Dell™ AppAssure™

User Guide

5.4.3 Revision B

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

Dell Inc.
Attn: LEGAL Dept
5 Polaris Way
Aliso Viejo, CA 92656

Refer to our web site (software.dell.com) for regional and international office information.

Legend

CAUTION: A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.

WARNING: A WARNING icon indicates a potential for property damage, personal injury, or death.

IMPORTANT NOTE, **NOTE**, **TIP**, **MOBILE**, or **VIDEO:** An information icon indicates supporting information.

Dell AppAssure User Guide
Updated - April 2015
Software Version - 5.4.3 Revision B

Contents

1

Introduction to AppAssure

This chapter provides an introduction and overview of Dell AppAssure. It describes the features, functionality, and architecture, and consists of the following topics:

- AppAssure core technologies
- Product features of AppAssure

AppAssure sets a new standard for unified data protection by combining backup, replication, and recovery in a single solution that is engineered to be the fastest and most reliable backup for protecting virtual machines (VM), physical machines, and cloud environments.

AppAssure combines backup and replication into one integrated and unified data protection product that also provides application awareness to ensure reliable application data recovery from your backups. AppAssure is built on the new, patent-pending True Scale architecture which delivers the fastest backup performance with very aggressive, near-zero recovery time objectives (RTO) and recovery point objectives (RPO).

AppAssure combines several unique, innovative, and breakthrough technologies:

- Live Recovery
- Verified Recovery
- Universal Recovery
- True Global Deduplication

These technologies are engineered with secure integration for cloud disaster recovery and deliver fast and reliable recovery. With its scalable object store, AppAssure is uniquely capable of handling up to petabytes of data very rapidly with built-in global deduplication, compression, encryption, and replication to any private or public cloud infrastructure. Server applications and data can be recovered in minutes for data retention and compliance purposes.

Today's legacy backup tools and first-generation VM backup tools are inefficient and ineffective. The outdated backup tools lack the ability to handle large-scale data and do not offer the level of performance and reliability needed for protecting business-critical applications. Combine this with complex and mixed IT environments and it presents an administrative challenge for IT professionals and vulnerability of system data.

AppAssure addresses this complexity and inefficiency through our core technology and support of multi-hypervisor environments including those running on VMware vSphere and Microsoft Hyper-V, which comprise both private and public clouds. AppAssure offers these technological advances while dramatically reducing IT management and storage costs.

AppAssure core technologies

Details about the core technologies of AppAssure are described in the following topics.

Live Recovery

Live Recovery is instant recovery technology for VMs or servers. It gives you near-continuous access to data volumes on virtual or physical servers. You can recover an entire volume with near-zero RTO and an RPO of minutes.

AppAssure backup and replication technology records concurrent snapshots of multiple VMs or servers, providing near instantaneous data and system protection. You can resume the use of the server directly from the backup file without waiting for a full restore to production storage. Users remain productive and IT departments reduce recovery windows to meet today's increasingly stringent RTO and RPO service-level agreements.

Verified Recovery

Verified Recovery lets you perform automated recovery testing and verification of backups. It includes, but is not limited to, file systems; Microsoft Exchange Server 2007, 2010, and 2013; and Microsoft SQL Server 2005, 2008, 2008 R2, 2012 and 2014. Verified Recovery provides recoverability of applications and backups in virtual and physical environments, and features a comprehensive integrity checking algorithm based on 256-bit SHA keys that check the correctness of each disk block in the backup during archiving, replication, and data seeding operations. This ensures that data corruption is identified early and prevents corrupted data blocks from being maintained or transferred during the backup process.

Universal Recovery

Universal Recovery technology gives you unlimited machine restoration flexibility. You can restore your backups from physical to virtual, virtual to virtual, virtual to physical, or physical to physical. You can also carry out bare metal restores to dissimilar hardware; for example, P2V, V2V, V2P, P2P, P2C, V2C, C2P, C2V.

It also accelerates cross-platform moves among virtual machines; for instance, moving from VMware to Hyper-V or Hyper-V to VMware. It builds in application-level, item-level, and object-level recovery: individual files, folders, email, calendar items, databases, and applications. With AppAssure, you can also recover or export physical to cloud, or virtual to cloud.

True Global Deduplication

AppAssure provides true global deduplication that dramatically reduces your physical disk capacity requirements by offering space reduction ratios exceeding 50:1, while still meeting the data storage requirements. True Scale inline block-level compression and deduplication with line speed performance, along with built-in integrity checking, prevents data corruption from affecting the quality of the backup and archiving processes.

Product features of AppAssure

Using AppAssure, you can manage all aspects of protection and recovery of critical data through the following features and functionality. They include:

- Repository
- True Global Deduplication
- Encryption
- Replication
- Recovery-as-a-Service (RaaS)
- Retention and archiving
- Virtualization and cloud
- Alerts and event management
- License Portal
- Web console
- Service management APIs
- White labeling
- AppAssure wizards

Repository

The AppAssure repository uses deduplication volume manager (DVM) to implement a volume manager that provides support for multiple volumes, each of which could reside on different storage technologies such as Storage Area Network (SAN), Direct Attached Storage (DAS), Network Attached Storage (NAS), or cloud storage. Each volume consists of a scalable object store with deduplication. The scalable object store behaves as a records-based file system, where the unit of storage allocation is a fixed-sized data block called a record. This architecture lets you configure block-sized support for compression and deduplication. Rollup operations are reduced to metadata operations from disk intensive operations because the rollup no longer moves data but only moves the records.

