, click the browse button and select the preseeded VM.

6 When configuring the primary datastore, ensure it is the same datastore as the restored VM and click **Finish**.

7 Run the replication job. Only the data changes will be passed over the WAN as opposed to the full VM.

Creating a Replication Job

vRanger uses a simple wizard to guide you through the process of creating a replication job. At a high level, the process includes the following steps. Each of the steps below is explained in more detail in the sections that follow.

- the VM you wish to replicate
- a specific name for this replication job
- the Target Host
- the method (differential or hybrid)
- the Network Mapping you desire
- the VM Hard Disk you wish to replicate
- the reccurrance (if any)

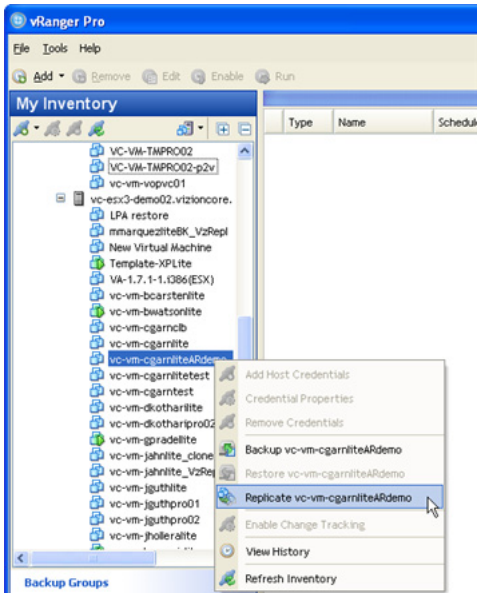
- the person you wish to be notified when the replication is completed.
- and from the following Options: replicate powered on machines only, check the destination for free space, use compression, update the VM with the replication results, guest quiescing, and ABM.

See the sections below for procedures on creating a vRanger replication job.

Step 1: Select Source VM

1 Click **My Inventory.**

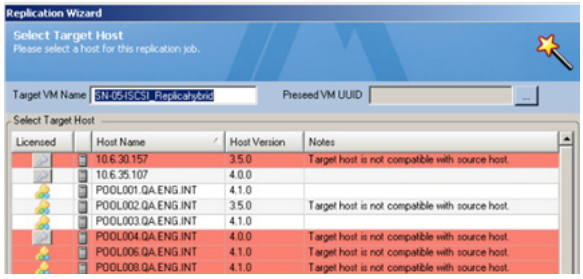
2 Right click the VM you wish to Replicate, and select **Replicate.**



3 Enter the **Job Name** and **Description**, then click **Next**

Step 2: Select Target Host

The Select Target Host dialogue appears.



1 Choose a **Target Host** from the list, then click **Next**.

or

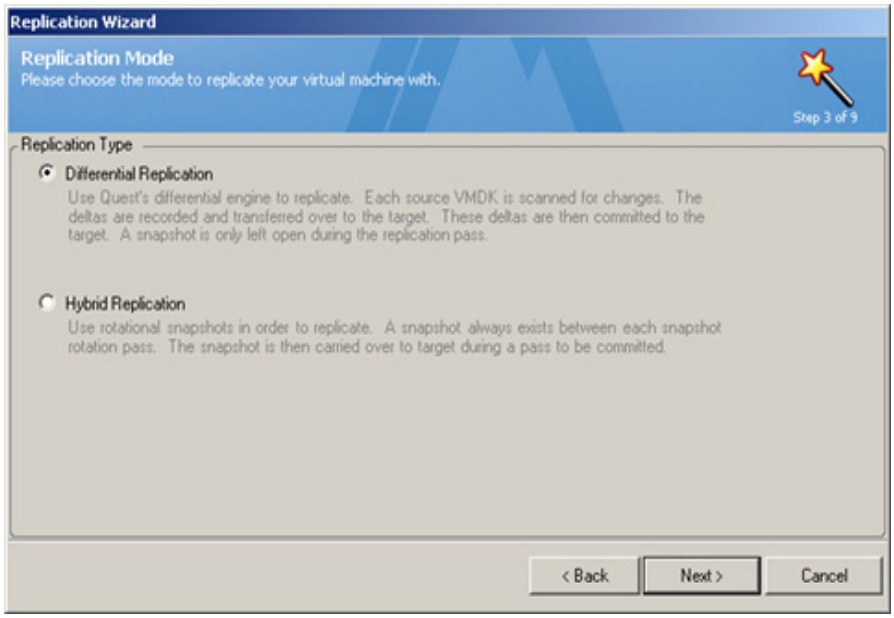
If you have pre-seeded the VM on the target host, select the host on the target side to which the source VM has been restored. See Pre-seeding Replication Jobs for more information.

Note When selecting the target VM name, ensure that you are using the same name as the source VM. vRanger will append "_VzReplicate" to the VM name on the target side.

2 **[Optional]** In the **Preseed VM UUID** field of the **Select Target Host** dialog, click the browse button and select the preseeded VM.

Step 3: Replication Mode

The Replication Mode dialogue appears. In this screen, you can select between differential replication and hybrid replication. See Replication Modes for more information.

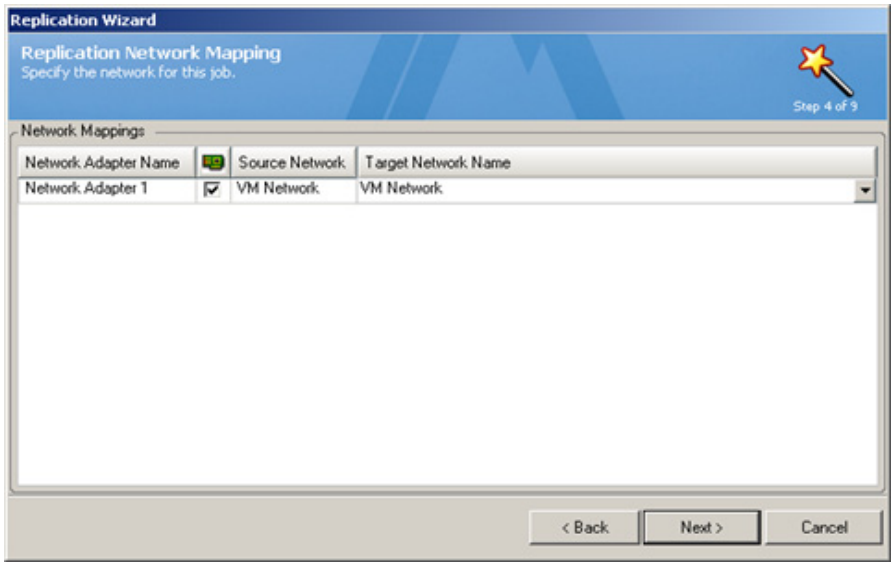


1 Click **Differential** or **Hybrid**, then **Next.**

Note Hybrid replication is not supported with the virtual appliance.

Step 4: Replication Network Mapping

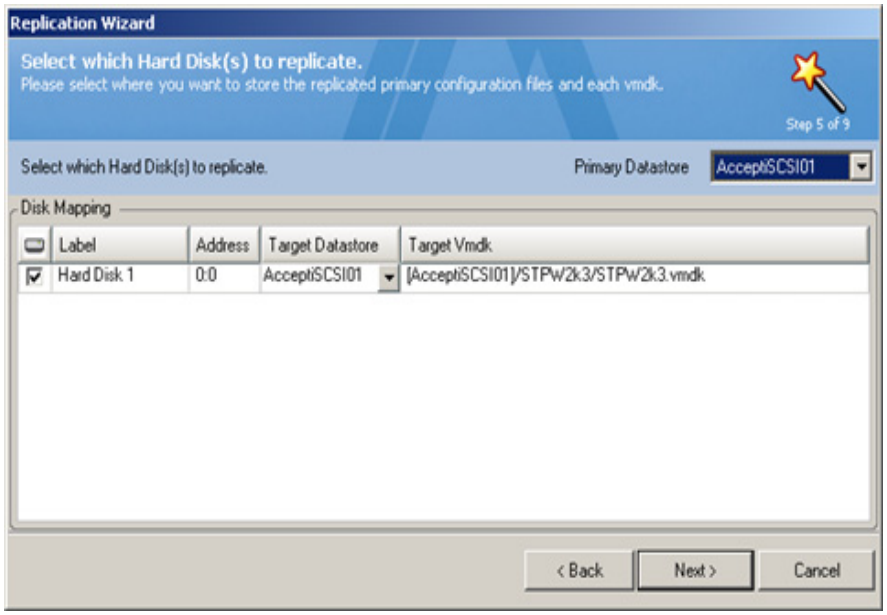
In the Replication Network Mapping dialogue, you specify the network on the target host to which the VM nics should connect.



1 Select the **Network Mapping** you wish to use for this job, then click **Next**.

Step 5: Hard Disk Selection

The Hard Disk selection dialogue appears.



1 Select the Hard Disk(s) you wish to replicate

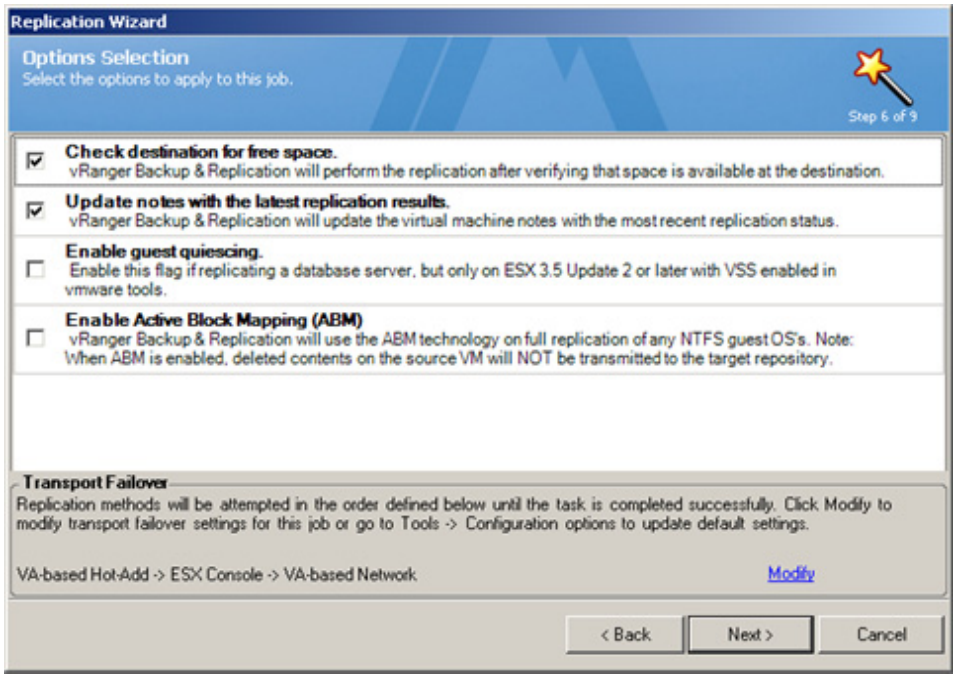
2 You may select a **Primary Datastore,** which will be used as the default value for all disks in this job, allowing you to configure only the exceptions.

3 In the **Target Datastore** column, verify that the Primary Datastore has been populated. If you want the disk to replicate to a different datastore, change it in this column.

4 Click **Next**.

Step 6: Options Selection

The Replication Options dialogue allows you to configure additional operations performed during the replication job.You may also view the transport failover preferences, and modify them as needed.



1 Configure the replication options, using the table below for guidance.

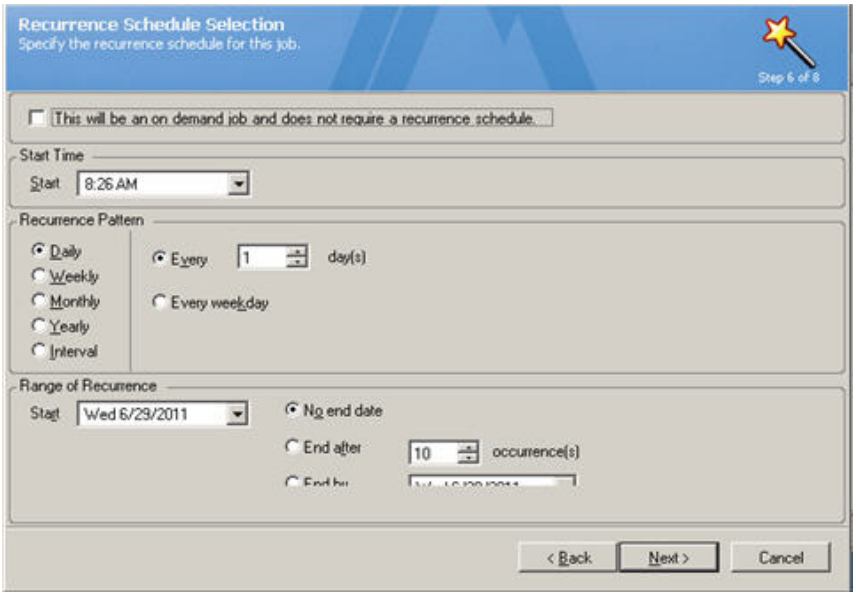
Option	Description
Check destination for free space.	This option prompts vRanger to check the destination location to ensure that there is enough space available for the replica VM.
Update notes with latest backup results.	If this flag is selected, vRanger will update the VM Notes in VC with the most recent replication task status. **Note** vRanger maintains the existing VM notes in vCenter, appending the replication status to the end.

Enable guest quiescing.	This option enables the VMware Tools when creating a backup. Use this flag if you are replicating a database server. During this process, vRanger freezes or pauses writes on supported applications to provide a consistent image of VSS-enabled databases. See Step 6: Retention Policies & Space Saving Technologies in the "Backup" chapter for more information.
Enable Active Black Mapping	Checking this box enables Active Block Mapping, which allows vRanger to skip zeroed out blocks and junk data. **Note** With this option enabled, deleted data on the VM will not be backed up.

2 Verify your Transport Failover preference. See Configuring Transport Failover Options for more information. To change the Transport Failover order, click **Modify.** Otherwise, click **Next.**

Step 7: Recurrence Schedule

On the Recurrence Schedule Selection screen, you define start time, recurrence pattern, and range of recurrence for the replication job. You may also configure the job as **On Demand**, which has no schedule.



1 Configure the schedule, using the table below for guidance.

Note To replicate more frequently than once per day, use the **Interval** reccurrance pattern.

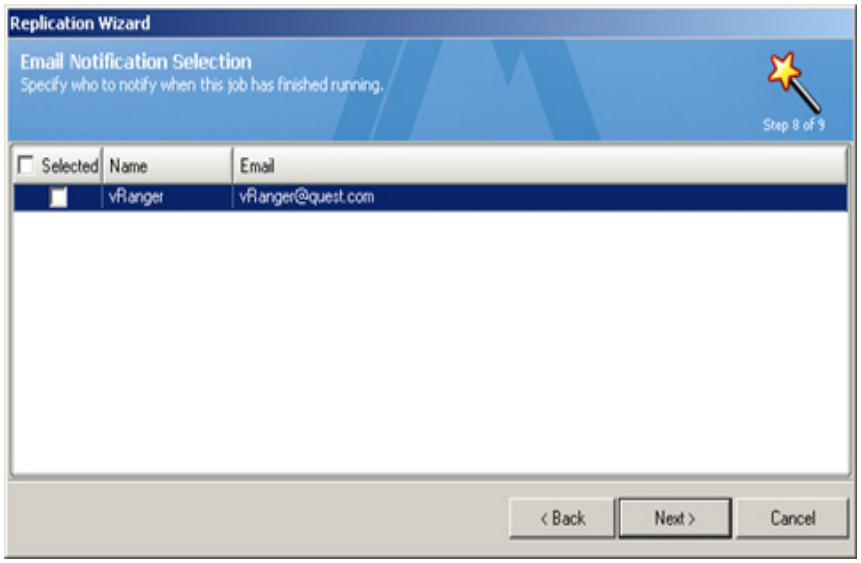
This will be an On Demand job . . .	There is no need to define a schedule. The job will be run on demand.
Start Time	The time of day at which to run the job.
Daily	The daily option can be scheduled to run every weekday or every x number of days.
Weekly	vRanger can be configured to run on weekly intervals, from every week to every 99 weeks. The day of the week on which to run jobs can also be configured.
Monthly	The monthly option offers the following configurations: • **Day** *[x]* **of every** *[y]* **month:** •*x* can be any value from 1 to 31. This determines the day of the month that the job will run. •*y* can be any value from 1 to 99. This value determines the monthly interval--for example, every 2 months sets the job to run every other month. • **The** *[f]* *[d]* **of every** *[y]* **month(s)**: •*f* can be either: first, second, third, fourth or last. •*d* can be: weekday, weekend day, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday. •*y* can be any value from 1 to 99. This value determines the monthly interval--for example, every 2 months sets the job to run every other month.

Yearly	The yearly option offers the following configurations: • **Every** *[m][x]*: • *m* is any month of the year. This determines the month of the year in which the job will run. • *x* can be any value from 1 to 31. This determines the day of the month that the job will run. • **The** *[f][d]* **of** *[m]*: • *f* can be either: first, second, third, fourth, or last. • *d* can be: day, weekday, weekend day, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday. • *m* is any month of the year. This determines the month of the year in which the job will run.
Interval	The Interval option offers the following configurations: • **day(s)**- the default value is 0, which allows you to replicate more frequently than daily. • **hour(s)**- the default value is "1". To replicate more frequently, change to "0" and configure a "minutes" value. • **minute(s)**- the minimum interval is 5 minutes. **Note** Configure a value sufficient to allow the previous replication pass to continue.
Range of Recurrence	Select the starting date and ending date. • **No end date**: The job will recur until stopped. • **End after**: Select the number of occurrences. • **End by**: Select the date by which to end the recurrence.

2 Click **Next**.

Step 8: Email Notification Selection

Through the **Email Notification Selection** screen, you select the email addresses to which notifications should be sent for this job. You can also request that a report be sent to the recipients identified once the job has completed. To configure email notifications, see Set up Mail Server Information/Add Email Addresses.

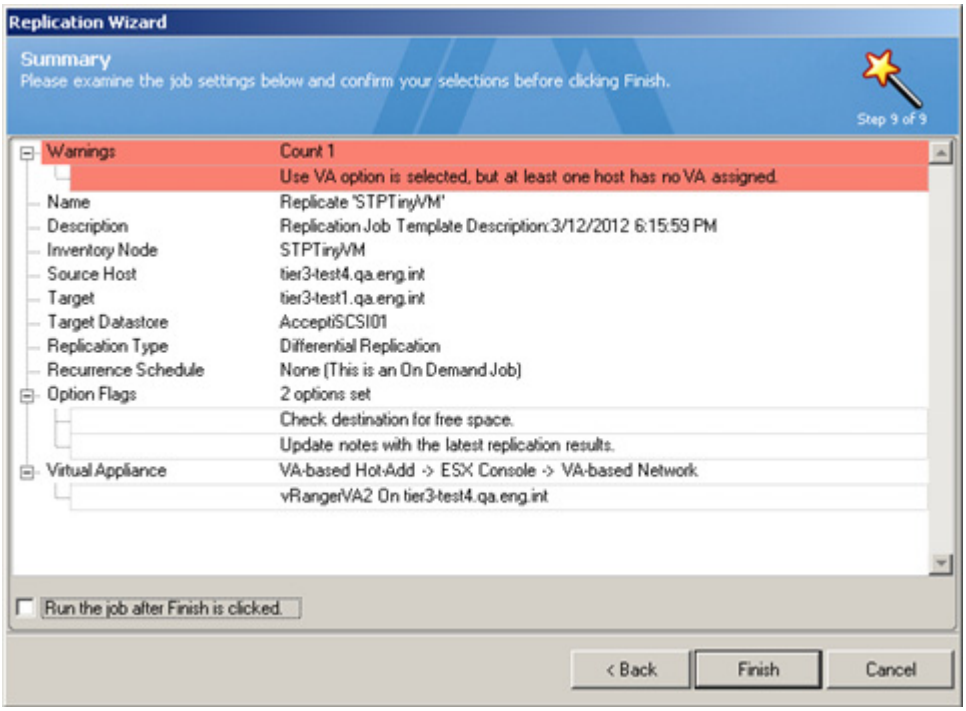


To select an email address:

1. In the **Selected** column, select the address to which notifications should be sent. To select all email addresses, select the check box in the Selected column header.
2. Click **Next**.

Step 9: Summary

The final screen in the **Replication Wizard** displays a summary of the selections you made during the process.



1 If your host configuration (source or target) does not match your configured options, you will receive a red indicator, as shown above.

Note The most common configuration error is configuring a Transport Failover sequence that does not match your hardware. For example, if you have VA-HotAdd selected in the Transport Failover section, but one of your hosts does not have a VA configured.

2 Review the job settings, select **Run the job after Finish is clicked** if you wish, and click **Finish**.

Failover

There are two kinds of replication Failover: Test Failover, and Failover.