The DVM can combine a set of object stores into a volume and they can be expanded by creating additional file systems. The object store files are pre-allocated and can be added on demand as storage requirements change. It is possible to create up to 255 independent repositories on a single AppAssure Core and to further increase the size of a repository by adding new file extents. An extended repository may contain up to 4,096 extents that span across different storage technologies. The maximum size of a repository is 32 Exabytes. Multiple repositories can exist on a single core.

True Global Deduplication

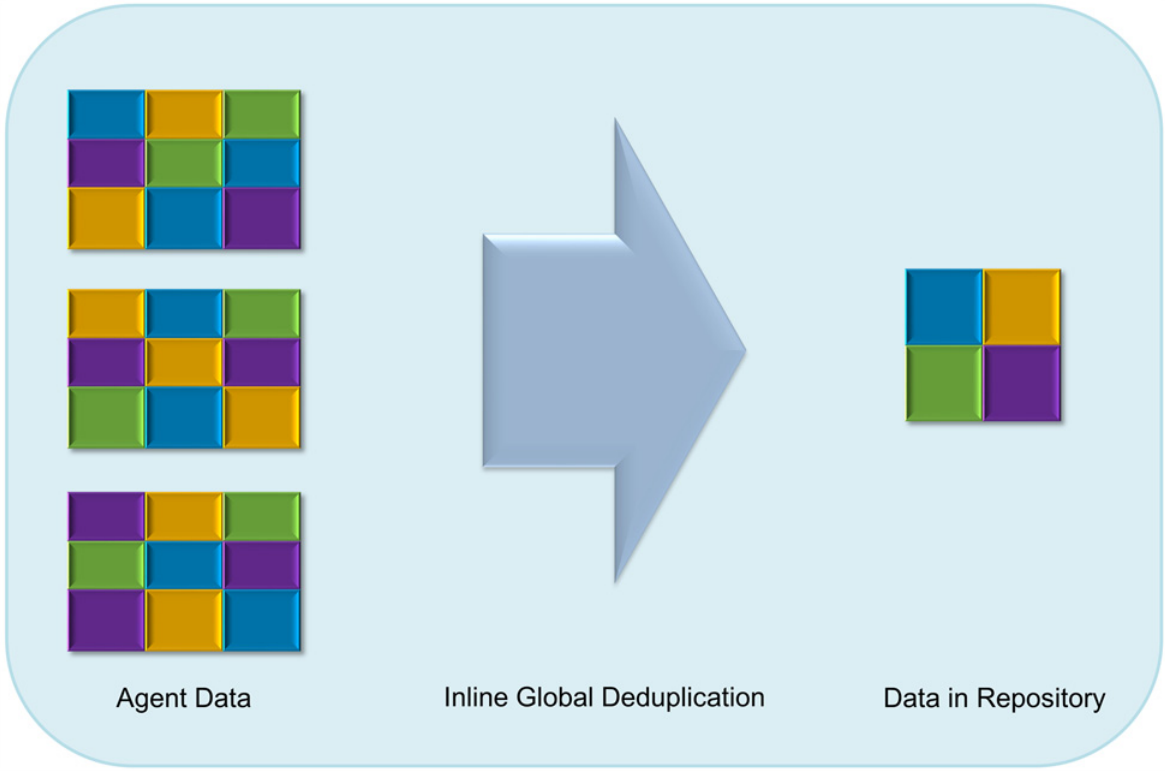
True global deduplication is an effective method of reducing backup storage needs by eliminating redundant or duplicate data. Deduplication is effective because only one unique instance of the data across multiple backups is stored in the repository. The redundant data is stored, but not physically; it is simply replaced with a pointer to the one unique data instance in the repository.

Conventional backup applications have been performing repetitive full backups every week, but AppAssure performs incremental block-level backups of the machines forever. This incremental-forever approach in tandem with data deduplication helps to drastically reduce the total quantity of data committed to the disk.

The typical disk layout of a server consists of the operating system, application, and data. In most environments, the administrators often use a common flavor of the server and desktop operating system across multiple systems for effective deployment and management. When AppAssure backs up at the block level across multiple machines simultaneously, it provides a more detailed view of what is in the backup and what is not,

irrespective of the source. This data includes the operating system, the applications, and the application data across the environment.

Figure 1. True global deduplication



AppAssure performs target-based inline data deduplication. This method transmits the snapshot data to the Core before it is deduplicated. Inline data deduplication simply means that the data is deduplicated before it is committed to disk. This is very different from at-source or post-process deduplication, where the data is deduplicated at the source before it is transmitted to the target for storage, and in post-process the data is sent raw to the target where it is analyzed and deduplicated after the data has been committed to disk. At-source deduplication consumes precious system resources on the machine whereas the post-process data deduplication approach needs all the requisite data on disk (a greater initial capacity overhead) before commencing the deduplication process. On the other hand, inline data deduplication does not require additional disk capacity and CPU cycles on the source or on the Core for the deduplication process. Lastly, conventional backup applications perform repetitive full backups every week, while AppAssure performs incremental block-level backups of the machines forever. This incremental forever approach in tandem with data deduplication helps to drastically reduce the total quantity of data committed to the disk with a reduction ratio of as much as 80:1.