- **Test Failover:** this function will disable the replication job, snapshot the replica VM,, and power it on. You have the option of performing a test failover with or without networking enabled on the replica VM.
- **Failover:** this manually initiated process simplifies the action of failover to a replica VM. You will have the option to perform a final synchronization, or to failover immediately. The vReplicator process gracefully and safely powers down your production VM, and boots your replica VM with a single click.

Test Failover

The Test Failover operation can be performed from the **Scheduled Jobs** and **On Demand Jobs** nodes of the My Jobs view.

The default Test Failover operation is described below. To perform a test failover with networking enabled, refer to Network-Enabled Test Failover.

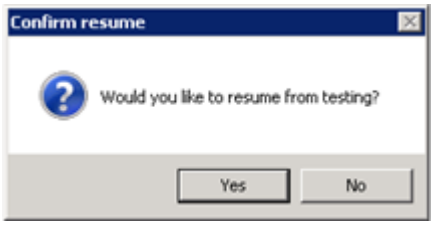
1 Click the **Test Failover**  icon. A confirmation message displays.

2 Click **Yes.**

The test failover process begins, and a job is added to the Current Jobs node.

3 When testing is complete, select the Test Failover task in the Current Jobs node, and click **Resume**.

4 Click **Yes** when the confirmation prompt appears.



- The snapshots made will be reverted.
- The network card will be “Connect at Power On”. This prepares the replica VM for failover.

- The VM will return to it's dormant state.

Network-Enabled Test Failover

Due to this option's potential for negatively impacting your production environment, it is disabled by default. To enable this feature, perform the steps described in Enabling Network Enabled Test Failover below.

Enabling Network Enabled Test Failover

This feature is enabled via the configuration file for the vRanger client - *Vizioncore.vRanger.Client.Shell.exe.config*. This file is located in the Client sub-directory of the vRanger installation directory. By default this is:

C:\Program Files\Quest Software\vRanger\Client

1. Open the *Vizioncore.vRanger.Client.Shell.exe.config* file with your text editor of choice.
2. Carefully make the required changes, using the information below as guidance.

   Note Not all text editors include line numbers. If you do not see the line numbers referenced below in your text editor, this is not an indication of a problem.

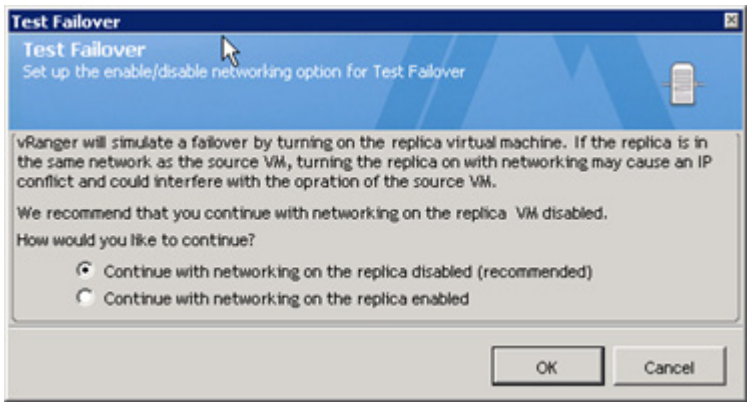
3. Scroll to line 38or search for "`Test Failover`".
4. Change the value for `ShowTestFailoverDialog` from "false" to "true", ensuring the quotation marks remain.
5. Save the file and exit.
6. Restart the vRanger Service for the configuration change to take effect.

Performing a Network-Enabled Test Failover

Once enabled, you will be prompted when performing a test failover; you can do so with or without networking enabled during the test.

To perform a test failover:

1. Click the **Test Failover**  icon. The Test Failover dialogue appears.



2. Select the desired option, as described below:.

- **Continue with networking on the replica disabled**-- when performing a test failover with networking disabled, the replica VM will be powered on with the network adapter disconnected. This protects the production environment, but eliminates the ability of testing your replica VM as part of a larger environment.
- **Continue with networking on the replica enabled**- when performing a network enabled test failover, your replica VM will be powered on with the network adaptor connected.

  Caution This has the potential to impact your production environment - exercise caution when using this option.

3 Click **OK**.

4 When testing is complete, select the Test Failover task in the Current Jobs node, and click **Resume**.

- The snapshots made will be reverted.
- The network card will be set to "Connect at power on".This prepares the replica VM for failover.
- The VM will return to it's dormant state.

Failover

If the production site becomes unavailable, or you need to move the workload to the DR site, you have the option of performing a failover operation from the vRanger interface.

A failover operation consists of the following steps:

- The production VM will be powered off.
- **[Optional]** Perform a final synchronization between sites. This ensures that no data is lost during a failover. If you do not want to synchronize changes (in the event of a virus, perhaps) or cannot (in the event of a hardware failure), data changed since the last replication pass will be lost.
- The D/R VM will be powered on.
- The replication job will be reversed, with the D/R VM now acting as the source.

  Note The first replication pass after failing over will send the entire image, not just the changed data.

To perform a failover:

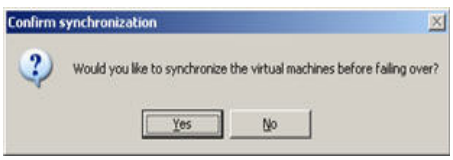
1 In the My Jobs view, select a replication job.

2 Click the **Failover** 

icon.

A confirmation message displays. Click **Yes.**

3 A **Confirm Synchronization** dialog appears.

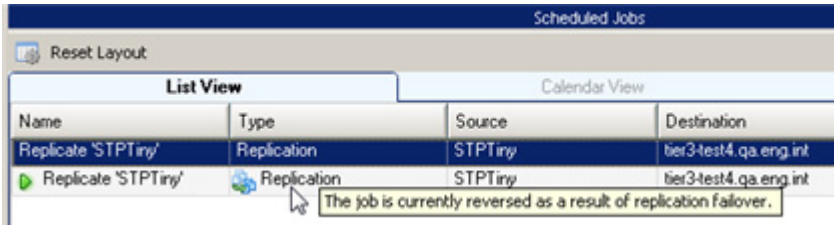


- **Yes** - this will shutdown the production VM and perform a final synchronization to the D/R site. This will ensure that no data is lost, but will also transfer any virus or corruption to the replica VM. View the replication progress in the Running Tasks section of the vRanger UI.

  Note If you are using differential replication, this will cause vRanger to perform a complete scan of the VM disks (as it does for every replication pass), but only changed data will be sent to the replica VM.

- **No** - The production VM will be powered off, but no changes will be sent to the D/R replica.

4 After the Failover completes, the Destination of the original replication job is now the Source, and the original Source is now the Destination. In other words, the job is now set to replicate from the D/R site to the Production site. This change is not reflected in the vRanger UI.Jobs that have been failed over will be indicated by a failover icon and tool-tip message:



5 If the production site is unavailable, or you do not want to replicate changes, disable the replication job.

Failback

When the production server has been returned to proper operation, you will usually need to failback the replicated VM to the production site. A failback is simply a failover back to the production site.

Caution When performing a failback operation, the first replication pass to the production site will send the entire VM image.

A failback operation includes the steps below:

- The D/R VM will be powered off.
- **[Optional]** Perform a final synchronization between sites. This ensures that any data changed on the D/R VM will be replicated back to the production site. If you do not want to synchronize changes data changed since the last replication pass will be lost.
- The production VM will be powered on.
- The replication job will be reversed (returned to the jobs original configuration), with the D/R VM now acting as the target.

- If the job is enabled, replication will continue, sending changes from the source production VM to the D/R target.

  Note The first replication pass after failing back will send the entire image, not just the changed data.

To perform a failback:

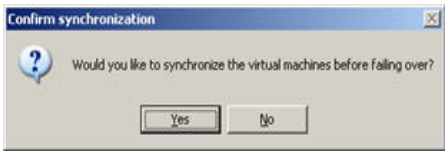
1 In the My Jobs view, select a replication job.

2 Click the **Failback** 

icon.

A confirmation message displays. Click **Yes.**

3 A **Confirm Synchronization** dialog appears.



- **Yes** - This ensures that any data changed on the D/R VM will be replicated back to the production site. If you do not want to synchronize, data changed since the last replication pass will be lost. View the replication progress in the Running Tasks section of the vRanger UI.
- **No** - The D/R VM will be powered off, but no changes will be sent to the Production site.

4 After the Failback completes, the direction of replication will again reverse, this time returning to the original configuration. The Failover icon will no longer be shown.

  Note The first replication pass to the D/R site will send the full VM image. Subsequent passes will send only changes.

Reports

This chapter contains information about reporting functionality in vRanger.

This chapter contains the following sections:This section contains the following topics:

Reports 168
My Reports View 168
Report Wizard 170

Reports

The My Reports view is a gateway to creating, editing, and distributing custom reports. There are several report types:

- Backup Job Summary Report
- Backup Task Report
- Replicate Job Summary Report
- Replicate Task Report
- Restore Job Summary Report
- Restore Task Report

As you create reports, the documents will nest under the node of the relevant report type. When you create a backup job, you can configure a report to display in the body of an email. The report will indicate which jobs were run, which succeeded, and which failed. You can also schedule a report to run, save it as a PDF, and send it as an email attachment.

Report information is organized by VM and includes when backups, replicates and restores were run, who ran them, and the status of each. You can create a report for results from one or multiple VMs. Reports can contain customized metrics--for example, date range (day, month), status (successful, failed, stopped). You can run reports as soon as you create them or you can schedule them to be generated later.

My Reports View

The **My Reports** view is made up of three areas--**My Reports** on the left, **Working Reports** on the right, and the **Report Template Detail** area on the lower right.

My Reports Pane

This area features several nodes in a tree structure--**Backup Job Summary Report, Backup Task Report, Replicate Job Summary Report, Replicate Task Report, Restore Job Summary Report,** and **Restore Task Report**. Under each of these nodes, all reports that you create and save will be nested by type. At the top of the pane, there is a menubar and toolbar. Within the pane itself, backup and restore reports are designated different icons.

Main Toolbar

	**Print**: Click this icon to print the report displayed in the **Working Reports** pane.

	**Print Preview**: use this command to review the report before printing it.
	**Page Setup**: use this command to make adjustments to the report before printing it.
	**Save**: this dropdown gives you access to two options--PDF and Excel. When you select a value, the Save As window opens. For the selected report, you can create a PDF or an Excel spreadsheet.
	**Refresh**: this causes recently added/removed reports to be correctly displayed.
	**Close Report**: Closes the reports currently displayed in the Working Reports pane.

Pane Toolbar

	**Add**: This icon allows you to add any of the four reports types using the Restore Wizard: • Backup Job Summary Report • Backup Task Report • Restore Job Summary Report • Restore Task Report When you first open the **My Reports** view, this is the only icon that is enabled.
	**Edit**: Opens the **Report Wizard**. Configurations for the currently selected report display. You can make changed to the settings and save the s.
	**Remove**: If you click this icon, the report that is currently selected in the **My Reports** pane will be removed.
	**Run**: With a reports selected in the **My Reports** pane, click this icon to run it.

Working Reports Pane

In this pane, report listings display. The name of the reports is at the top of the pane. The format of the text that displays depends upon how you configured using the **Report Wizard**.details that you included when you created the report. For example, if you added color or pagination to the report, those settings will be in effect in this pane.

Report Wizard

You can access the **Report Wizard** through the **My Reports** view two ways. You can right-click on one of the four main nodes and select **Add** or click to highlight one of the nodes and then click the **Add** icon in the toolbar.

Through the **Report Wizard**, you can generate reports that are filtered by date/time and status. For example, you can generate a report that includes only backup or restore tasks or jobs that failed over the last 10 days.

Report Filter Dialog

Date Range	Narrow report data using these fields: • **Over the last**: Enter the number in the first field and select Days or Months from the dropdown. Default is 30 days. • **Between**: Click this radio button to enabled two dropdown date fields. Click each to display a calendar from which you can select a date range.
VM/Host/ Repository	**VM**: Click a radio button--either All or Select VMs-- to restrict report data. **Host**: From this dropdown, you can select All or any individual ESX host. All licensed, available hosts are listed here. **Repository**: This dropdown features three values from which to select: All, cifs, or sftp.
Duration	In this section, you can enter a number in the Greater than or equal to __ minutes field. The value you enter will restrict the report to include only jobs that were run within that time limit. Default is 0.
Job Status	The checkboxes available in this section refer to the current status of any job that you might include in reports. Available values are: **Completed, Failed, Aborted,** and **Cancelled**. **Note** Since Jobs do not fail (Tasks within Jobs can), searching for **Job Status - Failed** will yield no results. To generate a list of unsuccessful activities, search for **Task Status - Failed**.
Task Status	The checkboxes available in this section refer to the current status of any task that you might include in reports. Available values are: **Success, Failed, Aborted,** and **Cancelled**.

Column Selection Dialog

There are two text boxes on this dialog--**Available Columns** and **Selected Columns**. You can select and order the column headings to customize how the data displays on your report.

Note All available column headings are listed in the Selected Columns textbox. You must click the Clear All button to move them to the Available Columns textbox. Then, you can start to select the ones you want in the report.

- **Job (or Task) Name**
- **Source Host**
- **Target Destination**
- **Job (or Task) Status**

- **Start Time**
- **End Time**
- **Duration (minutes)**
- **# Success**
- **# Failed**
- **# Canceled**
- **# Aborted**

Header and Footer Style Selection Dialog

On this dialog, you can select a font, style, and point size for the body, header, and footer text of the report. Arial 8 is the default. Other formatting options are available, such as underline and strikeout. You can assign colors--even customized colors--for each table in the report.

Body Style	Format the body of the report using these fields: • **Font:** pick the typeface, style, and size from the drop down menu. • **Even Row Color:** choose a color for the even rows of the report, to increase readability. • **Odd Row Color:** choose a color for the even rows of the report, to inc ease readability.
Header	In this section, you can format the header using these fields: • **Logo**: Browse to select an image to display in the header. • **Text**: In this field, enter any text that you want displayed in the header. • **Font**: Use the browse ellipsis to select a font and point size for any text that displays in the header. • **Show Date and Time**: Select this checkbox to format the header to include the current date and time.
Footer	Format the footer of the report using these fields: • **Logo**: Browse to select an image to display in the footer. • **Text**: In this field, enter any text that you want displayed in the footer. • **Font**: Use the browse ellipsis to select a font and point size for any text that displays in the footer. • **Show Page Numbers**: Select this checkbox to paginate the footer.

Recurrence Schedule Selection Dialog

Through this dialog, you can select a schedule for generating the report or make the report available on demand. This scheduling interface is similar to the one used for backup and restore jobs.

This will be an On Demand job . . .	No schedule for the report is set; it will be run on demand. When the dialog displays, this field is selected by default and all other fields are disabled. If you deselect this checkbox, the other fields will be enabled. This will allow you to schedule a report based on a specific timeframe.
Start Time	The time of day at which to run the report.
Recurrence Pattern	In this section, you establish how often the report should be generated. There are five options within this field: • Daily • Weekly • Monthly • Yearly • Interval
Daily	The daily option can be scheduled to run the report every weekday or every *x* number of days.
Weekly	vRanger can be configured to run on weekly intervals, from every week to every 99 weeks. The day of the week on which to run reports can be configured.
Monthly	The monthly option offers the following configurations: • **Day** *[x]* **of every** *[y]* **month:** •*x* can be any value from 1 to 31. This determines the day of the month that the job will run. •*y* can be any value from 1 to 99. This value determines the monthly interval--for example, every 2 months sets the job to run every other month. • **The** *[f]* *[d]* **of every** *[y]* **month(s)**: •*f* can be either: first, second, third, fourth or last. •*d* can be: weekday, weekend day, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday. •*y* can be any value from 1 to 99. This value determines the monthly interval--for example, every 2 months sets the report to run every other month.

Yearly	The yearly option offers the following configurations: • **Every** *[m][x]*: • *m* is any month of the year. This determines the month of the year in which the report will run. • *x* can be any value from 1 to 31. This determines the day of the month that the report will run. • **The** *[f][d]* **of** *[m]*: • *f* can be either: first, second, third, fourth, or last. • *d* can be: day, weekday, weekend day, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday. • *m* is any month of the year. This determines the month of the year in which the report will run.
Interval	This option allows you to choose: • **Days**: from 0 to 49 • **Hour(s)**: from 0 to 23 • **Minutes(s)**: from 0 to 59
Range of Recurrence	Select the start and end date for running the report. • **No end date**: The report will continue to run until it is stopped. • **End after**: Select the number of occurrences. • **End by**: Select the date by which to end the recurrence.

Report Publication Dialog

If you have set up an SMTP server to enable notification and completed the necessary configurations for reports to be sent via email, you can use the Report Publication dialog to notify recipients when a scheduled report is run.

- **To**--Accesses the address book where email addresses are stored. Browse to it. Requires that you previously entered email addresses.
- **Subject**--Use this field to identify the report to be run and sent.
- **Body**--The text that you enter in this field will be included in the body of the email sent.
- **Send as**--This setting determines the format of the report sent--either PDF or Excel.

Report Summary Dialog

This dialog brings together the settings that you selected during this process. It is important to confirm the information in advance of launching the report. If you would like to change any of these options, you can click the **Back** button to return to the dialog where you can make the change. Then, you can advance to the **Report Summary** dialog and click **Finish**.

Create a Summary Report

The process for creating reports is the same, whether the report covers a Backup, Replication, or Restore job or task. The process below uses a Backup Job report as an example.

1. In the My Reports pane, right-click **Backup Job Summary Report**-->**Add Backup Job Summary Report.**
2. On the **Report Wizard** dialog, populate the **Report Name** and **Report Description** fields. Click **Next.**
3. On the **Filter Selection** dialog, configure the report using **Date Range, VM/Host/ Repository, Duration,** and **Job Status**. Click **Next.**
4. The **Column Selection** dialog, click **Clear All** to move all headings to the **Available Columns** textbox.
5. In the **Available Columns** textbox, click to highlight a value and click **Add Column.** Click to select the columns that you want to display on the report. To select multiple columns, use the **Ctrl** key. Click **Add Column.**
6. To reorder the column headings, click to highlight the value you want to move. Then, click the up and down arrows adjacent to the **Selected Columns** textbox. Click **Next.**
7. From the fields on the **Header and Footer Style Selection** dialog, **Body Style** section, click the ellipsis (. . .) to select the font and a color for odd and even rows.
8. In **Header Style**, click the ellipsis (. . .) to select logo (stored on your computer), enter the report header text you wish, choose a font and point size for the text, and deselect **Show Time and Date** if you wish.
9. In **Footer Style**, click the ellipsis (. . .) to select logo (stored on your computer), enter the report header text you wish, choose a font and point size for the text, and deselect **Show page numbers** if you wish.
10. Click **Next**.
11. On the **Recurrence Schedule Selection** dialog, select the **This will be an On Demand Report...** checkbox if this is the case. Select a time from the **Start** dropdown. Select the appropriate radio buttons in the **Recurrence Pattern** section. In the **Range of Recurrence** section, make a selection from the Start dropdown. Select a radio button. If you chose **End after**, select a value from the occurrences spinbox. If you chose **End by**, select a value from the date/time dropdown. Click **Next.**
12. The **Report Summary** dialog includes a glimpse of the settings you selected during this process. Confirm that it is accurate before you click **Finish.**
13. The report is added under the appropriate node in the **My Reports** pane.
14. On the **Report Publication** dialog, click the browse ellipsis adjacent to the **To** field.
15. In the **Selected** column on the **Select Email Recipients** dialog, click the checkbox corresponding to the recipient you want notified. Click **OK.**
16. On the **Report Publication** dialog, populate the **Subject** and **Body** fields. In the **Send as** field, select a radio button--**PDF** or **Excel**. Click **Next.**

Using the vRanger Console

This chapter provides an overview of the vRanger Console and a description of the available commands.

This chapter contains the following sections:This section contains the following topics:

vRanger Console Overview 178
vRanger Console - One-time Configuration 178
vRanger vAPI Cmdlets 179

vRanger Console Overview

vRanger uses Microsoft PowerShell with a Quest snap-in to provide scripting access to vRanger features and functions. This allows for vRanger to be tightly integrated with other data protection solutions and offers an efficient configuration option for advanced users.

The snap-in includes the vAPI cmdlets required to create and manage jobs as well as view relevant information data reading your virtual environment.

For more information on Microsoft Powershell, refer to Windows Powershell Owner's Manual:

http://technet.microsoft.com/en-us/library/ee177003.aspx

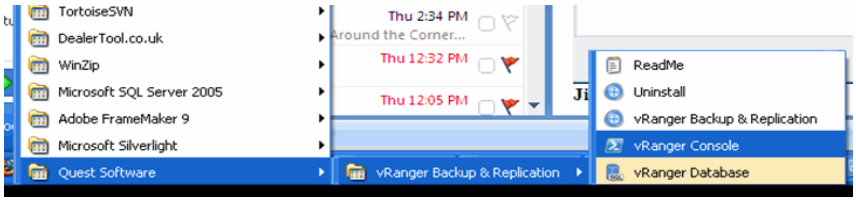
vRanger Console - One-time Configuration

The vRanger installer will automatically install Microsoft PowerShell 1.0 and the vAPI snap-in during the product installation. Before using the vAPI cmdlets, you must first configure the PowerShell execution policy to run the vRanger scripts.

To set the execution policy:

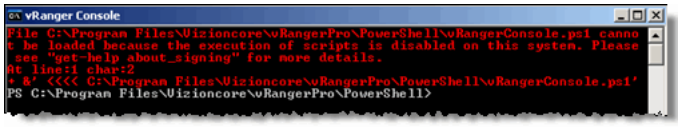
Step 1: vRanger Console

1. From the **Start** menu, navigate to the vRanger PowerShell directory. By default this is:**Start>All Programs>Quest Software>vRanger Backup & Replication>vRanger Console**
2. Select **vRanger Console**.



Step 2: Execution Policy

When opening the vRanger Console for the first time, you will see an error message shown below. This is due to the default PowerShell execution policy.



1. At the prompt, type the command below and press **Enter**.

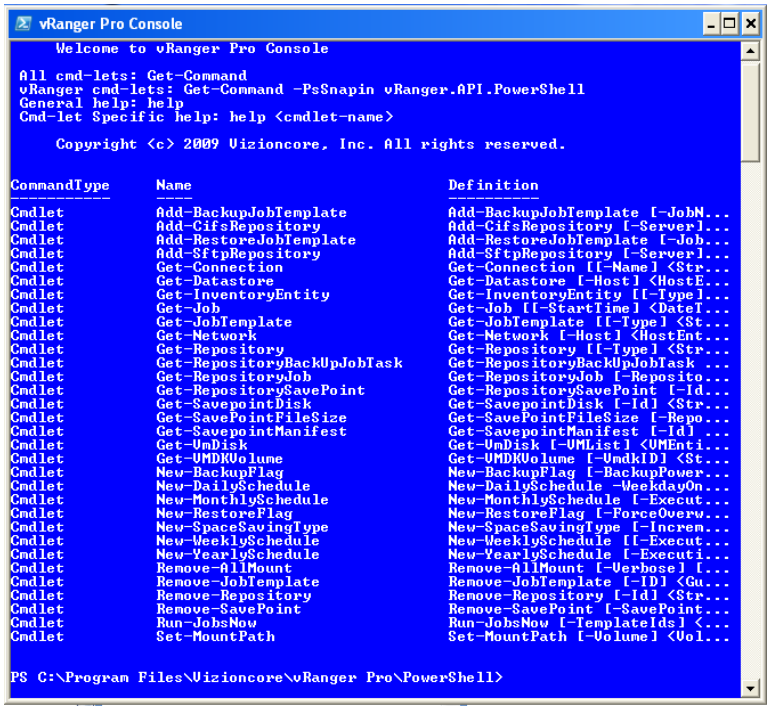
```
Set-ExecutionPolicy RemoteSigned
```

Note PowerShell and vAPI commands are *not* case sensitive.

2 When the prompt re-appears, close and re-open the vRanger Console.

Step 3: vAPI Snap-In

When the vRanger Console restarts, you should see the screen below, with a blue background and a list of vAPI cmdlets.



The vRanger vAPI is now configured and ready for use.

vRanger vAPI Cmdlets

The vRanger PowerShell snap-in includes the cmdlets listed below.

Inventory Management

Add-DdbRepository	Get-InventoryEntity
Add-VirtualAppforLinuxFLR	Get-IsInventoryRefreshing
Add-VirtualAppforLinuxFLRVA	Get-VirtualMachinesUnderInventory
Add-VirtualCenter	Update-Inventory
Get-Datastore	

Configuration

Dismount-LinuxVolume	New-TransportConfiguration
Get-Network	New-VirtualAppliance
Get-VirtualApplianceConfig	Remove-AllMount
Get-VmDisk	Remove-LinuxVolume
Get-VMDKVolume	Remove-VirtualAppliance
Install-VirtualAppliance	Set-CBTonVM
Mount_LinuxVolume	Set-LinuxVolume
Move-ToolsToLinuxFLRVA	Set-MountPath
Move-ToolsToVA	Set-Resources
New-EmailAddress	Update-VirtualAppliance
New-SMTPServer	

Job Management

Disable-Job	New-IntervalSchedule
Enable-Job	New-MonthlySchedule
Get-AddressBook	New-WeeklySchedule
Get-ConfigOption	New-YearlySchedule
Get-Connection	Remove-JobTemplate
Get-CurrentTemplateVersionID	Run-JobsNow

Get-Job	Set-Cataloging
Get-JobTemplate	Stop-vRangerJob
New-Daily Schedule	

Repositories

Add-CIFSRepository	Get-RepositoryJob
Add-DdbRepository	Get-RepositorySavePoint
Add-FTPRepository	Get-Savepoints
Add-NFSRepository	Remove-Repository
Add-NVSDRepository	Remove-SavePoint
Add-SFTPRepository	Remove-DdbStorageUnit
Get-Repository	

Backup

Add-BackupGroupEntity	New-BackupGroupMember
Add-BackupJobTemplate	Remove-BackupGroupEntity
Get-BackupGroupEntity	Remove-BackupGroupMember
New-BackupFlag	Update-BackupJobTemplate

Restore

Add-RestoreFromManifestJobTemplate	Get-SavepointManifest
Add-RestoreJobTemplate	New-RestoreFlag
Get-CatalogSearchData	Remove-Catalog
Get-CatalogStatus	Update-RestoreJobTemplate
Get-SavepointDisk	

Replication

Add-ReplicationJobTemplate	Run-ReplicationFailover
Get-GlobalTransportFailover	Run-ResumeReplicationFailover
Get-MonitorLog	Run-TestReplicationFailover
Get-TransportFailover	Update-GlobalTransportFailover
New-ReplicationFlag	Update-ReplicationJobTemplate

Cmdlet Help

To get help on any of the vAPI cmdlets, use the `Get-Help` command. At the prompt, enter `Get-Help` *`command`*, where *`command`* is any of the above vAPI cmdlets.

For example, enter `Get-Help Add-SFTP Repository` and the following is displayed:

```
Description The Add-SftpRepository cmdlet adds a SFTP repository to vRanger. The
repository can be later used to store backup files from backup jobs.
Syntax Add-SftpRepository [-Server] <string> [[-Port] [<int>]] [[-
TargetDirectory][<string>]] [-User] <string> [-Password] <string> [[-TimeOut]
[<short>]] [-Name] <string> [[-Description] [<string>]] [[-Overwrite]
[<Boolean>]][<CommonParameters>]
```

Note PowerShell and vAPI commands are not case sensitive. vAPI cmdlets are shown capitalized only to make them easier to read.

For more information, you may use the "-detailed" or "-full" parameters. For example: `Get-Help Add-SFTPRepository -Detailed` or `Get-Help Add-SFTP Repository - Full.`

Integrating vRanger

This chapter describes the available options for integrating vRanger with third party solutions.

This chapter contains the following sections:

Integrating vRanger 184
Monitoring vRanger 191

Integrating vRanger

Like any component of your virtual infrastructure, vRanger can be used in conjunction with other applications to achieve more complex tasks.

vRanger and Tape Backup Vendors

To configure the vRanger sweep-to-tape feature, both vRanger and the tape vendor's software must point to a common vRanger repository folder located locally on the vRanger server.

The vRanger backup administrator creates an on-demand (not scheduled), incremental job with a specific name, appending `-Tape` as a case-sensitive suffix (e.g., any-jobname-Tape). The -Tape suffix can be changed by editing the PowerShell script if you prefer something else. There should be a dedicated repository for each on-demand job.

The tape software administrator schedules a recurring incremental backup job whose PRE field calls a .cmd file containing the PowerShell script that searches for and then starts the vRanger on-demand job(s) with the -Tape suffix.

The vRanger on-demand job performs the incremental backup, placing the savepoint files in the repository and updating the repository's global manifest file.

After the vRanger job completes (e.g., the job status equals either success or failure), the PowerShell script exits, thus completing the Backup Exec PRE command.

With the tape software's PRE command completed, the tape software, which is also pointing to the vRanger repository folder, runs its scheduled incremental job, backing up the vRanger savepoints and global manifest file to tape or whatever media server type is configured.

Required Components

The components required differ depending on the installation scenario.

Scenario 1 - vRanger & 3rd-party tape software on the same server

- Tape vendor's software on a Windows 2003 server.

  Note Note: a physical server will provide better performance than a VM.
- vRanger resides on the tape software server.
- A local repository shared by vRanger and the tape software.
- PowerShell (PowerShell and vRanger must always be installed on the same server).
- A PowerShell script that finds and launches the vRanger on-demand job.
- A CMD file that contains the PowerShell script.

Scenario 2 - vRanger & 3rd-party tape software on different servers

The tape software is on its own server. All other components, including a remote tape backup agent, reside on the vRanger server. Note: Scenario 2 is not discussed in this document.

Server 1-Backup Exec:

- Tape vendor's software on a Windows 2003 server

Server 2-vRanger/PowerShell:

- Tape software's remote tape backup agent that can run a PRE command. The agent runs the PowerShell script contained in the .CMD file called via the PRE command
- vRanger resides on its own server
- A local repository located on the vRanger server and shared by vRanger and the tape software
- PowerShell (PowerShell and vRanger must always be installed on the same server).
- A PowerShell script that finds and launches the vRanger on-demand job
- A CMD file that contains the PowerShell script.

Configuration Steps

Use the procedures below to configure integration of your tape software and vRanger.

Note Although the procedure below uses Symantec Backup Exec, the procedures should be similar for any tape vendor's software.

These procedures assume that:

- You have your components installed according to Scenario 1 as described above.
- The customer already will have a Backup Exec physical server and vRanger will be installed on that server.

Installing vRanger

1. Install vRanger onto your Backup Exec server. Consider placing the local repository on the D: drive of the vRanger server.
2. Create the local share *D:\vranger_backups* (or similar).
   a. Assign the group Everyone (or similar) Full Control for the Share.
   b. The user credentials specified in the next step when adding the repository should have Full Control NTFS security. This folder will become the repository that contains the incremental job savepoints and global manifest file. The folder should have enough disk space to hold a full backup and several incremental backups.
3. Add a local repository on the vRanger server that points to the vranger_backups share. Click **My Repositories > Add >** specify the necessary repository details, pointing to the share.
4. Create an on-demand, daily, incremental backup job that backs up your environment. In this paper, we will backup a single server for simplicity. Click:
   a. **My Inventory > Add > Backup Job > Advanced > Job Name** = vRanger-Sweep-to-Tape. Click **Next**.
   b. Select VMs to exclude. Click **Next**
   c. Select hard disks to include. Click **Next**.
   d. Select your options. Click **Next**.

e For Retention Policy, select **Savepoint Count=7, Space Saving Technology=Incremental, Threshold Count=6.** Click **Next**.

If Backup Exec calls this on-demand job daily, there will be a full backup performed the first day and an incremental backup performed each of the remaining days of the week.

f Select the **This will be an on demand job and does not require a recurrence schedule** checkbox. Click **Next**.

g Click the vRanger-Sweep-to-Tape repository to select it. Click **Next**.

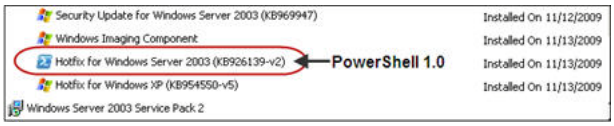
h Select the email address(es) that should be notified when the job has completed running. The **Email a report after the job has finished running** check box should remain checked. Click **Next**.

i Review your settings and click **Finish**.

Installing and Configuring PowerShell

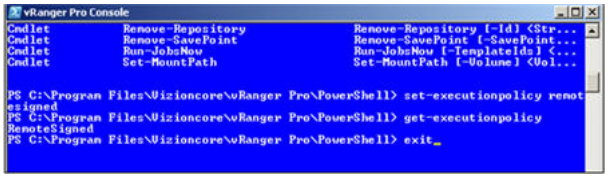
1 From *http://www.microsoft.com/download/en/details.aspx?id=20020*, download PowerShell 1.0 for Windows Server 2003 to the vRanger server.

2 Install PowerShell. PowerShell appears in Add/Remove Programs as a hotfix:



3 Click **Start > All Programs > Quest Software> vRanger > vRanger Console** to launch the vRanger Console.

4 Set the PowerShell Script Execution Policy to **RemoteSigned**.



a Set the PowerShell Script Execution Policy to **RemoteSigned** by running the cmdlet:

`Set-ExecutionPolicy RemoteSigned`.

b Confirm the policy setting by running the **Get-ExecutionPolicy** cmdlet.

c Exit PowerShell by running the **Exit** cmdlet.

See *http://www.microsoft.com/technet/scriptcenter/topics/msh/cmdlets/set-executionpolicy.mspx* for an explanation of the available PowerShell Execution Policies.

5 Edit the vRanger config files with Notepad (or similar) to use the vRanger server's local IP address instead of localhost.

a Edit the vRanger.API.exe.config file, located at: `\Program Files\Quest Software\vRanger\PowerShell \vRanger.API.exe.config`.

- Change:`<endpoint address="http://localhost:2480/VAPIHost.svc" binding="wsDualHttpBinding"`

To: `<endpoint address="http://<Server-IP>:2480/VAPIHost.svc" binding="wsDualHttpBinding"`

b Edit the vRanger.API.PowerShell.dll.config file, located at `\Program Files\Quest Software\vRanger\PowerShell \vRanger.API.PowerShell.dll.config`.

- Change:`<endpoint address="http://localhost:2480/VAPIHost.svc" binding="wsDualHttpBinding"`

  To:`<endpoint address="http://<Server-IP>:2480/VAPIHost.svc" binding="wsDualHttpBinding"`

Using the PowerShell Script

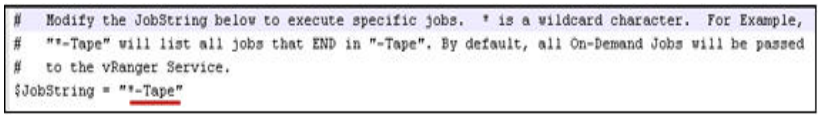
1 Download the PowerShell script that will launch the vRanger on-demand job.

a Go to the URL and save the zip file to any folder (e.g., c:\download) on the vRanger server: *http://communities.quest.com/docs/DOC-10738*.

b Extract the script from the zip file by right-clicking the file and clicking **Extract All**.

c While you do not need to edit the script, you can open it with Notepad to understand why your vRanger job name has to have the suffix -Tape (case sensitive).

The PowerShell $JobString = "*-Tape" must match the suffix used in your vRanger job name. As long as they match, you can change the $JobString and vRanger job name suffix to something other than -Tape.



d Close the file.

e Using Notepad, create a CMD file in the `C:\Download\vRangerTape.ps1` folder titled *vRangerTape.cmd*. Its contents are:

```
@echo off
c:\Windows\system\system32\windowspowershell\v1.0\powershell.exe
c:\download\vRangerTape.ps1\vRangerTape.ps1
```

Note Make sure that the vRangerTape.cmd file has .CMD extension, not a .TXT extension.

Configuring Backup Exec

1 Install and configure Backup Exec according to the applications instructions.

2 When configuring the Backup-to-disk folder type, select **Backup-to-disk** folder. Click **Next**.

3 At the Enter a name for the backup-to-disk-folder prompt, type vRanger Daily and click **Next**.

4 At the Enter a path for the backup-to-disk-folder prompt, type or browse to D:\Backup_Exec Backups and click **Next**.

5 At the Enter the maximum size for a backup-to-disk file prompt, type a number of GBs (default=4).

Note You can later change this number from within Backup Exec.

6 At the Allocate the Maximum Size for Backup-to-Disk Files prompt, click the **Yes** radio button and click **Next**. Pre-allocating the space improves performance.

7 At the Enter the maximum number of concurrent jobs prompt, type 1 (the default) and click **Next**.

8 At the Enter the size for the low disk space threshold prompt, type 0 (the default) and click **Next**.

9 Review your Device Configuration Wizard settings and click **Next**.

10 Click **Finish**.

11 Create a media set named vRanger Daily Backups. Set the **Overwrite Protection Period** to 4 weeks.

12 Set the **Append Period** to 4 weeks.

Create a Backup Job

1 Click on **Create a backup job**. This launches a wizard.

   a At the Welcome to the Backup Wizard page, click **Create a backup job with custom settings** and click **Next**.

   b At the Backup Selections page, drill down to and select the vRanger repository e.g., D:\vRanger_Backups, and click **Next**.

   c At the Select Volume Credentials page, click **Test All**. When the results are successful, click **Next**.

   d At the Select Volume Order page, click **Next**.

   e At the Backup Names page, title the backup job vRanger Daily Job, title the backup set vRanger Daily Backup, and click **Next**.

   f At the Backup Device and Media page, at the W**hich device would you use to backup your data** drop-down box, select **vRanger Daily**. At the **Which media set would like to use to back your data** drop-down box, select **vRanger Daily Backup**. When finished making your selections, click **Next**.

   g At the Backup Overwrite Method page, accept the default **Append to media, overwrite if no appendable media is available** radio button. Click **Next**.

   h At the Backup Options page, click the **Incremental - Changed Files - Reset Archive Bit** method from the drop-down list. Click **Next**.

   i At the Completing the Backup Wizard page, click the **No, schedule the job to run later** radio button and click **Finish.**

   j At the Schedule Options page, click the **Run according to schedule** radio button and click the **Edit Schedule Details** button.

   k At the Edit Calendar schedule by box, click the **Recurring Days of the Month** option and then click the **Select All** button.

   l At the Edit Calendar schedule by box, click the **Effective Date** option and confirm that today's date is placed in the Make the schedule go into effect on drop-down box.

m At the Edit Calendar schedule by box, click the **Time Window** option and accept the defaults: Start no earlier than 11:00:00 PM and no later than 10:59:59 PM the following day.

n Click **Submit**.

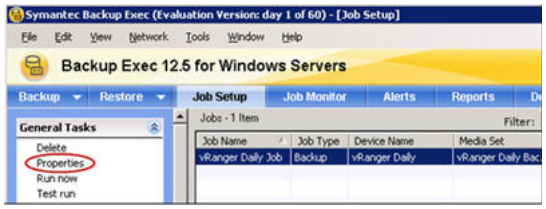
o At the Completing the Backup Wizard page, click the **No, schedule the job to run** later radio button and click **OK.**

p Click **Close** to close the Getting Started with Backup Exec page.

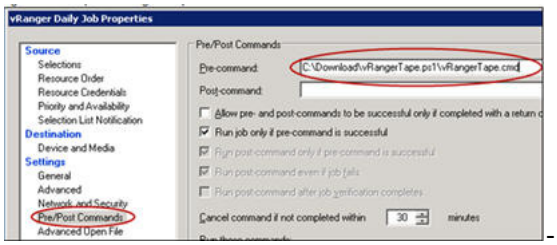
2 Specify the *vRangerTape.cmd* file in the Backup Exec job's PRE field.

a Close the Backup Exec Assistant window.

b Click on the vRanger Daily Job and click **Properties**.



c In the vRanger Daily Job Properties column, click Pre/Post Commands and type the PowerShell script path\command (e.g., `C:\Download\vRangerTape.ps1\vRangerTape.ps1`).

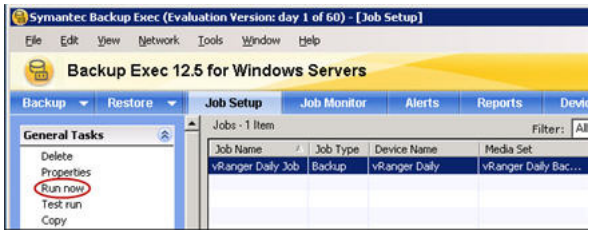


d Click **Submit**.

e At the Job Summary page, click **OK.**

Run Sweep-to-Tape

1 In Backup Exec, highlight the vRanger Daily Job and click **Run Now**. Click **Yes** to confirm.



2 Click the Job Monitor tab and double-click the Active job.

Restore From Backup Exec

Restoring from Backup Exec is the reverse procedure of doing backups.

1. From within Backup Exec, restore from tape or disk to the vRanger repository.
2. From within vRanger, restore from savepoints, using the global manifest file or savepoint manifest files.

Troubleshooting Sweep-to-Tape

Sweep-to-tape has three main components. Each component should be tested individually.

- The vRanger on-demand, incremental backup job.
- The PowerShell script contained in the .CMD file.
- The Backup Exec scheduled, incremental backup job.

1. From with vRanger, manually run the vRanger on-demand job and confirm that it works.
2. Run the PowerShell script from a CMD prompt. Manually cut and paste the executable statement into the CMD prompt and check for errors to troubleshoot further:

```
c:\windows\system32\windowspowershell\v1.0\powershell.exe
c:\download\vRangerTape.ps1\vRangerTape.ps1
```

3. Launch PowerShell
   a. Run the **Get-PSSnapin** command. `vRanger.API.PowerShell` should be registered.
   b. Run the **Get-Executionpolicy** command. The execution policy should be set to **RemoteSigned.**
4. From Backup Exec, confirm that its job runs. If the job fails quickly, confirm that the Backup Exec services are using credentials with local administrator rights. If so, then the PRE command is probably failing due to a PowerShell issue.