Encryption

AppAssure provides integrated encryption to protect backups and data-at-rest from unauthorized access and use, ensuring data privacy. AppAssure provides strong encryption. By doing so, backups of protected computers are inaccessible. Only the user with the encryption key can access and decrypt the data. There is no limit to the number of encryption keys that can be created and stored on a system. DVM uses AES 256-bit encryption in the Cipher Block Chaining (CBC) mode with 256-bit keys. Encryption is performed inline on snapshot data, at line speeds without impacting performance. This is because DVM implementation is multi-threaded and uses hardware acceleration specific to the processor on which it is deployed.

Encryption is multi-tenant ready. The deduplication has been specifically limited to records that have been encrypted with the same key; two identical records that have been encrypted with different keys will not be deduplicated against each other. This design decision ensures that deduplication cannot be used to leak data between different encryption domains. This is a benefit for managed service providers, as replicated backups for multiple tenants (customers) can be stored on a single core without any tenant being able to see or access

other tenant data. Each active tenant encryption key creates an encryption domain within the repository where only the owner of the keys can see, access, or use the data. In a multi-tenant scenario, data is partitioned and deduplicated within the encryption domains.

In replication scenarios, AppAssure uses SSL 3.0 to secure the connections between the two cores in a replication topology to prevent eavesdropping and tampering.

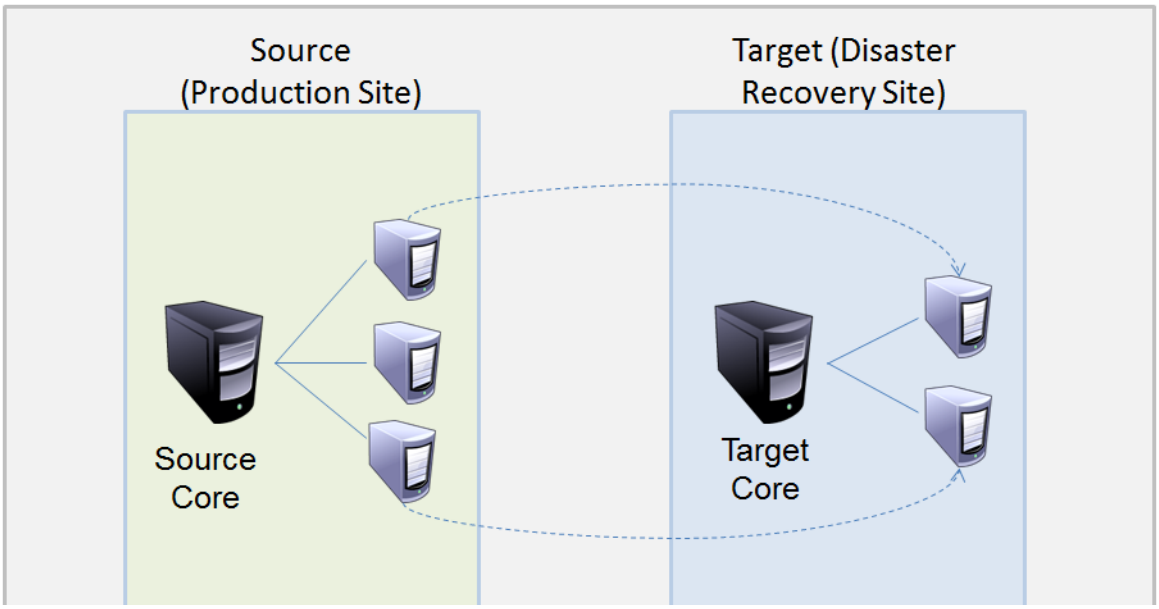
Replication

Replication is the process of copying recovery points from an AppAssure core and transmitting them to another AppAssure core in a separate location for the purpose of disaster recovery. The process requires a paired source-target relationship between two or more cores.

The source core copies the recovery points of selected protected machines, and then asynchronously and continually transmits the incremental snapshot data to the target core at a remote disaster recovery site. You can configure outbound replication to a company-owned data center or remote disaster recovery site (that is, a "self-managed" target core). Or, you can configure outbound replication to a third-party managed service provider (MSP) or cloud provider that hosts off-site backup and disaster recovery services. When replicating to a third-party target core, you can use built-in work flows that let you request connections and receive automatic feedback notifications.

Replication is managed on a per-protected-machine basis. Any machine (or all machines) protected or replicated on a source core can be configured to replicate to a target core.

Figure 2. Replication



Replication is self-optimizing with a unique Read-Match-Write (RMW) algorithm that is tightly coupled with deduplication. With RMW replication, the source and target replication service matches keys before transferring data and then replicates only the compressed, encrypted, deduplicated data across the WAN, resulting in a 10x reduction in bandwidth requirements.

Replication begins with seeding: the initial transfer of deduplicated base images and incremental snapshots of the protected agents, which can add up to hundreds or thousands of gigabytes of data. Initial replication can be seeded to the target core using external media. This is typically useful for large sets of data or sites with slow links. The data in the seeding archive is compressed, encrypted and deduplicated. If the total size of the archive is larger than the space available on the removable media, the archive can span across multiple devices based on the available space on the media. During the seeding process, the incremental recovery points replicate to the target site. After the target core consumes the seeding archive, the newly replicated incremental recovery points automatically synchronize.

Recovery-as-a-Service (RaaS)

Managed service providers (MSPs) can fully leverage AppAssure as a platform for delivering recovery as a service (RaaS). RaaS facilitates complete recovery-in-the-cloud by replicating customers' physical and virtual servers along with their data to the service provider's cloud as virtual machines to support recovery testing or actual recovery operations. Customers wanting to perform recovery-in-the-cloud can configure replication on their protected machines on the local cores to an AppAssure service provider. In the event of a disaster, the MSPs can instantly spin-up virtual machines for the customer.

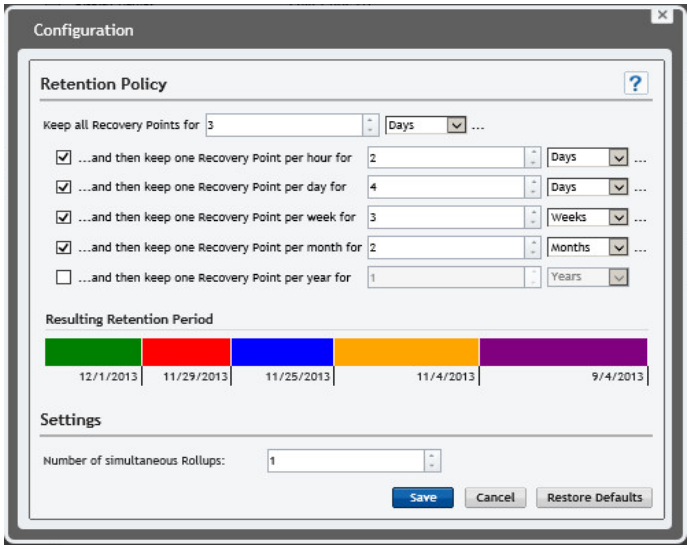
MSPs can deploy multi-tenant AppAssure-based RaaS infrastructure that can host multiple and discrete organizations or business units (the tenants) that ordinarily do not share security or data on a single server or a group of servers. The data of each tenant is isolated and secure from other tenants and the service provider.

Retention and archiving

AppAssure offers flexible backup and retention policies that are easily configurable. The ability to tailor retention polices to the needs of an organization not only helps to meet compliance requirements but does so without compromising recovery time objectives (RTO).

Retention policies enforce the periods of time in which backups are stored on short-term (fast and expensive) media. Sometimes certain business and technical requirements mandate extended retention of these backups, but use of fast storage is cost prohibitive. Therefore, this requirement creates a need for long-term (slow and cheap) storage. Businesses often use long-term storage for archiving both compliance and non-compliance data. The archive feature supports extended retentions for compliance and non-compliance data, as well as being used for seeding replication data to a target core.

Figure 3. Retention policy



In AppAssure, retention policies can be customized to specify the length of time a backup recovery point is maintained. As the age of the recovery points approach the end of their retention period, they age out and are removed from the retention pool. Typically, this process becomes inefficient and eventually fails as the amount of data and the period of retention start growing rapidly. AppAssure solves the big data problem by managing the retention of large amounts of data with complex retention policies and performing rollup operations for aging data using efficient metadata operations.

Backups can be performed with an interval of a few minutes; and, these backups age over days, months, and years. Retention policies manage the aging and deletion of old backups. A simple waterfall method defines the aging process. The levels within the waterfall are defined in minutes, hours, and days; weeks, months, and years. The retention policy is enforced by the nightly rollup process.

For long term archiving, AppAssure lets you create an archive of the source or target core on any removable media. The archive is internally optimized, and all data in the archive is compressed, encrypted, and

deduplicated. If the total size of the archive is larger than the space available on the removable media, the archive will span across multiple devices based on the available space on the media. Recovery from an archive does not require a new core; any core can ingest the archive and recover data if the administrator has the passphrase and the encryption keys.

Virtualization and cloud

The AppAssure Core is cloud-ready, which means you can leverage the compute capacity of the cloud for recovery and archiving.

AppAssure can export any protected or replicated machine to a virtual machine (VM), such as a licensed version of VMware or Hyper-V. You can perform a one-time virtual export, or you can establish a virtual standby VM by establishing a continual virtual export. With continual exports, the virtual machine is incrementally updated after every snapshot. The incremental updates are fast and provide standby clones that are ready to be powered up with a click of a button. Supported VM export types include: VMware Workstation and VMware Server on a folder; direct export to a vSphere or VMware ESX(i) host; export to Oracle VirtualBox; and export to Microsoft Hyper-V Server on Windows Server 2008 (x64), 2008 R2, 2012 (x64), and 2012 R2 (including support for Hyper-V generation 2 VMs).

Additionally, you can archive your repository data to the cloud using Microsoft Azure, Amazon S3, Rackspace Cloud Files, or other OpenStack-based cloud services.

Alerts and event management

In addition to HTTP REST APIs, AppAssure also includes an extensive set of features for event logging and notification using email, syslog, or Windows Event Log. Email notifications can be used to alert users or groups of the health or status of different events in response to an alert. The syslog and Windows Event Log methods are used for centralized logging to a repository in multi-operating system environments; while in Windows-only environments, only the Windows Event Log is used.

License Portal

The Dell AppAssure License Portal provides easy-to-use tools for managing license entitlements. You can download, activate, view, and manage license keys and create a company profile to track your license assets. Additionally, the portal enables service providers and re-sellers to track and manage their customer licenses.