Deduplication with NetVault SmartDisk and vRanger

NetVault SmartDisk delivers data de-duplication that significantly reduces storage costs. With powerful, byte-level, variable block de-duplication, and advanced data compression that packs more protected data in the same storage area than competing solutions, you can reduce your disk backup storage footprint by up to 90%.

vRanger versions 5.3 and later support the use of a NetVault SmartDisk server as a repository. Simply by adding a NetVault SmartDisk server to vRanger as a repository, backups written to that repository will be automatically de-duplicated.

Note NetVault SmartDisk repositories are only supported by the Professional edition of vRanger. If you purchased a vRanger Standard license, you will not be able to use a NetVault SmartDisk repository.

For more information about integrating vRanger and NetVault SmartDisk, see the *vRanger Netvault SmartDisk Integration Guide*.

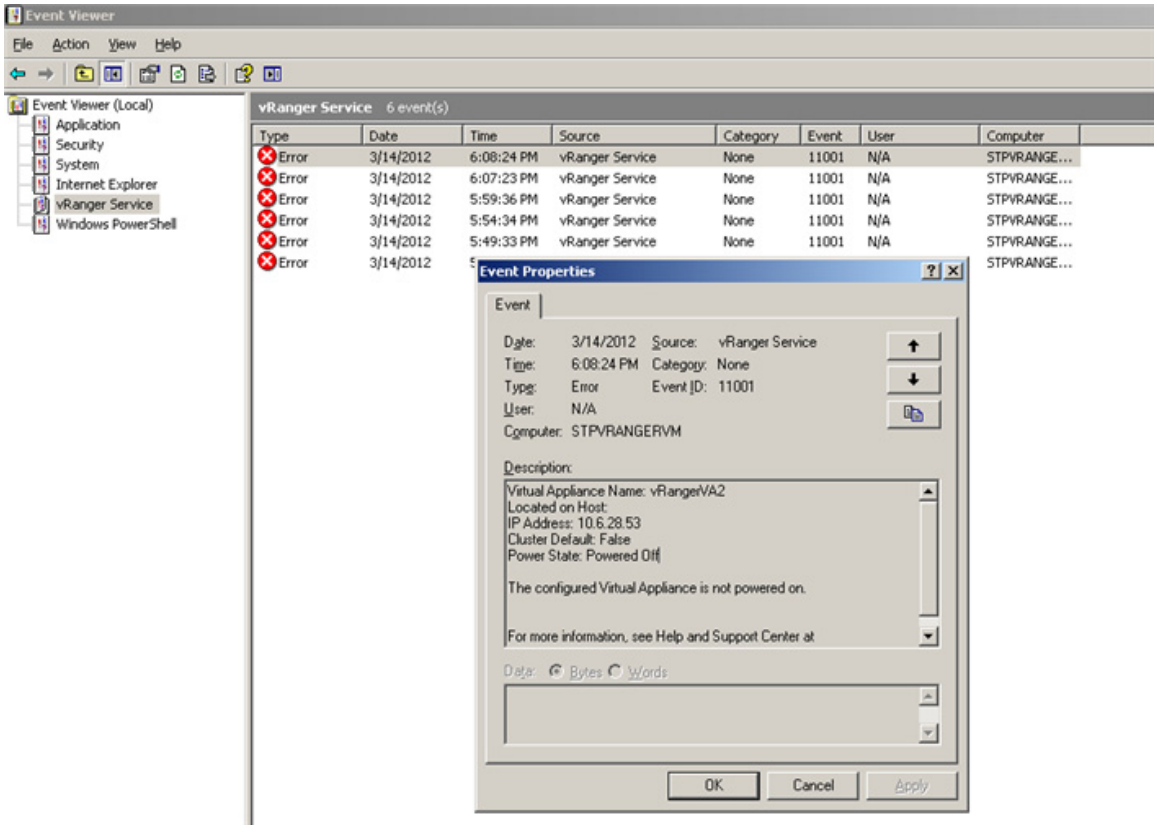
Data Domain and vRanger Repositories

Quest vRanger and Data Domain appliances provide companies with a simple and efficient method for backing up and recovering VMware environments. vRanger provides significant enhancements with regard to architecture, performance and communications over traditional backup solutions. Data Domain provides superior data de-duplication capabilities. The combination of vRanger and Data Domain Technology will dramatically reduce your backup and recovery time for virtual machines on VMware ESX hosts.

For more information about integrating vRanger and Data Domain, see the document *vRanger Pro & Data Domain Best Practices*.

Monitoring vRanger

vRanger can be configured to log events to the Windows Event Viewer on the vRanger machine. These events can be filtered and monitored using Microsoft's System Center Operations Manager in order to monitor replication functions in line with other IT operations. When monitoring is enabled, vRanger will create a "vRanger Service" node on the local Event Viewer, as shown below:



The events currently monitored and logged by vRanger are:

- **Replication Events - Reserved ID Range 10000 - 10999**
  - Replication task failure - logged as an error, event ID 10001
- **Virtual Appliance Events - Reserved ID Range 11000 - 11999**
  - VA not powered on - logged as an error, event ID 11999
  - VA connection failure - logged as an error, event ID 11999
  - VA credential failure - logged as an error, event ID 11999
  - VA unknown failure - logged as an error, event ID 11999
  - VA return to normal - logged as an Information event, ID 11998
- **Datastore Events - Reserved ID Range 12000 - 12999**
  - Datastore threshold exceeded - logged as an error, event ID 12000
  - Datastore unknown failure - logged as an error, event ID 12099
  - Datastore return to normal - logged as an Information event, ID 12098

Configuring Event Monitoring

The configurations for event monitoring are done via the configuration file for the vRanger service -*Vizioncore.vRanger.Service.exe.config*. This file is located in the Service sub-directory of the vRanger installation directory. By default this is :

C:\Program Files\Quest Software\vRanger\Service

Caution Change only the values listed below unless directed otherwise by Quest Support.

To configure event monitoring:

Step 1: Opening the Configuration File

1. Open the *Vizioncore.vRanger.Service.exe.config* with your text editor of choice.
2. Carefully make the required changes, using the information below as guidance.

   Note Not all text editors include line numbers. If you do not see the line numbers referenced below in your text editor, this is not an indication of a problem.

Step 2: Configuring Replication Monitoring

Replication monitoring will log replication task failures to the Windows event log with the Event ID of 10001. The log will also include the error message and information about the task such as source virtual machine, target host, and virtual center.

To enable replication monitoring:

1. In the *Vizioncore.vRanger.Service.exe.config* file, scroll to line 162 or search for "`Replication FailureWriteEventLog`".
2. Change the value "false" to "true", ensuring the quotation marks remain.

3 Save the file and exit.

or

Continue to the next section to make additional changes.

Step 3: Configuring Virtual Appliance Monitoring

When enabled, at the specified interval the Virtual Appliance monitor will go through all of the configured Virtual Appliances and check to make sure that its power is turned on and that vRanger is able to connect using the entered IP address and credentials.

To enable VA monitoring:

1 In the *Vizioncore.vRanger.Service.exe.config file,* find the lines corresponding to the desired setting, as shown below.

- **VirtualApplianceMonitoringEnabled** - line 164.

  Set this value to "true" to enable monitoring of the virtual appliance.

- **VirtualApplianceMonitoringEventLogFailureLogMax** - line 167.

  This is the maximum number of times that a particular VA failure will be logged to the Event log before waiting for a success to proceed again. Change the value of "7"to the desired value, ensuring the quotation marks remain.

  This setting is intended to limit the amount of "spam" in your event viewer. A refresher message will appear periodically based on the value below.

- **VirtualApplianceMonitorSecondaryIntervalInHours** -

  This value determines how often the refresher message occurs for a failed virtual appliance check after the FailureLogMax value has been reached. The default value is "6", which will log an event every 6 hours.

- **VirtualAppliancesConnectionCheckIntervalInMinutes** - line 169

  This setting determines how often vRanger will check the connection to configured virtual appliances. The default value is 5 minutes.

2 Save the file and exit.

or

Continue to the next section to make additional changes.

Step 4: Configuring Datastore Monitoring

When enabled, at the specified interval, the datastore monitor will go through all of the configured datastores and determine if the space available on the datastore exceeds the specified threshold. This may be either by percentage, or a specific value in gigabytes. When this occurs an event is logged in the windows event log with an ID of 12000.

To configure datastore monitoring:

1 In the *Vizioncore.vRanger.Service.exe.config file,* find the lines corresponding to the desired setting, as shown below.

- **DatastoreMonitoringEnabled** - line 171

Set this value to “true” to enable monitoring of the datastores.

- **DatastoreMonitoringSpaceThresholdValue** - line 173

  This is the threshold value for a datastore space before a warning is sent. This value is modified my the **DatastoreMonitoringSpaceThresholdType** setting below.

- **DatastoreMonitoringSpaceThresholdType -** line 175

  This value determines whether vRanger checks for percentage of free space or an absolute value un GB. Set ”0” for percent, “1” for GB.

- **DatastoreMonitorCheckIntervalInMinutes** - line 177

  How often vRanger checks the datastores for low space (in minutes). Default value: 30 minutes.

- **DatastoreMonitoringEventLogFailureLogMax** - line 180

  The maximum number of times a particular datastore monitor failure will be logged to the windows event log before waiting for a success to proceed again. The default value is “7”.

  This setting is intended to limit the amount of “spam” in your event viewer. A refresher message will appear periodically based on the value below.

- **DatastoreMonitorSecondaryIntervalInHours** -

  This value determines how often the refresher message occurs for a datastore that has reached its space threshold value after the FailureLogMax value has been reached. The default value is “6”, which will log an event every 6 hours.

2 Save the file and exit.

Step 5: Restart the vRanger Service

In order for the changes to take effect, you must restart the vRanger service.

1 On the vRanger machine, click Start, then Run.Type “services.msc” and click **OK**.

2 Find the vRanger Service. Right-click on the service, and click **restart**.

Appendix A: vAPI Cmdlet Details

Add-BackupGroupEntity

Description	The Add-BackupGroupEntity cmdlet adds a new Backup Group to vRanger. New-BackupGroupMember cmdlet is used to add Members to a Backup Group created using this cmdlet or UI.
Syntax	`Add-BackupGroupEntity [-BackupGroupName] <String> [<CommonParameters>]`
Parameters	• **BackupGroupName <String>** The BackupGroup Name • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type "get-help about_commonparameters".
Examples	`Add-BackupGroupEntity -BackupGroupName BG1`
Related Links	Remove-BackupGroupEntity New-BackupGroupMember Remove-BackupGroupMember Get-BackupGroupEntity

Add-BackupJobTemplate

Description	Adds a backup job. The user also has the option of running the job immediately after the template is created.
Syntax	Add-BackupJobTemplate [-JobName] <string> [[-JobDescription] [<string>]] [-JobEntity] <Entity> [[-ExcludedVMList] [<string[]>]] [[-IncludedDiskList] [<int[]>]] [-TargetRepository] <Repository> [-JobSchedule] <Schedule> [-Flags] <BackupFlags> [-NumberOfSavePoints] <int> [-SpaceSavingTechnologyTypeFlag]<SpaceSavingTechnologyType> [[-NotificationList] [<string >]] [[-RunJobNow] [<Boolean>]] [<CommonParameters>]

Parameters	• **JobName <string>** Job name. • **JobDescription [<string>]** Job description. • **JobEntity <Entity>** Inventory entity on which the job will be run: ESX host, vCenter etc. • **ExcludedVMList [<string[]>]** List of VMs to exclude from this job. All VMs in the inventory entity will be backed up if none are specified. • **IncludedDiskList [<int[]>]** List of disks to include in this job. All VM disks will be backed up if none are specified. • **TargetRepository <Repository>** Repository where the backups will be saved. • **JobSchedule <Schedule>** The recurrence schedule the job will be run on. • **Flags <BackupFlags>** Backup flags: NoFlags, BackupPoweredOnOnly, CheckDestinationFreeSpace, UseCompression, UpdateNotes, EnableGuestQuiescing, AllFlags. Multiple flags can be separated by '|' • **NumberOfSavePoints <Int32>** Max number of savepoints to keep before old ones are removed • **SpaceSavingTechnologyTypeFlag <SpaceSavingTechnologyType>** Spacesaving technology to use: differential, incremental or none. • **SpaceSavingCountThreshold [<Int32>]** The number to save before performing a full backup. • **SpaceSavingPersentSizeThreshold [<Int32>]** The size to allow as a % of the virtual machine hard disk file before performing a differential backup • **NotificationList [<string>]** Email notification list for the job. Separate multiple emails by ','. • **RunJobNow [<Boolean>]** Whether to run the job immediately after the job template is created. • **TransportConfiguration [<TransportConfiguration>]** The transport configuration options to use for this job. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Add-BackupJobTemplate -JobName myjob -JobEntity $inv -TargetRepository $repo -JobSchedule $s -Flags NoFlags -NumberOfSavePoints 3 -SpaceSavingTechnologyTypeFlag None` This command adds a backup job template. Since the RunJobNow flag is not specified the job will not be run immediately.

Related Links	Add-RestoreJobTemplate Add-ReplicationJobTemplate Update-BackupJobTemplate Get-Job Get-JobTemplate

Add-CIFSRepository

Description	The Add-CIFSRepository cmdlet adds a CIFS repository to vRanger. The repository can be later used to store backup files from backup jobs.
Syntax	`Add-CifsRepository [-Server] <String> [-ShareName] <String> [[-TargetDirectory] <String>] [-DomainName] <String> [-User] <String> [-Password] <String> [-Name] <String> [[-Description] <String>] [[-Overwrite] <Boolean>] [[-EnableEncryption] <Boolean>] [[-EncryptionPassword] <String>] [[-ImportOption] <String>] [<CommonParameters>]`
Parameters	• **Server <String>** The CIFS Server FQDN or IP. • **ShareName <String>** Name of the file system share the repository is on • **TargetDirectory [<String>]** The CIFS Target Directory. • **DomainName <String>** Name of the domain • **User <String>** The CIFS User Name. • **Password <String>** The CIFS Password. • **Name <String>** A friendly name describing this CIFS server. • **Description [<String>]** The Description describing the CIFS server. • **Overwrite [<Boolean>]** Whether to overwrite the existing repository if one exists • **EnableEncryption [<Boolean>]** Whether to enable encryption • **EncryptionPassword [<String>]** The encryption Password. • **ImportOption [<String>]** Whether to import the save points as read-write or readonly. • **<CommonParameters>** This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer and OutVariable. For more information, type, "get-help about_commonparameters".

Examples	`Add-CifsRepository -Server myserver.dev.corp -StorageUnit mystorage -TargetDirectory myrepo -User myname -Password mypassword -Name myCifs Repo -Overwrite 1` This command adds a repository on server myserver.dev.corp pointing to storage unit named mystorage using the credentials provided. Existing repositories at the location specified will be overwritten.
Related Links	Get-Repository Add-DdbRepository Add-NFSRepository Add-NVSDRepository Add-SFTPRepository Add-FTPRepository

Add-DdbRepository

Description	Adds a DDB repository to vRanger. The repository can be later used to store backup files from backup jobs.
Syntax	Add-DdbRepository [-Server] <String> [-StorageUnit] <String> [-User] <String> [-Password] <String> [-Name] <String> [[-Description] <String>] [[-Overwrite] <Boolean>] [[-ImportOption] <String>] [<CommonParameters>]
Parameters	• **Server <String>** The DDB Server FQDN or IP. • **StorageUnit <String>** The DDB Storage Unit • **User <String>** The User Name to access the DDB Server. • **Password <String>** The Password to access the DDB Server. • **Name <String>** A friendly name describing this DDB server. • **Description [<String>]** The Description describing the DDB server. • **Overwrite [<Boolean>]** Whether to overwrite the existing repository if one exists • **ImportOption [<String>]** Whether to import the save points as read-write or readonly. • **<CommonParameters>** This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer and OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Add-DdbRepository -Server myserver.dev.corp -StorageUnit mystorage -User myname -Password mypassword -Name myDDB Repo -Overwrite 1` This command adds a repository on server myserver.dev.corp pointing to storage unit named mystorage using the credentials provided. Existing repositories at the location specified will be overwritten.
Related Links	Get-Repository Add-CIFSRepository Add-NFSRepository Add-NVSDRepository Add-SFTPRepository Add-FTPRepository

Add-EsxHost

Description	Adds an ESX host to the inventory
Syntax	`Add-EsxHost [-DNSorIP] <String> [-Username] <String> [-RootPassword] <String> [[-UserPassword] <String>] [[-Port] <Int32>] [[-AddUser] <Boolean>] [<CommonParameters>]`
Parameters	• **DNSorIP <string>** The DNS or IP for the Esx/Esxi Host. • **Username <string>** The Username for the Esx/Esxi Host. • **RootPassword <string>** The Root password for the Esx/Esxi Host. • **UserPassword [<string>]** The User password for the Esx/Esxi Host. • **Port [<Int32>]** The SSH Port for the Esx/Esxi Host. • **AddUser [<Boolean>]** Add user of the user does not exist. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Add-EsxHost -DNSorIP 10.6.20.20 -Username root -RootPassword p@ssw0rd` This command adds an EsxHost to the inventory.
Related Links	Add-VirtualCenter Get-InventoryEntity

Add-FTPRepository

Description	The Add-FtpRepository cmdlet adds a FTP repository to vRanger. The repository can be later used to store backup files from backup jobs.
Syntax	Add-FtpRepository [-Server] <string> [[-Port] [<int>]] [[-TargetDirectory][<string>]] [-User] <string> [-Password] <string> [[-TimeOut] [<short>]] [-Name] <string> [[-Description] [<string>]] [[-Overwrite] [<Boolean>]] [[-Enable Encrytpion] [<Boolean>]] [[-EncryptionPassword] [<Boolean>]] [[-EnableDeduplication] [<Boolean>]] [[-BlockSize] [<int>]] [[-EnableCompress] [<Boolean>]] [[-ImportOption] [<string>]] [<CommonParameters>]
Parameters	• **Server <string>** The Ftp Server FQDN or IP. • **Port [<int>**] The FTP Server Port. • **TargetDirectory [<string>]** The FTP Target Directory. • **User <string>** The FTP user name. • Password <string> The FTP Password. • **TimeOut [<short>]** Timeout value. • **Name <string>** A friendly name describing this Ftp server. • **Description [<string>**] The Description describing the Ftp server. • **Overwrite [<Boolean>]** Whether to overwrite the existing repository if one exists. • **EnableEncryption [<Boolean>]** Whether to enable Encryption. • **EncryptionPassword <string>** The Encryption Password. • **EnableDeduplication [<Boolean>]** Whether to enable deduplication on backups on this repository. • **BlockSize [<Boolean>]** Set Block Size (KB) • **EnableCompress [<Boolean>]** Whether to compress all backups to this repository. • **ImportOption [<string>]** Whether to import the save points as read-write or readonly. Possible options 'ReadOnly' and 'ReadWrite'. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	None
Related Links	Get-Repository Add-CIFSRepository Add-NFSRepository Add-NVSDRepository Add-SFTPRepository

Add-NFSRepository

Description	The Add-NfsRepository cmdlet adds a Network File Share repository to vRanger. The repository can be later used to store backup files from backup jobs.
Syntax	Add-NfsRepository [-Server] <String> [-ExportDirectory] <String> [-TargetDirectory] <String> [-Name] <String> [[-Description] <String>] [[-Overwrite][<Boolean>]] [[-EnableEncryption] [<Boolean>]] [[-EncryptionPassword] <String>] [[-BlockSize] <Int32>] [[-EnableCompress] [<Boolean>]] [[-ImportOption] [<string>]] [<CommonParameters>]
Parameters	• **Server <String>** The NFS Server FQDN or IP. • **ExportDirectory <String>** The NFS Export Directory. • **TargetDirectory <String>** The NFS Target Directory. • **Name <String>** A friendly name describing this NFS server. • **Description <String>** The Description describing the NFS server. • **Overwrite [<Boolean>]** Whether to overwrite the existing repository if one exists • **EnableEncryption [<Boolean>]** Whether to enable encryption on backups on this repository • **EncryptionPassword <String**> The encryption Password. • **ImportOption [<string>]** Whether to import the save points as read-write or readonly. Possible options 'ReadOnly' and 'ReadWrite'. • **BlockSize <Int32>** Set Block Size (KB) • **EnableCompress [<Boolean>]** Whether to compress all backups to this repository • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`Add-NfsRepository -Server myserver.dev.corp -ExportDirectory /vol/vol 1 -TargetDirectory /test -Name MyNFSRepo -Overwrite 1`
	
	This command adds a repository on server myserver.dev.corp. Existing repositories at the location specified will be overwritten.
	
	---------- EXAMPLE 2 ----------
	`Add-NfsRepository -Server myserver.dev.corp -ExportDirectory /vol/vol 1 -TargetDirectory /test -Name MyNFSRepo -EnableEncryption 1 -EncryptionPassword myPassword -EnableCompress 1`
	
	This command adds a repository on server myserver.dev.corp. Enables encryption on backups of this repository. All backups to this repository will be compressed.