Web console

AppAssure features a web-based central console that manages distributed AppAssure Cores from one central location. Managed service providers (MSPs) and enterprise customers with multiple distributed cores can deploy this console to get a unified view for centralized management. The AppAssure Central Management Console lets you organize the managed cores in hierarchical organizational units. These organizational units can represent business units, locations, or customers for MSPs with role-based access. Using the central console, you can also run reports across all of your managed cores.

Service management APIs

AppAssure comes bundled with a set of service management APIs and provides programmatic access to all of the functionality available through the AppAssure Central Management Console. The service management API is a REST API. All the API operations are performed over SSL and are mutually authenticated using X.509 v3 certificates. The management service can be accessed from within the environment or directly over the Internet from any application that can send and receive an HTTPS request and response. The approach facilitates easy integration with any Web application such as relationship management methodology (RMM) tools or billing systems. Also included with AppAssure is an SDK client for PowerShell scripting.

White labeling

AppAssure can be re-branded and white-labeled for select enterprise and OEM partners under the Platinum service provider program. The Platinum service provider program lets partners brand AppAssure with their custom name, logo, and color themes and deliver the product or service with their own branding and look-and-feel to their customers.

As an AppAssure partner, you can tailor the software to meet your business requirements. To further explore how you can brand AppAssure to suit your business needs, contact Dell AppAssure Sales at sales@software.dell.com for more information.

AppAssure wizards

A wizard is a guided set of steps presented to a user in a pop-up window to automate one or more complex tasks. By definition, a wizard contains two or more pages of information for the user to complete. (A single-window pop-up is simply considered a dialog box). Wizards use a variety of familiar graphic conventions to collect information, such as text boxes, radio buttons, check boxes, and drop-down menus.

When you start a wizard, a window opens above the user interface, and guides you through a series of choices required to accomplish the task. You complete the information presented in each page of the wizard as befits your requirements, navigating back and forth between wizard pages by clicking **Next** and **Back** buttons. When complete, you click a submit button (to confirm your choices and perform the task) or the **Cancel** button (to cancel without making changes, and to return to the UI from which you opened the wizard).

AppAssure offers several wizards, which can be divided into two categories.

First, there are wizards to install, update, or remove AppAssure software and related components. These are typically launched by double-clicking an executable software installer program. Installers can be accessed by downloading specific components from the Dell AppAssure License Portal.

Secondly, wizards are available in the AppAssure Core Console user interface. These are typically launched by clicking on a button or link labeled with the wizard function or result. The wizards in this category are listed below.

Table 1. Wizards available in the AppAssure Core Console

Wizard Name	Launched by	Description
Quick Start Guide	Quick Start Guide option (Help menu)	Unifies multiple workflows to simplify common tasks for AppAssure. Steps user through the process to protect machines, to configure replication for new agents, to export protected data to virtual machines, to encrypt recovery point data, configure email notification groups, and to configure a retention policy.
Protect Machine Wizard	Protect Machine button (button bar)	Sets up protection on a single machine that you specify. Allows you to provide a name for that machine to display in the Core console. If Agent software is already installed, allows you to select volumes for protection. If not, the wizard will install the software and protect all volumes. Sets a default protection schedule or allows you to set a custom schedule. Using Advanced settings, you can choose (or create) a repository and establish encryption for your protected data.

Table 1. Wizards available in the AppAssure Core Console

Wizard Name	Launched by	Description
Protect Multiple Machines Wizard	Bulk Protect button (Home tab)	Sets up protection on multiple machines that you specify, either from a Windows Active Directory domain server, a VMware vCenter Server/ESX(i) virtual host, or manually (by entering a list in a specified format).
Replication Wizard	Add Target Core link (Replication tab)	Sets up replication from a primary (or source) Core so that a copy of your protected data is always available on a separate target Core.
Restore Machine Wizard	Restore button (button bar)	Steps you through the process of restoring data from a recovery point on the Core to a protected machine, or initiating a bare metal restore.
Export Wizard	Export button (button bar), Add link (Virtual Standby tab)	Exports recovery point data from a protected machine to a virtual machine in any supported VM format. You can perform a one-time export or set up virtual standby for continuous export.

While wizards typically set up or configure features in AppAssure, you can later modify most of those aspects from the Core Console.

2

Understanding the AppAssure Core Console

This chapter describes the different elements of the AppAssure Core Console user interface. It includes the following topics:

The Core Console is the main user interface (UI) through which users interact with AppAssure. When you log into the AppAssure Core Console, you see the following elements in the UI.

Table 2. UI elements included in the Core Console

UI Element	Description
Branding area	For typical environments, the AppAssure Core Console is branded Dell AppAssure. Clicking on the branding area results in a new tab opening in the web browser, taking you to product documentation on the Support site.
Button bar	The button bar contains buttons accessible from anywhere in the Core Console. These buttons launch wizards to accomplish common tasks such as protecting a machine, performing a restore from a recovery point, or exporting data to a virtual machine. These buttons include the following:
	The **Protect** button launches the Protect Machine Wizard, from which you can protect a single machine in the AppAssure Core. Additionally for other protection options, you can access the drop-down menu next to this button. • The **Protect Machine** option is another method to launch the Protect Machine Wizard to protect a single machine. • The **Protect Cluster** option allows you to connect to a server cluster. • The **Bulk Protect** option opens the Protect Multiple Machines Wizard to allow you to protect two or more machines simultaneously.
	The **Restore** button opens the Restore Machine Wizard to allow you to restore data from recovery points saved from a protected machine.
	The **Export** button opens the Export Wizard. From this wizard you can create a virtual machine from recovery points saved in the AppAssure Core. You have the option of creating a one-time export, or you can define parameters for a VM that is continually updated after every snapshot for a protected machine.

Table 2. UI elements included in the Core Console

UI Element	Description
Running tasks count	Shows how many jobs are currently running. This value is dynamic based on the system state. When you click the drop-down menu, you see a status summary for all jobs currently running. By clicking the X for any job, you can choose to cancel that job.
Help drop-down menu	The Help menu includes the following options: • **Help**. Links to AppAssure technical documentation on the Dell Support website. • **Support.** Links to the Dell Support website, providing access to Live Chat, video tutorials, AppAssure knowledge base articles, frequently asked questions, and more. • **Quick Start Guide.** The Quick Start Guide is a guided flow of suggested tasks to configure and use AppAssure. The guide opens automatically each time you log in to the Core Console, unless you disable this feature. You can also open the Quick Start Guide from the Help menu. For more information about the Quick Start Guide, see Navigating the Core Console. • **About**. Opens the About AppAssure Core dialog box, including version information and a description of the software.
Server date and time	The current time and date of the machine running the AppAssure Core service appears at the top right of the Core Console. This is the time recorded by the system for events such as logging, scheduling, and reporting. For example, when applying protection schedules, the time displayed on the Core Console is used. This is true even if the time zone is different on the database server or on the client machine where the browser is running.
Left navigation area	The left navigation area appears under the branding area, on the left side of the user interface. It contains the icon bar and the Protected Machines menu. If you have added replication to this Core, then this area contains a Replicated Machines menu. If you have any machines that were removed from protection but for which recovery points were saved, then this area contains a Recovery Points Only menu. If you added any custom groups, then this area contains a Custom Group menu. You can toggle the appearance of the left navigation area on and off. This is helpful when you need to see more content in the main navigation area of the UI. To hide this section, click the gray border between the left navigation and main navigation areas. To show this UI element once more, click the gray border again.
Icon bar	The icon bar includes a graphic representation for each tab on the Core Console. Clicking the appropriate item takes you to the corresponding section of the user interface where you can manage that function. Icons in the icon bar include:
Icon bar	 **Home.** Click the Home icon to navigate to the Core Home tab.
	 **Replication**. Click the Replication icon to view or manage incoming or outgoing replication.
	 **Virtual Standby.** Click the Virtual Standby icon to export information from a recovery point to a bootable virtual machine.

Table 2. UI elements included in the Core Console

UI Element	Description
	Events. Click the Events icon to view a log of all system events related to the AppAssure Core.
	Tools. Click the Tools icon to view or manage system information, boot CDs, mounts, bulk deploy, downloads, archives, cloud accounts, diagnostics, or reports.
	Configuration (Settings). Click the Configuration icon to view or manage configuration options for the AppAssure Core, including repositories; encryption keys; notification groups or email server settings for notifications; the Core retention policy; SQL Server attachability; core settings; AppAssure license details, constraints, license pool information, and license server information; backup and restore functionality for core settings; and core job settings.
Protected Machines menu	The Protected Machines menu appears in the left navigation area of the UI. The icon represents a single machine. In this menu, you can view any protected machines, protected clusters, or replicated machines configured in your Core. If you have any protected groups or recovery point-only machines, these also appear in this menu. You can collapse or expand the view for any of these objects in your Core by clicking the arrow on the left side of these categories. From the Protected Machines menu, you can perform all actions that were previously accessible only from the Machines tab. If you click the Protected Machines menu, the Machines tab appears, showing all protected machines on this Core in the Protected Machines pane. For more information, see Viewing the protected machines menu.
Replicated Machines menu	If replicating machines from another AppAssure Core, the name of that Core appears as a separate menu under the Protected Machines menu. The icon portrays replication from one machine to many machines. This menu lists each replicated machine. You can collapse or expand the view for any of these objects in your Core by clicking the arrow on the left side of these categories. From the Replicated Machines menu, you can perform actions on all replicated machines. If you click the Replicated Machines menu, the Machines tab appears, showing all replicated machines on this Core in the Replicated Machines pane. For more information, see Viewing the replicated machines menu.
Recovery Points Only menu	If any machines previously protected on the Core were removed from protection, but the recovery points were not deleted, then the Recovery Points Only menu appears. The icon represents a single machine. You can collapse or expand the view for any of these objects in your Core by clicking the arrow on the left side of these categories. From the Recovery Points Only menu, you can remove the recovery points for all the recovery-points only machines on this Core. If you click the Recovery Points Only menu, the Machines tab appears, showing the machines from which the recovery points were saved. For more information, see Viewing the Recovery Points Only menu.

Table 2. UI elements included in the Core Console

UI Element	Description
Custom Groups menu	 If your Core includes any custom groups, then the left navigation area includes a Custom Group menu. This icon represents a label, which is the property you can use to name a custom group. You can collapse or expand the view for any of these objects in your Core by clicking the arrow on the left side of these categories. From the Custom Groups menu, you can perform actions for like items in the group. If you click the Custom Groups menu, the Machines tab appears, showing a pane for each of the AppAssure objects that appear in your group: protected machines, replicated machines, and recovery points-only machines. For more information, see Viewing the Custom Groups menu.
Context-sensitive help	From the AppAssure Core Console, each time you click the **Help** icon (a blue question mark), a resizable browser window opens with two frames. The left frame contains a navigation tree showing topics from the *Dell AppAssure User Guide*. The right frame displays content for the selected help topic. At any given time, the help navigation tree expands to show the location in its hierarchy for the selected topic. You can browse through all User Guide topics using this context-sensitive help feature. Close the browser when you are done browsing topics. You can also open help from the Help option of the Help menu.