Related Links	Get-Repository Add-CIFSRepository Add-NVSDRepository Add-SFTPRepository Add-FTPRepository

Add-NVSDRepository

Description	The Add-NvsdRepository cmdlet adds a NVSD (Netvault Smart Disk) repository to vRanger. The repository can be later used to store backup files from backup jobs.
Syntax	`Add-NvsdRepository [-Server] <String> [-TargetDirectory] <String> [-User] <String> [-Password] <String> [-Name] <String> [[-Description] <String>] [[-Overwrite] <Boolean>] [[-ImportOption] <String>] [<CommonParameters>]`
Parameters	• **Server <String>** The NVSD Server FQDN or IP. • **TargetDirectory [<string>]** The NVSD Target Directory. • **User <string>** The NVSD user name. • **Password <string>** The NVSD Password. • **Name <String>** A friendly name describing this NVSD server. • **Description <String>** The Description describing the NVSD server. • **Overwrite [<Boolean>]** Whether to overwrite the existing repository if one exists • **ImportOption [<string>]** Whether to import the save points as read-write or readonly. Possible options 'ReadOnly' and 'ReadWrite'. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Add-NvsdRepository -Serve r myserver -User myname -Password mypassword -Name reponame -tagetDirectory /targetDirectoryName -Overwrite 1` This command adds a repository on server myserver pointing to a target directory named “targetDirectoryName” using the credentials provided. Existing repositories at the location specified will be overwritten.
Related Links	Get-Repository Add-DdbRepository Add-NFSRepository Add-CIFSRepository Add-SFTPRepository Add-FTPRepository

Add-ReplicationJobTemplate

Description	Adds a replication job template. The user also has the option of running the job immediately after the template is created.
Syntax	`Add-ReplicationJobTemplate [-JobName] <String> [[-JobDescription] <String>] [-JobEntity] <VMEntity> [-TargetHost] <HostEntity> [-ReplicateName] <String> [-Type] <String> [[-TargetNetworks] <List`1>] [[-PrimaryDatastore] <String>] [[-TargetDatastores] <List`1>] [[-JobSchedule] <Schedule>] [[-TransportFailover] <List`1>] [[-NotificationList] <List`1>] [-Flags] <NoFlags | Rep licatePoweredOnOnly | CheckDestinationFreeSpace | UseCompression | UpdateNotes | EnableGuestQuiescing | EnableABM | AllFlags> [[-RunJobNow] <Boolean>] [[-IncludedDisk] <List`1>] [<CommonParameters>]`
Parameters	• **JobName <string>** Job name. • **JobDescription [<string>]** Job description. • **JobEntity <VMEntity>** VM entity on which the job will be run on. • **TargetHost <HostEntity>** Target Host entity where the VM will be replicated • **ReplicateName [<string>]** Replicated VM Name • **Type <string>** Replication type. Valid types are Differential and Hybrid • **TargetNetworks [<List '1>]** Networks to use on the target host. Multiple values are seperated by a comma. • **PrimaryDatastore <string>** Primary Datastore on target host where the replicated VM will be located. • **TargetDatastores [<List '1>]** Datastores to use on the target host. Multiple values are seperated by a comma. • **JobSchedule <Schedule>** The recurrence schedule the job will be run on. • **TransportFailover [<List '1>]** Transport Failover order seperated by ',' to be attempted until the tas k is successful. Valid types are VA-HotAdd, VA-Network, COS. • **NotificationList [<string[]>]** Email notification list for the job. Seperate multiple emails by ','.

Parameters (cont.)	• **Flags <ReplicationFlags>** Replication flags: NoFlags, ReplicatePoweredOnOnly, CheckDestinationFreeSpace, UseCompression, UpdateNotes, EnableGuestQuiescing, AllFlags. M ultiple flags can be seperated by '|' • **RunJobNow [<Boolean>]** Whether to run the job immediately after the job template is created. • **IncludedDisk [<List '1>]** List of disks to include in this job. All VM disks will be replicated if none is specified. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	----------EXAMPLE 1---------- `Add-ReplicationJobTemplate -JobName vRepJob1 -JobEntity $vm -TargetHost $targethost -ReplicateName TestRep -Type Differential -Flags $flags -targetDatastores MyDatastore1, MyDatastore2 -PrimaryDatastore MyDatastore -targetnet works VM Network -TransportFailover VA-HotAdd, VA-Network -RunJobNow 1 -includeddisk 0,2,4`  This command adds a replication job template.  ----------EXAMPLE 2----------  `Add-ReplicationJobTemplate -JobName vRepJob1 -JobEntity $vm -TargetHost $targethost -ReplicateName TestRep -Type Differential -Flags $flags -targetDatastores MyDatastore1, MyDatastore2 -PrimaryDatastore MyDatastore -targetnetworks VM Network -includeddisk 0,1,2` This command adds a replication job template. Since the RunJobNow flag is not specified the job will not be run immediately. Since no transport failover is specified the default order of transport failovers will be used.
Related Links	Add-BackupJobTemplate Add-RestoreJobTemplate Update-ReplicationJobTemplate Get-Job Get-JobTemplate

Add-RestoreFromManifestJobTemplate

Description	Adds a restore job template from a manifest file. The manifest file can be a vRanger manifest file or a Legacy vRanger manifest file. Users also have the option of running the job immediately after the template is created.
Syntax	`Add-RestorefromManifestJobTemplate [-JobName] <String> [[-JobDescription] < String>] [-Username] <String> [-Password] <String> [-ManifestPath] <String> [[-EncryptionPassword] <String>] [[-JobSchedule] <Schedule>] [[-Flags] <No Flags | ForceOverwrite | ForcePowerOn | UseFiber | PerformNetworkRestoreOnFailure | AllFlags>] [[-NotificationList] <List`1>] [[-RunJobNow] <Boolean>] [[-VMName] <String>] [[-TargetHost] <String>] [[-PrimaryDatastore] <String >] [[-SkipDisks] <List`1>] [[-TargetDatastores] <List`1>] [[-TargetNetworks] <List`1>] [[-SkipAdapters] <List`1>] [[-TransportConfiguration] <TransportConfiguration>] [<CommonParameters>]`

Parameters	• **JobName <String>** Job name • **JobDescription [<String>]** Job description • **Username <String>** Username • **Password <String>** Password • **ManifestPath <String>** Path for the manifest file • **EncryptionPassword [<String>]** Encryption Password • **JobSchedule [<Schedule>]** Schedule on which the job will be run. If not set, the job will be ondemand. • **Flags [<RestoreFlags>]** Restore flags. If not set, the job will have no flags. • **NotificationList [<List`1>]** Email addresses seperated by comma. • **RunJobNow [<Boolean>]** Whether to run the job now. If not set, the job will not be run immediately after creation of the template. • **VMName [<String>]** Name of the restored VM. If not set, the VM name in the savepiont will be used • **TargetHost [<String>]** Fully qualified domain name or IP address of the target host. • **PrimaryDatastore [<String>]** Datastore on target host where the restored VM is located. If not set, the original primary datastore will be used when restoring to the original VM; otherwise the application will randomly choose a datastore attached to the target host. • **SkipDisks [<List`1>]** Sequence number (base 0) of the disks to skip from the savepoint. Disks in savepiont are order by their SCSI controller number and disk number. Multiple numbers are seperated by comma. • **TargetDatastores [<List`1>]** Datastores to use on the target host. Multiple values are seperated by comma. If only one is supplied, it will be used for all disks on the VM. If not set, the original datastores will be used when restoring to the original VM; otherwise the application will randomly choose a datastore attached to the target host.

Parameters (cont.)	• **TargetNetworks [<List`1>]** Networks to use on the target host. Multiple values are separated by comma. If only one network name is provided, it will be used for all adapters on the VM. If not set, the original networks will be used when restoring to the original VM; otherwise the application will randomly choose a network attached to the target host. • **SkipAdapters [<List`1>]** Sequence number (base 0) of the adapters to skip in the restore job. Multiple numbers are seperated by comma. • **TransportConfiguration [<TransportConfiguration>]** The transport configuration options to use for this job. • **<CommonParameters>** This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer and OutVariable. For more information, type, "get-help about_commonparameters".

Examples

```
 ------------------EXAMPLE 1-----------------------
Add-RestorefromManifestJobTemplate -
jobnamevapirestore_manifest -username
user1 -password Password1 -encryptionpassword encryptPwd -
VmName Test -manifestpath \\pathtomanifestfile\XYZ-
FreeDos_501ec05f-6088-5a14-8ae5-483bef2e0 54f\XYZ-
FreeDos_20100302_085918_22c5f3ee-9a09-47ae-8a70-
dae9233e5776\XYZ-Fr eeDos_20100302_085918_F.Manifest.metadata
-targethost $targethost.name -targetDatastores $targetds.name
-PrimaryDatastore $targetds.name -targetnetworks
$targetnw.name -Flags $flags -RunJobNow 1
```

This command creates a restore job template from a encrypted savepoint in the manifest file that restores to the specified target host. It also specifies datastore and network to use on the target host.

```
 ------------------EXAMPLE 2-----------------------
$repo = get-repository |where{$_.name -eq 'local'}
$targethost = get-inventoryentity -type EsxHost |
where{$_.name -eq 'dev015.dev.eng.int'}
$targetds = get-datastore $targethost | where{$_.Name -eq
'Dynamo-ISCSI'}$targetnw = get-network -host
$targethost|where{$_.Name -eq 'Virtual Machine Network'}
$flags = New-RestoreFlag -ForceOverwrite 1 Add-
RestorefromManifestJobTemplate -jobname vapirestore_manifest -
username user1 -password Password1 -VmName Test -manifestpath
\\pathtomanifestfile\X YZ-FreeDos_501ec05f-6088-5a14-8ae5-
483bef2e054f\XYZ-FreeDos_20100302_085918_22c5f3ee-9a09-47ae-
8a70-dae9233e5776\XYZ-
FreeDos_20100302_085918_F.Manifest.metadata -targethost
$targethost.name -targetDatastores $targetds.name -
PrimaryDatastore $targetds.name -targetnetworks $targetnw.name
-Flags $flags -RunJobNow 1
```

This command creates a restore job template from a manifest file that restores to the specified target host. It also specifies datastore and network to use on the target host.

```
------------------EXAMPLE 3-----------------------
$skipdisks = '0'
$skipnetworks = '0'
Add-RestorefromManifestJobTemplate -jobname
vapirestore_manifest -username
user1 -password Password1 -VmName Test -manifestpath
\\pathtomanifestfile\XYZ-FreeDos_501ec05f-6088-5a14-8ae5-
483bef2e054f\XYZ-FreeDos_20100302_085918_22c5f3ee-9a09-47ae-
8a70-dae9233e5776\XYZ-
FreeDos_20100302_085918_F.Manifest.metadata -targethost
$targethost.name -targetDatastores $targetds.name -
PrimaryDatastore $targetds.name -targetnetworks $targetnw.name
-Flags $flags -SkipDisks $skipdisks -Skipadapters
$skipnetworks -RunJobNow 1
```

Examples (cont'd)	This command creates a restore job template from a manifest file that restores to the specified target host. It also specifies datastore and network to use on the target host. The first disk (0) in the savepoint and first adapter of the VM will be skipped by the restore job.
	
	`------------------EXAMPLE 4----------------------`
	`Add-RestorefromManifestJobTemplate -jobname vapirestore_manifest -username user1 -password Password1 -VmName Test -manifestpath \\pathtomanifestfile\pintsize_20090724093457\pintsize.info -targethost $targethost.name -targetDatastores $targetds.name -PrimaryDatastore $targetds.name -targetnetworks $targetnw.name -Flags $flags -RunJobNow 1`
	
	This command creates a restore job template from a legacy manifest file that restores to the specified target host. It also specifies datastore and network to use on the target host.

Related Links	Add-RestoreJobTemplate

Add-RestoreJobTemplate

Description	Adds a restore job template. The user also has the option of running the job immediately after the template is created.
Syntax	Add-RestoreJobTemplate [-JobName] <string> [[-JobDescription] [<string>]] [-SavePoint] <SavePoint> [[-JobSchedule] [<Schedule>]] [[-Flags] [<RestoreFl ags>]] [[-NotificationList] [<string[]>]] [[-VMName] [<string>]] [[-RunJobN ow] [<Boolean>]] [[-TargetHost] [<string>]] [[-PrimaryDatastore] [<string>]] [[-SkipDisks] [<string[]>]] [[-TargetDatastores] [<string[]>]] [[-TargetNetworks] [<string[]>]] [[-SkipAdapters] [<string[]>]] [<CommonParameters>]
Parameters	• **JobName <string>** Job name. • **JobDescription [<string>]** Job description. • **SavePoint <SavePoint>** Savepoint to restore. • **VMName [<string>]** Name of the restored VM. If not set, the VM name in the savepiont will be used. • **JobSchedule [<Schedule>]** Schedule on which the job will be run. If not set, the job will be on demand. • **Flags [<RestoreFlags>]** Restore flags. If not set, NoFlag. Restore flags: NoFlags, AllFlags, ForceOverwrite, ForcePowerOn. Multiple flags can be seperated by '|' • **NotificationList [<string[]>]** Email notification list for the job. Seperate multiple emails by ','. • **RunJobNow [<Boolean>]** Whether to run the job now. If not set, the job will not be run immediately after creation of the template. • **TargetHost [<string>]** Fully qualified domain name or IP address of the target host. • **PrimaryDatastore [<string>]** Datastore on target host where the restored VM is located. If not set, the original primary datastore will be used when restoring to the original VM; otherwise the application will randomly choose a datastore attached to the target host. • **SkipDisks [<string[]>]** Sequence number (base 0) of the disks to skip from the savepoint. Disks in savepiont are order by their SCSI controller number and disk number. Multiple numbers are seperated by comma.

Paramters (cont.)	• **TargetDatastores [<string[]>]** Datastores to use on the target host. Multiple values are seperated by comma. If only one is supplied, it will be used for all disks on the V M. If not set, the original datastores will be used when restoring to the original VM; otherwise the application will randomly choose a data store attached to the target host. • **TargetNetworks [<string[]>]** Networks to use on the target host. Multiple values are seperated by comma. If only one network name is provided, it will be used for all adapters on the VM. If not set, the original networks will be used when restoring to the original VM; otherwise the application will randomly choose a network attached to the target host. • **SkipAdapters [<string[]>]** Sequence number (base 0) of the adapters to skip in the restore job. Multiple numbers are seperated by comma. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`$repo = get-repository \|where{$_.name -eq 'local'}`
	`$spList = get-repositorysavepoint $repo.id`
	`$mysp = ($spList)[1]`
	`add-restorejobtemplate -jobname vapirestore -savepoint $mysp`
	
	This command creates a restore job template that restore to the original VM in the savepoint. The job will run on-demand.
	
	---------- EXAMPLE 2 ----------
	`$repo = get-repository \|where{$_.name -eq 'local'}`
	`$sp = get-repositorysavepoint $repo.id`
	`$mysp = ($sp)[0]$targethost = get-inventoryentity -type EsxHost \| where{$_.name -eq 'dev015 .dev.eng.int'}$targetds = get-datastore $targethost \| where{$_.Name -eq 'Dynamo-ISCSI'}$targetnw = get-network -host $targethost\|where{$_.Name -eq 'Virtual Machine Network'}add-restorejobtemplate -jobname vapirestore_totaraget -savepoint $mysp -targethost $targethost.name -targetDatastores $targetds.name -PrimaryDatastore $targetds.name -targetnetworks $targetnw.name`
	
	This command creates a restore job template that restore to the specified target host. It also specifies datastore and network to use on the target host.

Examples (cont.)	---------- EXAMPLE 3 ----------
	`$repo = get-repository \|where{$_.name -eq 'local'}`
	`$sp = get-repositorysavepoint $repo.id`
	`$mysp = ($sp)[0]$targethost = get-inventoryentity -type EsxHost \| where{$_.name -eq 'dev015 .dev.eng.int'}$targetds = get-datastore $targethost \| where{$_.Name -eq 'Dynamo-ISCSI'}$targetnw = get-network -host $targethost\|where{$_.Name -eq 'Virtual Machine Network'}`
	`$skipdisks = '0'add-restorejobtemplate -jobname vapirestore_skipdisk -savepoint $mysp -targethost $targethost.name -targetDatastores $targetds.name - PrimaryDatastore $targetds.name -targetnetworks $targetnw.name -SkipDisks $skipdisks`
	
	This command creates a restore job template that restore to the specified target host. The first disk (0) in the savepoint will be skipped by the restore job.
	
	---------- EXAMPLE 4 ----------
	`$repo = get-repository \|where{$_.name -eq 'local'}`
	`$sp = get-repositorysavepoint $repo.id$mysp = ($sp)[0]`
	`$targethost = get-inventoryentity -type EsxHost \| where{$_.name -eq 'abc.my domain'}$targetds = get-datastore $targethost \| where{$_.Name -eq 'ISCSIxyz'}$targetnw = get-network -host $targethost\|where{$_.Name -eq 'Virtual Machine Network'}$skipnetworks = '0'`
	`$skipnetworksadd-restorejobtemplate -jobname vapirestore_skipadapter -savepoint $mysp - targethost $targethost.name -targetDatastores $targetds.name - PrimaryDatast ore $targetds.name -targetnetworks $targetnw.name -Skipadapters $skipnetworks`
	
	This command creates a restore job template that restores to the specified target host. The first adapter of the VM in the savepoint will be skipped by the restore job.

Related Links	Add-BackupJobTemplate Add-ReplicationJobTemplate Update-RestoreJobTemplate Get-Job Get-JobTemplate

Add-SFTPRepository

Description	The Add-SftpRepository cmdlet adds a SFTP repository to vRanger. The repository can be later used to store backup files from backup jobs.
Syntax	Add-SftpRepository [-Server] <string> [[-Port] [<int>]] [[-TargetDirectory] [<string>]] [-User] <string> [-Password] <string> [[-TimeOut] [<short>]] [-Name] <string> [[-Description] [<string>]] [[-Overwrite] [<Boolean>]] [[-Enable Encrytpion] [<Boolean>]] [[-EncryptionPassword] [<Boolean>]] [[-BlockSize] [<int>]] [[-EnableCompress] [<Boolean>]] [[-ImportOption] [<string>]] [<CommonParameters>]

Parameters	• **Server <string>** The SFTP Server FQDN or IP. • **Port [<int>]** The SFTP Server Port. • **TargetDirectory [<string>]** The SFTP Target Directory. • **User <string>** The SFTP user name. • **Password <string>** The SFTP Password. • **TimeOut [<short>]** Timeout value. • **Name <string>** A friendly name describing this SFTP server. • **Description [<string>]** The Description describing the SFTP server. • **Overwrite [<Boolean>]** Whether to overwrite the existing repository if one exists. • **EnableEncryption [<Boolean>]** Whether to enable Encryption. • **EncryptionPassword <string>** The Encryption Password. • **BlockSize [<Boolean>]** Set Block Size (KB) • **EnableCompress [<Boolean>]** Whether to compress all backups to this repository. • **ImportOption [<string>]** Whether to import the save points as read-write or readonly. Possible options 'ReadOnly' and 'ReadWrite'. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Add-SftpRepository -Server myserver.dev.corp -TargetDirectory mydir -User myname -Password mypassword -Name myrepo` This command adds a repository on server myserver.dev.corp pointing to share named myshare using the credentials provided. Existing repositories at the location specified will be overwritten.
Related Links	Get-Repository Add-DdbRepository Add-CIFSRepository Add-NFSRepository Add-NVSDRepository Add-FTPRepository

Add-VirtualAppforLinuxFLR

Description	The Add-VirtualAppforLinuxFLR command is now obsolete and has been replaced with Add-VirtualAppforLinuxFLRVA.
Related Links	Add-VirtualAppforLinuxFLRVA

Add-VirtualAppforLinuxFLRVA

Description	Adds a virtual appliance to perform Linux FLR.
Syntax	Add-VirtualAppforLinuxFLRVA [-VirtualApplianceUUID] <string> [<CommonParameters>]
Parameters	• **VirtualApplianceUUID <string>** Virtual Appliance UUID • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Add-VirtualAppforLinuxFLRVA -VirtualApplianceUUID 564d8ab2-6c7f-5915-0a94-b224b5a71570`
Related Links	None

Add-VirtualCenter

Description	Adds a VirtualCenter to the inventory
Syntax	Add-VirtualCenter [-DNSorIP] <string> [-Username] <string> [-UserPassword]<string> [[-Port] [<int>]] [<CommonParameters>]
Parameters	• **DNSorIP <string>** The DNS or IP for the VirtualCenter. • **Username <string>** The Username for the VirtualCenter. • **UserPassword <string>** The User password for the VirtualCenter. • **Port [<int>]** The SDK Port for the VirtualCenter. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Add-VirtualCenter -DNSorIP 10.6.20.20 -Username administrator -RootPassword p@ssw0rd` This command adds a VirtualCenter to the inventory.
Related Links	Add-DdbRepository Get-InventoryEntity

Disable-Job

Description	Disables enabled jobs.
Syntax	Disable-Job [-JobTemplateIDs] <string[]> [<CommonParameters>]
Parameters	• **JobTemplateIDs <string[]>** Job Template IDs for which to enable jobs. Disable multiple jobs by passing in comma-seperated job template Ids • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`Disable-Job -JobTemplateId 'F9C9E458-8E06-498D-8C2C-03CBE5515FD4'`
	
	This command disables the job if it is enabled and returns the job template entity.
	