Complete the following steps to access the AppAssure Core Console.

To access the AppAssure Core Console

- Perform one of the following to access the AppAssure Core Console:
  - a Log in locally to your AppAssure Core server, and then double click the **Core Console** icon.
  - b Or, type one of the following URLs in your Web browser:
    - https://<yourCoreServerName>:8006/apprecovery/admin/ or
    - https://<yourCoreServerIPaddress>:8006/apprecovery/admin/

> **NOTE:** Because the AppAssure Core Console UI is dependent on JavaScript, the Web browser you use to access the Core Console must have JavaScript enabled.
>
> If you have changed the default port for the AppAssure service, update the port in the URL above accordingly.

Navigating the Core Console

When you log into the Core Console, and any time you click the Home icon or tab, the Home tab appears and is automatically selected. You can navigate to any other tab by clicking the tab name or the corresponding icon.

The other tabs accessible from the Core include Replication, Virtual Standby, Events, Tools, and Configuration.

When you select a protected machine or replicated machine, the focus of the Core Console changes to display information about the selected machine only, rather than the Core.

Tabs accessible from a protected or replicated agent include Summary (replacing the Home tab), Recovery Points, Events, Tools, and Configuration.

To return to viewing information about the Core, including views of multiple protected or replicated machines, click on the Home icon on the top left of the UI.

NOTE: Any time you see the Home tab, you are viewing information about the Core. If you see a Summary tab, you are viewing information about a single machine protected by or replicated in the Core. If you see a Machines tab, you are viewing information about all of the machines protected by or replicated in the AppAssure Core. If you see a Recovery Points tab, you are viewing information about a recovery points only machine in the Core.

For information about the features and functions available from each tab, see the appropriate section below.

For more information on viewing protected machines, see Viewing the protected machines menu. For more information on managing protected machines, see Managing protected machines.

For more information on viewing replicated machines, see Monitoring replication.

Viewing the Home tab

The Home tab is only applicable to the Core. It displays all of the machines the Core protects, as well as all associated repositories and alerts for those machines. You can reach the Home tab by clicking the tab or on the house icon above the machines column on the left of the page. The following table describes the various elements on the Home tab.

Table 3. Home tab options

UI Element	Description
Protected Machines	The Protected Machines menu lists the machines that this Core protects. • If you click the Protected Machines menu, the Machines tab appears, showing the Protected Machines pane. For more information on what you can accomplish on the Machines tab, see Viewing the Machines tab for a protected machine. • If you click on a specific machine under the Protected Machines menu, the Summary tab appears, showing summary information for that machine. For more information on what you can accomplish on the Summary tab, see Viewing the Summary tab.
Replicated Machines	If you see the name of another AppAssure Core under the Protected Machines menu, this is the Replicated Machines menu. This menu lists the machines this Core protects through replication, which means this Core is the target core for the listed protected machines. The section includes the following information for each replicated machine: • If you click the Replicated Machines menu, the Machines tab appears, showing the Replicated Machines pane. For more information on what you can accomplish on the Machines tab, see Viewing the Machines tab for a protected machine. • If you click on a specific replicated machine under the Replicated Machines menu, the Summary tab appears, showing summary information for that machine. For more information on what you can accomplish on the Summary tab, see Viewing the Summary tab. For more information about replication, see Understanding replication.

Table 3. Home tab options

UI Element	Description
Recovery Points Only	Recovery points only machines are machines that you no longer protect with an AppAssure Core, but for which you continue to store recovery points. If you remove a machine from protection but choose to keep the recovery points, that machine is then listed under Recovery Points Only. These can be used for file-level recovery, but cannot be used for bare metal restore, for restoring entire volumes, or for adding snapshot data. This section includes the following information for each recovery points only machine: • **Status.** There is no status indicator for recovery points-only machines, because there is no connection with the machine. • **Machine Name.** The display name of the machine for which you kept recovery points. • **Repository.** The name of the repository storing the recovery points for that machine. • **Last Snapshot.** The time at which AppAssure took the most recent recovery point for that machine. • **Recovery Points.** The number of recovery points stored in the repository for that machine. • **Total Protected Space.** The amount of space the recovery points for this machine use in the repository.
Repositories	This section lists all of the repositories this Core uses for storing recovery points. It includes the following information for each repository: • **Status.** Colored circles in the Status column show whether the repository is mounted and can accept recovery point transfers. • **Repository Name.** The display name of the repository. • **Total Size.** The total amount of space in the repository. • **Free Space.** The amount of space available in the repository. • **Protected Data.** The amount of used space in the repository. • **Machines.** The number of machines for which the repository stores recovery points. • **Recovery Points.** The number of recovery points stored in the repository. • **Compression Ratio.** The rate at which the repository compresses the protected data to save space. For more information, see Understanding repositories.
Alerts	This section lists the alerts for the Core and every machine it protects. The section includes the following information: • **Icons.** The column of icons indicates the nature of the alert. • **Date.** Displays the date and time of when AppAssure issued the alert. • **Message.** Describes the alert. You can also see these details on the Core Events tab. For more information, see Viewing tasks, alerts, and events.