	---------- EXAMPLE 2 ----------
	`P$jobs = Get-JobTemplate \| where {$_.IsEnabled -eq 1}`
	`Disable-Job $jobs[0].Id`
	
	This command disables the first enabled job in the list and returns the job template entity.

Related Links	Enable-Job Run-JobsNow Get-JobTemplate Remove-JobTemplate

Dismount-LinuxVolume

Description	Unmounts VAR disk from the LinuxFLR VA.
Syntax	Dismount-LinuxVolume [<CommonParameters>]
Parameters	• **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	---------- EXAMPLE 1 ---------- `Dismount-LinuxVolume`
Related Links	Mount_LinuxVolume

Enable-Job

Description	Enables disabled jobs.
Syntax	Enable-Job [-JobTemplateIDs] <Guid[]> [<CommonParameters>]
Parameters	• **JobTemplateIDs <Guid[]>** Job Template IDs for which to enable jobs. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`Enable-Job -JobTemplateId 'F9C9E458-8E06-498D-8C2C-03CBE5515FD4'`
	
	This command enables the job if it is disabled and returns the job template entity.
	
	---------- EXAMPLE 2 ----------
	`$jobs = Get-JobTemplate \| where {$_.IsEnabled -eq 0}`
	
	`Enable-Job $jobs[0].Id`
	
	This command enables the first disabled job in the list and returns the job template entity.

Related Links	Disable-Job Run-JobsNow Get-JobTemplate Remove-JobTemplate

Get-AddressBook

Description	Gets a list of email addresses from the address book.
Syntax	Get-Addressbook [<CommonParameters>]
Parameters	• **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-Addressbook`
Related Links	New-EmailAddress

Get-BackupGroupEntity

Description	Returns a list of backup group entities.
Syntax	Get-BackupGroupEntity [<CommonParameters>]
Parameters	• **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`Get-BackupGroupEntity`
	
	This command gets a list of all backup group entities
	
	---------- EXAMPLE 2 ----------
	`Get-BackupGroupEntity | where{$_.Name -eq 'BackupGroupName'}`
	
	This command gets a list of backup groups by name

Related Links	None

Get-CatalogSearchData

Description	Returns the list of results for the matched search string, repositories and virtual machines.
Syntax	Get-CatalogSearchData [-SearchText] <string> [[-RepositoryList] [<string>]] [[-VMList] [<string>]] [<CommonParameters>]
Parameters	• **SearchText <string>** Search string • **RepositoryList [<Repository[]>]** Filter by Repositories. If not specified, all repositories will be searched. • **VMList [<VMEntity[]>]** Filter by VMs. If not specified, all VMs will be searched. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	----------EXAMPLE 1----------
	`$repo = Get-Repository \| where {$_.Name -eq 'Repo1'}`
	`$vmList = Get-InventoryEntity -Type VirtualMachine \| {$_.Name -eq 'VM1'} Get-CatalogSearchdata -SearchText abcd -RepositoryList $repo -VMList $vm`
	
	Gets the result set from the catalog database for matched search text filtered by repository and machine.
	
	----------EXAMPLE 2----------
	`Get-CatalogSearchdata -SearchText abcd`
	
	Gets the result set from the catalog database for matched search text. No repository or machine filtering is applied.

Related Links	Remove-Catalog Get-CatalogStatus Set-Cataloging

Get-CatalogStatus

Description	Gets the active status of Catalog for a list of savepoints.
Syntax	Get-CatalogStatus [-Savepoints] <Guid[]> [<CommonParameters>]
Parameters	• **Savepoints <Guid[]>** List of Savepoint IDs • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-CatalogStatus -Savepoints 5a62765e-6a04-4df4-af54-46e5a38ad876` Gets the active status of the catalog data for the associated savepoint.
Related Links	Get-CatalogSearchData Remove-Catalog Set-Cataloging

Get-ConfigOption

Description	Returns the config option values for a specified type.
Syntax	Get-Configoption [-Type] <string> [<CommonParameters>]
Parameters	• **Type <string>** Valid types are Database, Email, Resources, Timeout, General, LinuxFLR or Cataloging • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-Configoption -Type Resources`
Related Links	Set-Resources Set-Cataloging Add-VirtualAppforLinuxFLR

Get-Connection

Description	Returns connection objects to Hosts or VirtualCenters. Without parameters, it returns all configured connections in vRanger. You can also get connections by name or type. Using the two parameters in the same command will return an error.
Syntax	Get-Connection [[-Name] [<string>]] [[-Type] [<string>]] [<CommonParameters>]
Parameters	• **Name [<string>]** Host name of the connection. • **Type [<string>]** Specifies the type of connection to retrieve: VirtualCenter, EsxHost or ALL. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, “get-help about_commonparameters”.

Examples	---------- EXAMPLE 1 ----------
	`Get-Connection`
	
	`This command gets a list of all vRanger loaded connections`
	
	---------- EXAMPLE 2----------
	`Get-Connection -type EsxHost`
	
	This command gets a list of all ESX host connections
	
	---------- EXAMPLE 3----------
	`Get-Connection -Name myhost.dev.corp`
	
	This command returns a connection to the specified host

Related Links	None

Get-CurrentTemplateVersionID

Description	Returns the current template version ID of a job template based on a historic template version ID. A new template version ID is assigned to the job template when it is modified.
Syntax	Get-CurrentTemplateVersionID [-TemplateVersionID] [<string>] [<CommonParameters>]
Parameters	• **TemplateVersionID <Guid>** Template version ID. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-CurrentTemplateVersionID 6baedeed-227a-4350-ba3f-470f9e90dc07` This command returns the current template version id of the job template re ferenced by the input template version id.
Related Links	Get-Job Get-JobTemplate

Get-Datastore

Description	The Get-Datastore cmdlet returns all datastores associated with the specified host .
Syntax	Get-Datastore [-Host] <HostEntity> [<CommonParameters>]
Parameters	• **Host <HostEntity>** The host object. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-Datastore -Host $host` This command returns a list of datastores associated with the host
Related Links	Get-Network

Get-GlobalTransportFailover

Description	Gets the default Transport Failover order for replication jobs.
Syntax	Get-GlobalTransportFailover [<CommonParameters>]
Parameters	• **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-GlobalTransportFailover` This command gets the default Transport Failover order for replication jobs.
Related Links	Update-GlobalTransportFailover Get-TransportFailover

Get-InventoryEntity

Description	Returns a list of inventory entities based on the input repository type. The command will return all inventory entities if no type is provided. You can also get inventory entities by type. Note that there's a performance delay if all types are requested or no type is specified.
Syntax	`Get-InventoryEntity [[-Type] <String>] [[-Block] <Boolean>] [<CommonParameters>]`
Parameters	• **Type [<string>]** Specifies the type of inventory entities: EsxHost, VirtualCenter, VirtualMachine, ClusterComputeResource, ComputeResource, DataCenter, Folder, ResourcePool, VirtualApp or All. • **Block [<Boolean>]** If set to true, the call will block if inventory is being refreshed until it’s done. Possible timeout may occur on large inventory sets. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`Get-InventoryEntity`
	
	This command gets a list of all inventory entities
	
	---------- EXAMPLE 2----------
	`Get-InventoryEntities -Type EsxHost`
	
	This command gets a list of all loaded host entities

Related Links	

Get-IsInventoryRefreshing

Description	The Get-IsInventoryRefreshing cmdlet returns “true” or “false” based on whether the inventory is refreshing at the time the cmdlet is run.
Syntax	.
Parameters	• **This cmdlet has no parameters**
Examples	while (Get-IsInventoryRefreshing) { Write-Host "sleeping for 1 second"; Start-Sleep -s 1 }.
Related Links	

Get-Job

Description	The Get-Job cmdlet returns jobs. Without parameters, Get-Job returns all past and current jobs. You can also get jobs by JobState within a time span or JobTemplateIDs. If JobTemplateIDs are specified, no other parameters can be used.
Syntax	Get-Job [[-EndTime] [<string>]] [[-StartTime] [<string>]] [[-JobState] [<string>]] [[-JobTemplateIDs] [<string>]] [<CommonParameters>]
Parameters	• **EndTime [<string>]** End time of time period. • **StartTime [<string>]** Start time of time period • **JobState [<string>]** Job state: Queued, Running, Completed or All. • **JobTemplateIDs [<string>]** Job template IDs for which to retrieve jobs. Separate multiple IDs by ','. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`Get-Job`
	
	This command returns jobs in all states
	
	---------- EXAMPLE 2----------
	`Get-Job -JobState Running`
	
	This command returns all running jobs
	
	---------- EXAMPLE 3----------
	`Get-Job -starttime 2009-03-30 -endtime 2009-03-31 -JobState Completed`
	
	This command returns completed jobs that started during the specified time span
	
	---------- EXAMPLE 4----------
	`Get-Job d7ebc5c4-7916-44c4-a0c2-2412999735f3,d7ebc5c4-7916-44c4-a0c2-2412990005f3`
	
	This command returns jobs based on specified templates

Related Links	Get-JobTemplate

Get-JobTemplate

Description	The Get-JobTemplate cmdlet returns a list of all jobs, on demand and scheduled (job templates). Without parameters, the command will return all job templates. You can also get job templates by Type or Inventory IDs.
Syntax	`Get-InventoryEntity [[-Type] <String>] [[-Block] <Boolean>] [<CommonParameters>]`
Parameters	• **Type [<string>]** Specifies the type of job template to retrieve: Backup, Restore or All. • **InventoryIds [<string>]** The inventory nodes' virtualization object ref IDs. Separate multiple IDs by ','. • **Thin [<Boolean>]** Whether to return 'thin' version of job templates. Returning thin objects can speed up commands and avoid performance related errors. By default, full objects are returned. • **ID [<String>]** Job Template ID • **VersionID [<String>]** Job Template version ID • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`Get-JobTemplate`
	
	This command gets a list of all job templates
	
	---------- EXAMPLE 2----------
	`Get-JobTemplate -type Backup`
	
	This command gets a list of all backup jobs.
	
	---------- EXAMPLE 3----------
	`Get-JobTemplate -InventoryIds d7ebc5c4-7916-44c4-a0c2-2412999735f3,d7ebc5c4-7916-44c4-a0c2-2412990005f3 -Type Backup`
	
	This command returns backup job templates in the specified inventories

Related Links	Get-Job

Get-MonitorLog

Description	
Syntax	Get-MonitorLog [[-EventId] <Int32>] [[-StartDateTime] <DateTime>] [[-EndDateTime] <DateTime>] [-Verbose] [-Debug] [-ErrorAction <ActionPreference>] [-WarningAction <ActionPreference>] [-ErrorVariable <String>] [-WarningVariable <String>] [-OutVariable <String>] [-OutBuffer <Int32>] [-WhatIf] [-Confirm]
Parameters	
Examples	
Related Links	

Get-Network

Description	The Get-Network cmdlet returns all networks associated with the host specified.
Syntax	Get-Network [-Host] <HostEntity> [<CommonParameters>]
Parameters	• **Host <HostEntity>** The host object. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-Network -Host $host` This command returns a list of networks associated with the host
Related Links	Get-Datastore

Get-Repository

Description	Returns repositories configured in vRanger. Without parameters, "Get-Repository" gets all repositories configured in vRanger. You can also get repositories by Name or Type. Using the two parameter in the same command will return an error.
Syntax	Get-Repository [[-Id] [<string>]] [[-Type] <string>] [<CommonParameters>]
Parameters	• **Id <string>** Repository ID in GUID. • **Type <string>** Specifies the type of repository to retrieve: CIFS, SFTP, FTP, NFS ,NVSD or ALL. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`Get-Repository`
	
	This command gets a list of all vRanger repositories
	
	---------- EXAMPLE 2----------
	`Get-Repository -id 034538e5-58a7-4d28-89c9-3249b620064a`
	
	This command a repository with the specified ID
	
	---------- EXAMPLE 3----------
	`Get-Repository -type CIFS`
	
	This command gets a list of all CIFS repositories.

Related Links	Add-CIFSRepository Add-DdbRepository Add-FTPRepository Add-NFSRepository Add-SFTPRepository Remove-Repository

Get-RepositoryJob

Description	The Get-RepositoryJob cmdlet returns a list of associated jobs based on the input repository ID.
Syntax	Get-RepositoryJob [-RepositoryId] <string> [<CommonParameters>]
Parameters	• **RepositoryId <string>** Repository ID in GUID. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-RepositoryJob -RepositoryId 034538e5-58a7-4d28-89c9-3249b620064a` This command returns a list of jobs associated with the repository with the specified ID
Related Links	Get-Job Get-JobTemplate

Get-RepositorySavePoint

Description	The Get-RepositorySavePoint cmdlet returns a list of savepoints in the repository. All savepoints in the repository will be returned if no time parameter is specified
Syntax	Get-RepositorySavePoint [-Id] <string> [[-StartTime] [<string>]] [[-EndTime] [<string>]] [<CommonParameters>]
Parameters	• **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-RepositorySavePoint -RepositoryId 034538e5-58a7-4d28-89c9-3249b620064a` This command returns a list of savepoints associated with the repository with the specified ID
Related Links	None

Get-SavepointDisk

Description	Returns the VM disks that the savepoint contains.
Syntax	-SavepointDisk [-Id] <String> [-WhatIf] [-Confirm] [<CommonParameters>]
Parameters	• **Id <String>** SavePoint ID in GUID format • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-SavePointDisk 91d27db5-4562-4456-918d-b0dc587e0e7f` This command returns all disks that has been backed up into the savepoint specified
Related Links	Get-VMDKVolume

Get-SavepointManifest

Description	Get-SavePointManifest cmdlet returns manifest of the savepoint
Syntax	Get-SavePointManifest [-ID] <ID> [<CommonParameters>]
Parameters	• **ID <ID>** The SavePoint ID. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-SavePointManifest $id` This command returns savepoint manifest object
Related Links	Get-RepositorySavePoint Get-VMDKVolume

Get-Savepoints

Description	Returns a list of savepoints for a specified BackupJobTemplateID
Syntax	Get-SavePoints [-BackupJobTemplateID] <Guid> [<CommonParameters>]
Parameters	• **BackupJobTemplateID <Guid>** The backup job template ID to retrieve all savepoints for. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-SavePoints -BackupJobTemplateID 034538e2-58a7-4d28-89c9-3249b620064a` This command returns a list of savepoints associated with the backup template with the specified ID.
Related Links	Get-Job Get-JobTemplate

Get-TransportFailover

Description	Gets the Transport Failover order for replication jobs by TemplateVersion ID.
Syntax	Get-TransportFailover [-TemplateVersionID] <string> [<CommonParameters>]
Parameters	• **TemplateVersionID <string>** Template Version Id for job. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	`$jobtemplates = Get-JobTemplate -Type Replication`
	`$template = $jobtemplates[0]`
	`Get-TransportFailover -TemplateVersionID $template.TemplateVersionId`
	
	Gets the Transport Failover order for replication jobs by TemplateVersion ID.

Related Links	Get-GlobalTransportFailover Update-GlobalTransportFailover

Get-VirtualApplianceConfig

Description	Gets the VirtualApplianceConfig configuration from vRanger.
Syntax	Get-VirtualApplianceConfig [[-VAName] [<string>]] [[-HostName] [<string>]][<CommonParameters>]
Parameters	• **VAName [<string>]** The VirtualAppliance Name. • **HostName [<string>]** The Host Name where the VirtualAppliance resides. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`$va = Get-VirtualApplianceConfig`
	
	This command gets all the VirtualAppliance entities.
	
	---------- EXAMPLE 2 ----------
	`$va = Get-VirtualApplianceConfig -VAName ABC-Vzva-1.1.1`
	
	This command gets the VirtualAppliance entity for the VA specified.
	
	---------- EXAMPLE 3 ----------
	`$va = Get-`
	`VirtualApplianceConfig -HostName xyz.dev.eng.int`
	
	This command gets the VirtualAppliance entities for the Host specified.

Related Links	New-VirtualAppliance Update-VirtualAppliance Remove-VirtualAppliance

Get-VirtualMachinesUnderInventory

Description	Gets the child VM entities of the entity (host, cluster, etc) provided..
Syntax	Get-VirtualMachinesUnderInventory [[-InventoryEntityMoRefId] [<string>]] [<CommonParameters>]
Parameters	• **InventoryEntityMoRefId [<string>]** The MoRef ID of the parent inventory entity. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	
	`Add-BackupGroupEntity -BackupGroupName backupGroup`
	`$backupGroup = Get-BackupGroupEntity \| where {$_.Name -eq 'backupGroup'}`
	`$vms = $vms=Get-VirtualMachinesUnderInventoryEntity -InventoryEntityMoRefId resgroup-878`
	`foreach($vm in $vms) { $entity=Get-InventoryEntity -Type virtualmachine \| where {$_.Name -eq $vm.Name}; New-BackupGroupMember -BackupGroupId $backupGroup.EntityId -MemberEntity $entity }`

Related Links	Add-BackupGroupEntity Get-BackupGroupEntity Get-InventoryEntity

Get-VmDisk

Description	The Get-VmDisk cmdlet returns a list of VmDisks for the VM list in the same order of the input.
Syntax	Get-VmDisk [-VMList] <VMEntity[]>] [<CommonParameters>]
Parameters	• **VMList <VMEntity[]>** A list of VM entities to retrieve disk information for. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-VmDisk -VMList $mylist` This command gets a list of VMDisks objects for the input VM list
Related Links	None

Get-VMDKVolume

Description	Returns all volumes on a certain VM disk.
Syntax	Get-VMDKVolume [-VmdkID] <String> [-Repository] <Repository> [-RepositoryPassword] <String>[SavepointManifest] <ManifestFile> [<CommonParameters>]
Parameters	• **VmdkID <String>** UUID of the VMDK • **Repository <Repository>** Repository in which the SavePoint resides in • **RepositoryPassword <String>** Repository password • SavepointManifest <ManifestFile> The savepoint manifest object • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Get-VMDKVolume 05ce535a-13cc-4959-9da2-85cea751958a` This command returns all volumes of the specified VMDK
Related Links	None

Install-VirtualAppliance

Description	Deploys a new virtual appliance for vRanger.
Syntax	Install-VirtualAppliance [-VAName] <String> [-SelectedHost] <HostEntity> [- OVFLocation] <String> [-DownloadOVF] <Boolean> [[-IPAddressOverride] <String>] [[-ClusterDefault] <Boolean>] [-CPUAllocation] <Int32> [-MemorySizeInMB] <Int64> [-VADatastore] <Datastore> [[-ScratchDiskDatastore] <Datastore>] [[-ScratchDiskSizeInKB] <Int64>] [[-NewPassword] <String>] [[-UseVAForLFLR] <Boolean>] [[-PowerOnVA] <Boolean>] [<CommonParameters>]

Parameters	• **VAName <String>** The VirtualAppliance name • **SelectedHost <HostEntity>** The host to associate the VirtualAppliance with • **OVFLocation <String>** The path to the local OVF • **DownloadOVF <Boolean>** Should the OVF be downloaded? • **IPAddressOverride [<String>]** IP Address to use for the VirtualAppliance instead of the one retrieved from the Virtual Center API. • **ClusterDefault [<Boolean>]** If the selected host belongs to a cluster, share the VirtualAppliance among all hosts in the cluster • **CPUAllocation <Int32>** Allocate CPU • **MemorySizeInMB <Int64>** Assign memory in MB • **VADatastore <Datastore>** Select a datastore for the VirtualAppliance. • **ScratchDiskDatastore [<Datastore>]** Select a datastore for the ScratchDisk. • **ScratchDiskSizeInKB [<Int64>]** Assign ScratchDisk in KB. • **NewPassword [<String>]** New password for virtualAppliance,if this is not mentioned, Original password (vzroot1) is used • **UseVAForLFLR [<Boolean>]** Use VirtualAppliance for Linux FLR. • **PowerOnVA [<Boolean>]** Power On VirtualAppliance after deployment complete • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	$host = Get-InventoryEntity -Type esxhost \|where{$_.Name -eq 'hostX.dev.eng.int'} $ds=Get-Datastore $host \|where{$_.Name -eq 'DEV-iSCSI01'} Install-VirtualAppliance -VAName vapiVA -SelectedHost $host -OVFLocation $p athToLocalOVF -DownloadOVF 0 -CPUAllocation 1 -MemorySizeInMB 512 -VADatastore $ds -PowerOnVA 1 This command deploys a new Virtual Appliance configuration to vRanger using a locally saved OVF.

Related Links	Get-VirtualApplianceConfig Remove-VirtualAppliance Update-VirtualAppliance

Mount_LinuxVolume

Description	Mounts a volume from the specified manifest FileMaps.
Syntax	Mount-LinuxVolume [-ManifestFileMaps] <FileMapEntryLine[]> [-Repository] <Repository> [-RepositoryPassword] <String> [<CommonParameters>]
Parameters	• **ManifestFileMaps <FileMapEntryLine[]>** The savepoint manifest file map • **Repository <Repository>** The repository containing the VAR • **RepositoryPassword<string>** The password to the repository • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`$path = Mount-LinuxVolume  -repositorypassword xxxxxxxx -manifestfilemap [filemap] -repository [repo]`
Related Links	Dismount-LinuxVolume

Move-ToolsToLinuxFLRVA

Description	Adds the Linux FLR tools to the specified VA.
Syntax	Move-ToolsToLinuxFLRVA [-VirtualApplianceUUID] <string> [<CommonParameters>]
Parameters	• **VirtualApplianceUUID <string>** The Virtual Appliance UUID • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Move-ToolsToLinuxFLRVA -VirtualApplianceUUID 564d8ab2-6c7f-5915-0a94-b224b5a71570`
Related Links	None

Move-ToolsToVA

Description	The Move-ToolsToVA command is now obsolete and has been replaced with Move-ToolsToLinuxFLRVA.
Related Links	Move-ToolsToLinuxFLRVA

New-BackupFlag

Description	The New-BackupFlag cmdlet creates a new backup flag for later use.
Syntax	New-BackupFlag [<CommonParameters>]
Parameters	• **BackupPoweredOnOnly [<Boolean>]** Whether to backup only virtual machines that are powered on. • **CheckDestinationFreeSpace [<Boolean>]** Whether to check repository free space before backup. • **UseCompression [<Boolean>]** Whether to use compression for backup. • **UpdateNotes [<Boolean>]** Whether to update notes on VM after backup. • **EnableGuestQuiescing [<Boolean>]** Whether guest quiescing is enabled. • **EnableABM [<Boolean>]** Whether Active Block Mapping is enabled. • **PerformNetworkBackupOnFailure [<Boolean>]** Whether to perform the backup using the network if there is a problem during the fiber or iSCSI backup. • **UseFiber [<Boolean>]** [Deprecated] Formerly, whether to use fiber or iSCSI to perform this backup. Currently, whether to perform a Lan-Free backup. Use **LanFree** instead. • **UseCataloging [<Boolean>]** Whether to use cataloging to perform this backup. • **LanFree [<Boolean>]** Whether to perform a Lan-Free backup. • **AllFlags [<Boolean>]** Whether to use all available flags for the backup. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`New-BackupFlag`
	This command returns a new BackupFlags object with the value of no flags.
	---------- EXAMPLE 2----------
	`New-BackupFlag -EnableGuestQuiescing 1 -UseCompression 1`
	This command returns a new BackupFlags object with the values of EnableGuestQuiescing and UseCompression

Related Links	Add-BackupJobTemplate

New-BackupGroupMember

Description	The New-BackupGroupMember cmdlet is used to add Members to a Backup Group created using Add-BackupGroupEntity cmdlet or UI. User can add any Inventory item as a member.
Syntax	New-BackupGroupMember [-BackupGroupId] <Guid> [-MemberEntity] <Entity> [<CommonParameters>]
Parameters	• **BackupGroupId <Guid>** The BackupGroup Id • **MemberEntity <Entity>** Inventory entity to add to Backup Group • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`$vm = Get-InventoryEntity \| where {$_.Name -eq 'myVM'}New-BackupGroupMember -BackupGroupId 5a62765e-6a04-4df4-af54-46e5a38ad876 -MemberEntity $vm`
Related Links	None

New-Daily Schedule

Description	Returns a Schedule object. The schedule provides a template daily schedule that can be further customized before use.Returns a Schedule object. The schedule provides a template daily schedule that can be further customized before use.
Syntax	New-DailySchedule [-WeekdayOnly] <Boolean> [-StartTime] <string> [<CommonParaeters>]
Parameters	• **WeekdayOnly <Boolean>** Whether to run the job only on weekdays. • **StartTime <string>** Job start time. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`New-DailySchedule -WeekdayOnly 1 -StartTime "2009-03-31 12:00:00AM"` This command returns a daily job schdule that runs weekdays at midnight.
Related Links	None

New-EmailAddress

Description	The New-EmailAddress cmdlet is used to add a new email address to the address book.
Syntax	New-EmailAddress [-Name] <string> [-Email] <string> [<CommonParameters>]
Parameters	• **Name <string>** Name of Person • **Email <string>** Email address • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`New-EmailAddress -Name Employee1 -Email firstname.lastname@company.com`
Related Links	Get-AddressBook

New-IntervalSchedule

Description	The New-IntervalSchedule cmdlet returns a interval Schedule object.
Syntax	New-IntervalSchedule [[-IntervalDays] <int>] [[-IntervalHours] [<int>]] [-IntervalMinutes] <int> [-StartTime] <string> [[-RangeEndDate] <string>] [[-RangeByOccurrence] <int>] [<CommonParameters>]
Parameters	• **IntervalDays <int>** Number of days to use for interval schedule. Default is 0. • **IntervalHours <int>** Number of hours to use for interval schedule. Default is 0. • **IntervalMinutes <int>** Number of minutes to use for interval schedule. Default is 0. • **StartTime <string>** Job start time. • **RangeEndDate <string>** Range end date. • **RangeByOccurrence <int>** The maximum number of times a job is executed. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`New-IntervalSchedule -StartTime 2009-03-31 -IntervalHours 1`
	
	This command returns a interval job schdule that runs every hour.
	
	---------- EXAMPLE 2 ----------
	`New-IntervalSchedule -StartTime 2009-03-31 -IntervalMinutes 30 -RangeEndDate 2010-12-31`
	
	This command returns a interval job schdule that runs every 30 minutes till 2010-12-31
	
	---------- EXAMPLE 3 ----------
	
	`New-IntervalSchedule -StartTime 2009-03-31 -IntervalHours 1 -intervalMinutes 30 -RangeByOccurrence10`
	
	This command returns a interval job schdule that runs every hour and half, the job will be executed 10 times at this interval.

Related Links	New-Daily Schedule New-WeeklySchedule New-MonthlySchedule New-YearlySchedule

New-MonthlySchedule

Description	The New-MonthlySchedule cmdlet returns a Schedule object. The schedule provides a template monthly schedule that can be further customized before use.
Syntax	New-MonthlySchedule [-ExecutionDayOfMonth] <int> [-StartTime] <string> [<CommonParameters>]
Parameters	• **ExecutionDayOfMonth <int>** Day of month on which the job will be run (1-31). • **StartTime <string>** Job start time. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`New-MonthlySchedule -ExecutionDayOfMonth 1 -StartTime "2009-03-31 12:00:00AM"` This command returns weekly job schdule that runs on the first of every month at midnight
Related Links	New-Daily Schedule New-WeeklySchedule New-YearlySchedule New-IntervalSchedule

New-ReplicationFlag

Description	The New-ReplicationFlag cmdlet helps the user create a ReplicationFlags Enum object. Without parameters, the command returns NoFlags. You can pass in boolean parameters to specify which flags you want on.
Syntax	New-ReplicationFlag [[-CheckDestinationFreeSpace] [<Boolean>]] [[-UpdateNotes] [<Boolean>]] [[-EnableGuestQuiescing] [<Boolean>]] [[-EnableABM] [<Boolean>]] [[-AllFlags] [<Boolean>]] [<CommonParameters>]
Parameters	• **CheckDestinationFreeSpace [<Boolean>]** Whether to check repository free space before replication. • **UpdateNotes [<Boolean>]** Whether to update notes on VM after replication. • **EnableGuestQuiescing [<Boolean>]** Whether guest quiescing is enabled. • **EnableABM [<Boolean>]** Whether active blaock mapping is enabled. • **AllFlags [<Boolean>]** Whether to use all available flags for the replication. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, ype, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`NewReplicationFlag`
	
	This command returns a new ReplicationFlags object with the value of NoFlags
	---------- EXAMPLE 2 ----------
	`New-ReplicationFlag -EnableGuestQuiescing 1 -EnableABM 1`
	
	This command returns a new ReplicationFlags object with the values of EnableGuestQuiescing and EnableABM
	
	---------- EXAMPLE 3 ----------
	`New-ReplicationFlag -AllFlags 1`
	
	This command returns a new ReplicationFlags object with all flags set to on

Related Links	Add-ReplicationJobTemplate

New-RestoreFlag

Description	The New-RestoreFlag cmdlet helps the user create a RestoreFlags Enum object. Without parameters, the command returns NoFlags. You can pass in boolean parameters to specify which flags you want on.
Syntax	New-RestoreFlag [[-ForceOverwrite] [<Boolean>]] [[-ForcePowerOn] [<Boolean>]] [[- AllFlags] [<Boolean>]] [<CommonParameters>]
Parameters	• **ForceOverwrite [<Boolean>]** Whether to force overwrite of existing VM with same name. • **ForcePowerOn [<Boolean>]** Whether to power on the VM for restore. • **UseFiber [<Boolean>]** [Deprecated] Formerly, whether to use fiber or iSCSI to perform this restore. Currently, whether to perform a Lan-free Restore. Use LanFree instead. • **PerformNetworkRestoreOnFailure [<Boolean>]** Whether to perform a network restore on a Lan-free restore failure. • **LanFree [<Boolean>]** Whether to perform a Lan-free Restore. • **AllFlags [<Boolean>]** Whether to use all available flags for the restore. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- Example 1 ----------
	`New-RestoreFlag`
	
	This command runs RestoreFlags with the value of NoFlags
	
	---------- Example 2----------
	`New-RestoreFlag -ForcePowerOn 1 -UseCompression 0`
	
	This command runs RestoreFlags with the values of ForcePowerOn
	
	---------- Example 3----------
	`New-RestoreFlag -AllFlags 1`
	
	This command runs RestoreFlags with all flags set to on.

Related Links	None

New-SMTPServer

Description	The New-SMTPServer cmdlet is used to add a new SMTP Server.
Syntax	New-SMTPServer [-SMTPServer] <string> [-Port] <int> [-FromEmail] <string> [<CommonParameters>]
Parameters	**-SMTPServer <string>** SMTP Server address **-Port <int>** Port No for SMTP Server. **-FromEmail <string>** Email address for sent emails. **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`New-SMTPServer -SMTPServer company.smtp.com -Port 22 -FromEmail support@company.com`
Related Links	None

New-TransportConfiguration

Description	Helps the user create a TransportConfiguration object. Without parameters, the command returns the default Automatic configuration. You can pass in boolean parameters to specify which settings you want on.
Syntax	New-TransportConfiguration [-AutoSelectTransport] <Boolean> [<CommonParameters>] New-TransportConfiguration [-RunLocal] <Boolean> [-UseAdvancedTransport] <Boolean> [-UseLanTransport] <Boolean> [[-UseSsLforLan] <Boolean>] [<CommonParameters>]
Parameters	• **AutoSelectTransport <Boolean>** The service will determine which transport type to use at run-time. • **RunLocal <Boolean>** Whether to run the job on the local machine vs the VA. • **UseAdvancedTransport <Boolean>** Whether the job should attempt to use an advanced transport (SAN/HotAdd) first. • **UseLanTransport <Boolean>** Whether the job should attempt to use a network transport. • **UseSsLforLan [<Boolean>]** Whether the job should use an encrypted connection when attempting a network transport. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	`------------------EXAMPLE 1----------------------`
	`New-BackupFlag -RunLocal 1 -UseAdvancedTransport 1 UseLANTransport 0`
	
	This command returns a new TransportConfiguration object with the values of RunLocal, UseAdvancedTransport.
	
	`------------------EXAMPLE 2----------------------`
	
	`New-TransportConfiguration`
	
	This command returns a new TransportConfiguration object with the configuration set to Automatic.

Related Links	Add-BackupJobTemplate Add-RestoreJobTemplate

New-VirtualAppliance

Description	Configures a VirtualAppliance for vRanger.
Syntax	New-VirtualAppliance [-VAEntity] <Entity>[-IPAddressOverride] <string> [-RootPassword] <string> [<CommonParameters>]
Parameters	• **VAEntity <Entity>** The VM Entity where the VirtualAppliance is deployed. • **RootPassword <string>** Root Password for the VirtualAppliance. • **IPAddressOverride<string>** New IP Address for the VA • **ClusterDefault [<Boolean>]** Share the VirtualAppliance among all hosts in the cluster. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	---------- EXAMPLE 1 ---------- `$va = Get-InventoryEntity -Type VirtualMachine \| where {$_.Name -eq 'ABC-Vzva-1.1.1'} New-VirtualAppliance -VAEntity $va -IPAddressOverride 10.6.50.19 -RootPassword vzroot1` This command adds a VirtualAppliance configuration to vRanger.
Related Links	Update-VirtualAppliance Remove-VirtualAppliance Get-VirtualApplianceConfig

New-WeeklySchedule

Description	The New-WeeklySchedule cmdlet returns a Schedule object. The schedule provides a template weekly schedule that can be further customized before use.
Syntax	New-WeeklySchedule [-ExecutionDays] <string> [-StartTime] <string> [<CommonParameters>]
Parameters	• **ExecutionDays <string>** Days of the week on which the job will be run: Monday, Tuesday etc. Operate multiple days by ','. • **StartTime <string>** Job start time. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`NewWeeklySchedule -ExecutionDays "Monday" -StartTime "2009-03-31 12:00:00AM"`
	
	This command returns weekly job schdule that runs on Mondays at midnight
	
	---------- EXAMPLE 2----------
	`NewWeeklySchedule -ExecutionDays "Monday,Tuesday" -StartTime "2009-03-31 12:00:00AM"`
	
	This command returns weekly job schdule that runs on Mondays and Tuesdays at midnight

Related Links	None

New-YearlySchedule

Description	The NewYearlySchedule cmdlet returns a Schedule object. The schedule provides a template yearly schedule that can be further customized before use.
Syntax	NewYearlySchedule [-ExecutionMonth] <int> [-ExecutionDayOfMonth] <int> [-StartTime] <string> [<CommonParameters>]
Parameters	• **ExecutionMonth <int>** Month in which the job will be run (1-12). • **ExecutionDayOfMonth <int>** Day of month on which the job will be run (1-31). • **StartTime <string>** Job start time. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`NewYearlySchedule -ExecutionMonday 1 -ExecutionDayOfMonth 1 -StartTime "2009-03-31 12:00:00AM"` This command returns weekly job schdule that runs on the January 1 each year at midnight
Related Links	None

Remove-AllMount

Description	Removes all previously mounted paths. This operation should be performed as a cleanup step after all needed content is retrieved from the mounted path.
Syntax	`Remove-Allmount` [<CommonParameters>]
Parameters	• **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Remove-AllMount` This command removes all mounted paths
Related Links	None

Remove-BackupGroupEntity

Description	The Remove-BackupGroupEntity cmdlet deletes a Backup Group from vRanger. Remove-BackupGroupMember cmdlet is used to remove Members from a Backup Group created using this cmdlet or UI.
Syntax	`Remove-BackupGroupEntity [-BackupGroupId] <Guid> [<CommonParameters>]`
Parameters	• **BackupGroupId <Guid>** The BackupGroup Id • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`$bg = Get-BackupGroupEntity \| where {$_.Name -eq 'BG1'}Remove-BackupGroupEntity -BackupGroupId $bg.Id`
Related Links	N/A

Remove-BackupGroupMember

Description	The Remove-BackupGroupMember cmdlet is used to remove Members from a Backup Group.
Syntax	`Remove-BackupGroupMember [-BackupGroupId] <Guid> [-MemberEntity] <Entity> [<CommonParameters>]`
Parameters	• **BackupGroupId <Guid>** The BackupGroup Id • **MemberEntity <Entity>** Inventory entity to add to Backup Group • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`$vm = Get-InventoryEntity \| where {$_.Name -eq 'myVM'}Remove-BackupGroupMember -BackupGroupId 5a62765e-6a04-4df4-af54 -46e5a38ad876 -MemberEntity $vm`
Related Links	N/A

Remove-Catalog

Description	The Remove-Catalog cmdlet is used to delete the catalog data from the database for a list of savepoints.
Syntax	`Remove-Catalog [-Savepoints] <Guid[]> [<CommonParameters>]`
Parameters	• **Savepoints <Guid[]>** List of Savepoint Ids • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Remove-Catalog -Savepoints 5a62765e-6a04-4df4-af54-46e5a38ad876` Removes the catalog data associated with the savepoint from the database.
Related Links	Get-CatalogSearchData Get-CatalogStatus Set-Cataloging

Remove-DdbStorageUnit

Description	Removes the DDB repository from vRanger and all savepoints in the repository will also be deleted from the disk. The storage unit will be removed from the DD device.
Syntax	`Remove-DdbStorageUnit [-Repository] <Repository> [-Password] <String> [<CommonParameters>]`
Parameters	• **Repository <Repository>** The DDB repository entity to remove. • **Password <String>** Password for DD Boost user. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Remove-DdbStorageUnit -Repository $repo -Password Password123` This command removes the repository from vRanger and all associated savepoints will be deleted from the disk. The storage unit will also be removed from the DD device.
Related Links	Remove-Repository

Remove-JobTemplate

Description	The Remove-JobTemplate cmdlet removes a JobTemplate from vRanger.
Syntax	`Remove-JobTemplate [-Id] <string> [<CommonParameters>]`
Parameters	• **ID <string>** The JobTemplate ID in GUID. • **<ComMonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Remove-JobTemplate -Id $guid` This command removes the JobTemplate with the specified ID from vRanger.
Related Links	Add-BackupGroupEntity Add-RestoreJobTemplate Get-JobTemplate

Remove-LinuxVolume

Description	The Remove-LinuxVolume cmdlet unmounts a VAR disk from the LinuxFLR VA.
Syntax	`Remove-LinuxVolume [-FileUUID] <string> [<CommonParameters>]`
Parameters	• **FileUUID <string>** VAR file UUID • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Remove-LinuxVolume -fileuuid [map].DestinationFileUUID`
Related Links	None

Remove-Repository

Description	The Remove-Repository cmdlet removes a repository from vRanger. Optionally, all savepoints in the repository can be deleted from the disk.
Syntax	`Remove-Repository [-Id] <string> [[-RemoveSavePoints] [<Boolean>]] [<CommonParameters>]`
Parameters	• **ID <string>** The repository ID in GUID. • **RemoveSavePoints [<Boolean>]** Whether to remove all savepoints in the repositories. 0 or 1. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`Remove-Repository -Id $guid`
	
	This command removes the repository with the specified ID from vRanger. All savepoints associated with the repository will be un-touched.
	
	---------- EXAMPLE 2----------
	`Remove-Repository -Id $guid -RemoveSavePoints 1`
	
	This command removes the repository withe the specified ID from vRanger. All savepoint files in the repository will be removed from disk as well.

Related Links	Add-CIFSRepository Add-SFTPRepository Add-FTPRepository Add-NFSRepository Add-NVSDRepository Get-Repository

Remove-SavePoint

Description	The Remove-SavePoint cmdlet removes savepoints. If any of the specified savepoint has descendant savepoints, those savepoints will also be removed.
Syntax	`Remove-SavePoint [-SavePointsToRemove] <list<SavePoint>> [<CommonParameters>]`
Parameters	• **SavePointsToRemove <list<SavePoint>>** The list of savepoints to remove. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Remove-SavePoint -SavePointsToRemove $splist` This command removes the list of savepoints specified
Related Links	Get-RepositorySavePoint Remove-Repository

Remove-VirtualAppliance

Description	Removes the VirtualAppliance configuration from vRanger.
Syntax	`Remove-VirtualAppliance [-VAId] <string> [<CommonParameters>]`
Parameters	• **VAId <string>** The VirtualAppliance ID for VirtualAppliance to be updated. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`$va = Get-VirtualApplianceConfig -VAName KRN-Vzva-1.1.1 Remove-VirtualAppliance -VAId $va.VirtualApplianceID` This command removed a VirtualAppliance configuration.
Related Links	New-VirtualAppliance Update-VirtualAppliance Get-VirtualApplianceConfig

Run-JobsNow

Description	The Run-JobsNow cmdlet runs jobs specified by their template IDs. Use the `Get-JobTemplate` cmdlet to view the template IDs of configured job templates.
Syntax	`Run-JobsNow [-TemplateIds] <string> [<CommonParameters>]`
Parameters	• **TemplateIds <string>** The job template IDs. Seperate multiple IDs by ','. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Run-JobsNow d7ebc5c4-7916-44c4-a0c2-2412999735f3,d7ebc5c4-7916-44c4-a0c2-2412990005f3` This command runs jobs specified by their templates
Related Links	Add-BackupGroupEntity Add-RestoreJobTemplate Get-Job Get-JobTemplate

Run-ReplicationFailover

Description	The Run-ReplicationFailover cmdlet fails over the replication pass to the replica.
Syntax	`Run-ReplicationFailover [-TemplateId] <string> [[-Sync] [<string>]] [<CommonParameters>]`
Parameters	• **TemplateId <string>** Replication job template ID. • **Sync [<Boolean>]** Whether to sync replica before failing over. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Run-ReplicationFailover d7ebc5c4-7916-44c4-a0c2-2412999735f3,d7ebc5c4-7916-44c4-a0c2-2412990005f3` This command runs replication failover based on the job template id.
Related Links	Run-TestReplicationFailover Run-ResumeReplicationFailover

Run-ResumeReplicationFailover

Description	The Run-ResumeReplicationFailover cmdlet resumes test replication failover based on the test replication task id.
Syntax	Run-ResumeReplicationFailover [-TaskId] <string> [<CommonParameters>]
Parameters	• **TaskID <string>** Replication test failover task ID. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Run-ResumeReplicationFailover d7ebc5c4-7916-44c4-a0c2-2412999735f3,d7ebc5c4-7916-44c4-a0c2-2412990005f3`
Related Links	Run-TestReplicationFailover Run-ReplicationFailover

Run-TestReplicationFailover

Description	The Run-TestReplicationFailover cmdlet runs test replication failover based on the replication job's template version id.
Syntax	Run-TestReplicationFailover [-TemplateVersionId] <Guid> [[-EnableNetwork] <Boolean>] [<CommonParameters>]
Parameters	• **TemplateVersionID <string>** Replication job template version ID. • **EnableNetwork [<Boolean>]** Allow replica network to be enabled during a test failover. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Run-TestReplicationFailover d7ebc5c4-7916-44c4-a0c2-2412999735f3,d7ebc5c4-7916-44c4-a0c2-2412990005f3` This command runs test failover on a replication job
Related Links	Run-ResumeReplicationFailover Run-ReplicationFailover

Set-Cataloging

Description	The Set-Cataloging cmdlet enables/disables global cataloging flag.
Syntax	`Set-Cataloging [-Catalog] <Boolean> [<CommonParameters>]`
Parameters	• **Catalog <Boolean>** Boolean value to enable or disable global cataloging • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Set-Cataloging -Catalog 1`
Related Links	Remove-Catalog Get-CatalogSearchData Get-CatalogSearchData

Set-CBTonVM

Description	Enables/Disables block tracking for virtual machines if it is supported
Syntax	`Set-CBTonVM [-VM] <VMEntity> [-CBT] [<Boolean>] [<CommonParameters>]`
Parameters	• **VM <VMEntity>** Virtual machine for which to enable block tracking • **CBT [<Boolean>]** Enable/Disable Block Tracking • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`$vm = Get-InventoryEntity -type virtualmachine \| where{$_.Name -eq 'x yz'} Set-CBTonVM -VM $vm -CBT 1` This command enables block tracking on the virtual machine xyz if it is supported on the ESX host.
Related Links	None

Set-LinuxVolume

Description	The Set-LinuxVolume cmdlet mounts a volume from the specified manifest FileMap.
Syntax	`Set-LinuxVolume [-FileUUID] <string> [-ManifestFileMap] <FileMapEntryLine>[-Repository] <Repository> [-RepositoryPassword] <string> [<CommonParameters>]`
Parameters	• **FileUUID <string>** UUID of the VAR file • **ManifestFileMap <FileMapEntryLine>** The savepoint manifest file map • **Repository <Repository>** The repository containing the VAR • **RepositoryPassword <string>** The password to the repository • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Set-LinuxVolume -fileuuid [map].DestinationFileUUID -repositorypassword xxxxxxxx -manifestfilemap [map] -repository [repo]`
Related Links	None

Set-MountPath

Description	Sets mount path for a volume. After this operation, the specified volume will be mounted to the path specified.
Syntax	`Set-MountPath [-Volume] <Volume> [-Path] <String> [<CommonParameters>]`
Parameters	• **Volume <Volume>** The volume to be mounted • **Path <String>** The path to which the volume will be mounted to. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Set-MountPath  -volume $v -path 'E:\'` This command mount the specified volume in the savepoint to the E:\ drive
Related Links	Get-Repository Get-RepositorySavePoint Get-SavepointDisk Get-VMDKVolume Remove-AllMount

Set-Resources

Description	The Set-Resources cmdlet sets value for vRanger resources.
Syntax	`Set-Resources [[-NumberTasksVRanger] [<int>]] [[-NumberTasksLUN] [<int>]][-NumberTasksHost] [<int>]] [[-NumberTasksRepository] [<int>]] [[-NumberTaksLocal] [<int>]] [[-Timeout] [<int>]] [[-MinimumSpaceHost] <int>] [<CommoParameters>]`
Parameters	• **NumberTasksVRanger [<Int32>]** Maximum number of tasks running on vRanger (>=1) • **NumberTasksLUN [<Int32>]** Maximum number of tasks running off a LUN (1-5) • **NumberTasksHost [<Int32>]** Maximum number of tasks running on a host (1-4) • **NumberTasksRepository [<Int32>]** Maximum number of tasks running per Repository (>=1) • **NumberTasksLocal [<Int32>]** Maximum number of tasks running locally (1-20) • **Timeout [<Int32>]** Task Timeout in hours (1-999) • **MinimumSpaceHost [<Int32>]** Minimum space needed on host (GBs) • **NumberTasksVA [<Int32>]** Maximum number of tasks running on a VA (1-8) • **CommonParameters>** This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer and OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Set-Resources -NumberTasksVRanger 10 -Timeout 2` Sets the maximum tasks run on vRanger and the Timeout for each task.
Related Links	Get-ConfigOption

Stop-vRangerJob

Description	Cancels a running job.
Syntax	`Stop-vRangerJob [-JobID] <string> [<CommonParameters>]`
Parameters	• **JobID <string>** Job ID for which to cancel job. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`Stop-vRangerJob -JobId 'F9C9E458-8E06-498D-8C2C-03CBE5515FD4'`
	
	This command cancels the job if it is running and returns the success/failure message.
	
	---------- EXAMPLE 2 ----------
	`$jobs = Get-Job  \| where {$_.JobState -eq 'Running'}`
	
	`Stop-vRangerJob $jobs[0].Id`
	
	This command cancels the first running job in the list and returns the success/failure message.

Related Links	Run-JobsNow Get-Job

Update-BackupJobTemplate

Description	Updates a backup job template. The user also have the option of running the job immediately after the template is updated.
Syntax	`Update-BackupJobTemplate [-JobTemplate] <BackupJobTemplate> [[-JobName] <String>] [[-JobDescription] <String>] [[-ExcludedVMList] <List`1>] [[-IncludedDiskList] <List`1>] [[-TargetRepository] <Repository>] [[-JobSchedule] <Schedule>] [[-Flags] <NoFlags | BackupPoweredOnOnly | CheckDestinationFreeSpace | UseCompression | UpdateNotes | EnableGuestQuiescing | EnableABM | UseFiber | PerformNetworkBackupOnFailure | UseCataloging | AllFlags>] [[-NumberOfSavePoints] <Int32>] [[-SpaceSavingTechnologyTypeFlag] <None | Incremental | Differential>] [[-SpaceSavingCountThreshold] <Int32>] [[-SpaceSavingPercentSizeThreshold] <Int32>] [[-NotificationList] <List`1>] [[-RunJobNow] <Boolean>] [[-TransportConfiguration] <TransportConfiguration>] [<CommonParameters>]`

Parameters	• **JobTemplate <BackupJobTemplate>** Existing job template • **JobName [<String>]** Job name • **JobDescription [<String>]** Job description • **ExcludedVMList [<List`1>]** List of virtual machines to exclude from this job • **IncludedDiskList [<List`1>]** List of virtual disks to include in this job • **TargetRepository [<Repository>]** Repository where the backups will be stored • **JobSchedule [<Schedule>]** Schedule on which the job will be run • **Flags [<Nullable`1>]** Backup flags • **NumberOfSavePoints [<Nullable`1>]** Max number of savepoints to store before older ones are removed. • **SpaceSavingTechnologyTypeFlag [<Nullable`1>]** Spacesaving technology to use: differential, incremental or none. • **SpaceSavingCountThreshold [<Nullable`1>]** The number to save before performing a full backup. • **SpaceSavingPercentSizeThreshold [<Nullable`1>]** The size to allow as a % of the virtual machine hard disk file befoRE performing a differential backup. • **NotificationList [<List`1>]** Email notification list for the job. • **RunJobNow [<Nullable`1>]** Whether to run the job now • **TransportConfiguration [<TransportConfiguration>]** The transport configuration options to use for this job. • **<CommonParameters>** This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer and OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`get-jobtemplate backup |where{$_.jobname -eq 'my first backup job'}| Update-BackupJobTemplate -JobName 'modified by vapi'` This command adds a backup job template. Since the RunJobNow flag is not specified the job will not be run immediately.
Related Links	Add-BackupJobTemplate Get-Job Get-JobTemplate

Update-GlobalTransportFailover

Description	Updates the default Transport Failover order for replication jobs.
Syntax	`Update-GlobalTransportFailover [-TransportFailover] <string[]> [<CommonParameters>]`
Parameters	• **TransportFailover <List ‘1>** Transport Failover order seperated by ',' to be attempted until the task is successful. Valid types are VA-HotAdd, VA-Network, COS. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Update-GlobalTransportFailover -TransportFailover VA-Network, COS` This command updates the default Transport Failover order for replication jobs.
Related Links	Get-GlobalTransportFailover Get-TransportFailover

Update-Inventory

Description	Refreshes the inventory.
Syntax	`Update-Inventory [<CommonParameters>]`
Parameters	• **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`Update-Inventory` This command refreshes the inventory.
Related Links	None

Update-ReplicationJobTemplate

Description	Updates a replication job template. The user also have the option of runni ng the job immediately after the template is updated.
Syntax	`Update-ReplicationJobTemplate [-JobTemplate] <ReplicationJobTemplate> [[-JobName] [<string>]] [[-JobDescription] [<string>]] [[-TargetHost] [<HostEntity>]] [[-ReplicateName] [<string>]] [[-Type] [<string>]] [[-TargetNetworks] [<string[]>]] [[-PrimaryDatastore] [<string>]] [[-TargetDatastores] [<string[]>]] [[-JobSchedule] [<Schedule>]][[-TransportFailover] [<string[]>]] [[-NotificationList] [<string>]] [[-Flags] [<BackupFlags>]] [[-RunJobNow] [<Boolean>]] [<CommonParameters>]`

Parameters	• **JobTemplate <ReplicationJobTemplate>** Replication job template to update. • **JobName [<string>]** Job name. • **JobDescription [<string>]** Job description. • **TargetHost [<HostEntity>]** Target Host entity where the VM will be replicated • **ReplicateName [<string>]** Replicated VM Name • **Type [<string>]** Replication type. Valid types are Differential and Hybrid • **TargetNetworks <string[]>** Networks to use on the target host. Multiple values are seperated by comma. • **PrimaryDatastore [<string>]** Primary Datastore on target host where the replicated VM will be located. • **TargetDatastores [<string[]>]** Datastores to use on the target host. Multiple values are seperated by comma. • **JobSchedule [<Schedule>]** The recurrence schedule the job will be run on. • **TransportFailover [<string[]>]** Transport Failover order seperated by ',' to be attempted until the tas k is successful. Valid types are VA-HotAdd, VA-Network, COS. • **NotificationList [<string>]** Email notification list for the job. Seperate multiple emails by ','. • **Flags [<ReplicationFlags>]** Replication flags: NoFlags, ReplicatePoweredOnOnly, CheckDestinationFreeSpace, UseCompression, UpdateNotes, EnableGuestQuiescing, AllFlags. Multiple flags can be seperated by '|' • **IncludedDisk [<int[]>]** List of disks to include in this job. All VM disks will be replicated if none is specified. • **RunJobNow [<Boolean>]** Whether to run the job immediately after the job template is created. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".
Examples	`get-jobtemplate replicati on|where{$_.jobname -eq 'my first replication job'}|Update-ReplicationJobTe mplate -JobName 'new name'` This command updates a replication job template with a new name.

Related Links	Add-ReplicationJobTemplate Get-Job Get-JobTemplate

Update-RestoreJobTemplate

Description	Updates a restore job template. The user also have the option of running the job immediately after the template is updated.
Syntax	`Update-RestoreJobTemplate [-JobTemplate] <string> [[-JobName] [<string>]] [[-JobDescription] [<string>]] [[-SavePoint] [<SavePoint>]] [[-JobSchedule][<Schedule>]] [[-Flags] [<RestoreFlags>]] [[-NotificationList] [<string[]>]] [[-VMName] [<string>]] [[-RunJobNow] [<Boolean>]] [[-TargetHost] [<string>]] [[-PrimaryDatastore] [<string>]] [[-SkipDisks] [<string[]>]] [[-TargetDatastores] [<string[]>]] [[-TargetNetworks] [<string[]>]] [[-SkipAdapters] [<string[]>]] [<CommonParameters>]`
Parameters	• **JobTemplate <string>** The restore job template to update. • **JobName [<string>]** Job name. • **JobDescription [<string>]** Job description. • **SavePoint [<SavePoint>]** Savepoint to restore. This parameter is not required if the restore job template contains only one savepoint. • **VMName [<string>]** Name of the restored VM. If not set, the VM name in the savepoint will be used. • **JobSchedule [<Schedule>]** Schedule on which the job will be run. If not set, the job will be on demand. • **Flags [<RestoreFlags>]** Restore flags. If not set, NoFlag. Restore flags: NoFlags, AllFlags, ForceOverwrite, ForcePowerOn. Multiple flags can be seperated by '|' • **NotificationList [<string[]>]** Email notification list for the job. Seperate multiple emails by ','. • **RunJobNow [<Boolean>]** Whether to run the job now. If not set, the job will not be run immediately after creation of the template. • **TargetHost [<string>]** Fully qualified domain name or IP address of the target host. • **PrimaryDatastore [<string>]** Datastore on target host where the restored VM is located. If not set, the original primary datastore will be used when restoring to the original VM; otherwise the application will randomly choose a datastore att ached to the target host.

Parameters (cont.)	• **SkipDisks [<string[]>]** Sequence number (base 0) of the disks to skip from the savepoint. Disks in savepiont are order by their SCSI controller number and disk number. Multiple numbers are seperated by comma. • **TargetDatastores [<string[]>]** Datastores to use on the target host. Multiple values are seperated by a comma. If only one is supplied, it will be used for all disks on the VM. If not set, the original datastores will be used when restoring to the original VM; otherwise the application will randomly choose a data store attached to the target host. • **TargetNetworks [<string[]>]** Networks to use on the target host. Multiple values are seperated by acomma. If only one network name is provided, it will be used for all adapters on the VM. If not set, the original networks will be used when restoring to the original VM; otherwise the application will randomly choose a network attached to the target host. • **SkipAdapters [<string[]>]** Sequence number (base 0) of the adapters to skip in the restore job. Multiple numbers are seperated by comma. • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	---------- EXAMPLE 1 ----------
	`$temp = get-jobtemplate restore |where{$_.jobname -eq 'my restore job'}`
	`update-restorejobtemplate -jobtemplate $temp -jobname vapirestore`
	
	This command updates a restore job template name.
	
	---------- EXAMPLE 2 ----------
	`$savepoint = get-repository |where{$_.name -eq 'local'}|get-repositorysavepoint |where{$_.VMUUID -eq '12312312312313'}`
	`get-jobtemplate restore |where{$_.jobname -eq 'my restore job'}|update-rest`
	`orejobtemplate -savepoint $savepoint`
	
	This command updates a restore job template to restore a different savepoint.

Related Links	Add-ReplicationJobTemplate Get-Job Get-JobTemplate

Update-VirtualAppliance

Description	Updates the configuration of a VirtualAppliance for vRanger.
Syntax	`Update-VirtualAppliance [-VAId] <string> [-IPAddressOverride] <string> [-RootPassword] <string> [<CommonParameters>]`
Parameters	• **VAId <string>** The VirtualAppliance ID for VirtualAppliance to be updated. • **RootPassword <string>** Root Password for the VirtualAppliance. • **IPAddressOverride<string>** New IP Address for the VA • **<CommonParameters>** This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, and -OutVariable. For more information, type, "get-help about_commonparameters".

Examples	`$va = Get-VirtualApplianceConfig -VAName KRN-Vzva-1.1.1`
	`Update-VirtualAppliance -VAId $va.VirtualApplianceID -IPAddressOverride 10.6.50.19 -RootPassword vzroot1`
	
	This command updates the password for the VirtualAppliance.

Related Links	New-VirtualAppliance Remove-VirtualAppliance Get-VirtualApplianceConfig

Appendix B: Using the VI Editor

This information is a summary of VI functionality, intended to satisfy the basic needs of a user new to VI that just wants to edit the required files for the vRanger virtual appliance configuration. Much more information can be found freely on the internet by searching for “vi editor commands”.

Note Linux commands and file names are case sensitive.

VI Editor Overview

VI is a modal text editor commonly used in Linux and Unix environments. VI operates in either *insert mode* (where keystrokes are entered as part of the document) or *command mode* (where keystrokes are interpreted as commands). The dual modes of VI allow users to perform both text editing and command operations without removing their hands from the keyboards home row. The VI editor starts in command mode.

Starting VI

When starting the VI editor, you may create a new file or open an existing file.

- The command `vi` without any file name will open a new file. You will need to name and save the file when editing is complete.
- The command `vi` followed by an existing file name will open that file for editing.

Changing Modes

Files are opened by the VI editor in command mode, which means that input from the keyboard will be treated as VI commands rather than text. Before editing the file, you must change the VI editor to insert mode. The most commands to get into insert mode are listed below. You may also use any of the insertion commands listed in “Inserting Text” on page 302.

Command	Description

a	enter insert mode; the characters typed will be inserted after the current cursor position.
Esc	switches back to command mode

Editing Commands

The sections below contain basic information about navigating the file and editing the text within.

Moving the Cursor

Moving the cursor around a file using the VI editor can be a challenge to those not familiar with the commands. Use the commands below as a quick reference.

Command	Description
h	moves the cursor one space to the left
j	moves the cursor one line down
k	moves the cursor one line up
l	moves the cursor one space to the right
$	moves the cursor to the end of the current line
^	moves the cursor to the beginning of the current line
Enter	moves the cursor to the beginning of the first word on the next line
G	moves the cursor to the end of the file
:*n*	moves the cursor to line *n*
w	moves the cursor to the beginning of the next word
e	moves the cursor to the end of the next word
b	moves the cursor to the beginning of the previous word

Inserting Text

Use the commands below to insert text into the open file.

Command	Description
a	append text starting right of cursor
A	append at the end of the current line
i	insert text starting left of cursor
I	insert text at beginning of the current line
o	open line below cursor, then enter insert mode
O	open line above cursor, then enter insert mode

Deleting Text

Use the commands below to delete text from the file.

Command	Description
x	delete single character
dw	delete word
dd	delete line
s	delete character, remain in insert mode
cw	delete word, remain in insert mode
cc	delete line, remain in insert mode
u	undo last change
U	undo all changes to current line

Saving Your Changes

VI will not automatically save your changes. To save your changes, or leave the file without saving changes, use the appropriate command below:

Command	Description

:w	writes the changes to disk
:zz	writes the changes to disk and quits VI
:q!	quits without saving changes.

Index

A

About 9
  vRanger 9
Active Block Mapping (ABM) 14
  in backups 106, 158
  in replication 150

B

Backup
  custom backup group 96
  overview 87
  retention policies 106
  transport options 87
    selection order 89
backup process
  dfferentials 112
  incrementals 111
  limitations 87

C

Catalog
  filtering 38
  search 130
Cataloging 81, 106
Change Block Tracking
  enabling 113
configuration
  options 37
  with Startup Wizard 19
  without Startup Wizard 28
Custom Backup Group 96
Custom Transport
  enabling/disabling 152

D

Data Domain 191
deduplication 190

E

encryption 15
ESXi
  root 23

F

Failback 165
Failover 162
File-Level Restore 130
  catalog search 130
  from manifest 132
  limitations 130
  linux 132

H

Help
  about
    System Info 66
HotAdd
  limitations 88

L

Licensing 9
  host licensing 32
  importing a file 31
Linux FLR
  configuration 36
  limitations 132

P

PowerShell 191
  Console 178
  execution policy 178
  vRanger commandlets 179

R

Replication
  about 145
  creating a job 151
  ESXi 147
  failback 165
  failover 153

modes 148
differential 149
Hybrid 149
pre-seeding 152
requirements 145
via VA 147
VM hardware changes 148
with CBT 149
with custom transports 152
Reports
My Reports View 168
overview 168
Report Wizard 170
Repositories
adding 25
encrypting 26
repositories
using a Data Domain appliance 191
repository
DD Boost 43
FTP 40
NFS 39
NVSD 42
SFTP 41
Restore
file-level 130
catalog search 130
from manifest 132
limitations 130
linux 132
full restore 118
from manifest 130
overview 118
retention policies 106
root
with ESXi 23

S

Startup Wizard
add repositories 25
email notification 27
select a license 20
Support
getting system info 66
support 11
System Info 66

T

tape backups 184
technical support 11
Test Failover 162
text conventions 9
transport selection
automatic 103
custom 103

V

VI
using the VI editor 300
Views
My Inventory 67
My Jobs 75
My Reports 83
My Repositories 74
virtual appliance
adding to vRanger 58
changing scratch disk size 62
deploying mutltiple instances 49
disabling DRS 58, 63
required configurations 50
sizing the scratch disk 47
VA utility script 53
vRanger
and Data Domain 191
and PowerShell 191
and tape backup vendors 184
interface 65
licensing 9
overview 13
vRanger Console 178