Viewing the Replication tab

The Replication tab lets you view and change details for cores that are configured to transfer replicated data with your current Core. You can also view and change details for the machines being replicated. The layout of the tab is the same whether your core serves as a source core or a target core, so there are sections designated for both situations.

You can reach the Replication tab by clicking the tab or the Replication icon that resembles a split computer monitor above the machines column on the left of the page. The following table provides descriptions of the sections and capabilities available on the Replication tab.

Table 4. Replication tab options

UI Element	Description
Outgoing Replication	The section displays the core or cores that are replicating data to a remote target core. Displayed information includes the following: • **Target Core Name.** The name of the Core to which your current source is replicating data. • **State.** The state of the replication pairing; for example, Established indicates a successful pairing between the source and target cores so that transfers can take place. • **Machines.** This column lists the number of protected machines or agents being replicated to the target core. • **Progress.** When a replication job occurs, this column displays a bar with which you can monitor the progress of the job.
Incoming Replication	The section displays the core or cores from which the current core is receiving replicated data (for example, if you are in the Core Console of a target core). Displayed information includes the following: • **Target Core Name.** The name of the Core from which your current source receiving replicated data. • **State.** The state of the replication pairing; for example, Established indicates a successful pairing between the source and target cores so that transfers can take place. • **Machines.** This column lists the number of protected machines or agents your current core is replicating. • **Progress.** When a replication job occurs, this column displays a bar with which you can monitor the progress of the job.
Add Target Core	Use this button to add a target core to which you can replicate. For more information, see Configuring replication. **NOTE:** You cannot add a source core from a target core.
Settings	Use this button to change the settings for all incoming or outgoing replication. For more information, see Managing replication settings.
> (Core)	The expand button next to the name of a core displays the protected machines or agents being replicated. It includes the following details: • **Agent Name.** The name of the protected machine. • **State.** The state of the pairing between this machine and the target core; for example, the state of Established indicates a successful pairing with which transfer can occur. • **Status.** This column is present only for agents that appear under target cores and displays whether the machine is “Not yet replicated” or “In Sync as of [date time].” • **Progress.** When a replication job occurs, this column displays a bar with which you can monitor the progress of the job.
> (Agent)	This bracket appears only next to agents listed under a target core. It expands a list of the recovery points that were replicated from that machine. For more information, see Step 2 in Viewing recovery points.

Table 4. Replication tab options

UI Element	Description
Core drop-down menu (under Outgoing Replication)	The drop-down menu (gear icon) next to the name of a target core includes the following functions: • **Details.** Shows the customer ID, URL, display name, state, Core ID, email address, and comments associated with this core. For more information, see Monitoring replication. • **Change Settings.** Lets you edit the host and port for this core. For more information, see Changing target core settings. • **Delete.** Removes the Core from replication. For more information, see Removing a target core from replication. • **Schedule.** Lets you designate specific times when replication should occur. For more information, see Scheduling replication. • **Add Machines.** Lets you add protected machines you want to replicate to this target core.
Core drop-down menu (under Incoming Replication)	The drop-down menu (gear icon) next to the name of a source core includes the following functions: • **Details.** Shows the customer ID, URL, display name, state, Core ID, email address, and comments associated with this core. For more information, see Monitoring replication. • **Consume.** Adds a seed drive of a protected machine to the target core repository to establish the foundation of a replicated agent. For more information, see Consuming the seed drive on a target core. • **Delete.** Removes the Core from replication. For more information, see Removing a source core from replication.
Agent drop-down menu (table heading row)	The drop-down menu (gear icon) in the heading row applies to any protected machines or agents selected from the table. It includes the following functions: • **Force.** Forces a transfer of replicated data. • **Pause.** Pauses replication and prevents the job from occurring until you resume. • **Resume.** Takes replication out of a paused state • **Copy.** Creates a seed drive of the protected machine. For more information, see Understanding seed drives and Using the Copy function to create a seed drive. • **Delete.** Removes the selected agents from replication. For more information about these functions, see Managing replication settings and Removing replication.
Force Refresh	There is a Force Refresh button following the Outgoing Replication and the Incoming Replication sections. For each section, this circular button establishes on-demand communication between the cores to update any data that may have changed since the last scheduled transfer.

Table 4. Replication tab options

UI Element	Description
Agent drop-down menu (individual agents under Outgoing Replication)	The drop-down menu (gear icon) next to an individual protected machine or agent offers functions that apply only to that machine. It includes the following functions: • **Settings.** Lets you set the replication priority for this agent. For more information, see Setting replication priority for a protected machine. • **Force.** Forces a transfer of replicated data. Forcing replication. • **Delete.** Removes the agent from replication. For more information, see Removing a protected machine from replication on the source core.
Agent drop-down menu (individual agents under Incoming Replication)	The drop-down menu (gear icon) next to an individual protected machine or agent offers functions that apply only to that machine. It includes the following functions: • **Settings.** Lets you set the replication priority for this agent. For more information, see Setting replication priority for a protected machine. • **Failover.** If there is a disaster recovery situation, this function lets you fail over to the replicated machine until you repair the original machine. For more information, see Understanding failover and failback. • **Delete.** Removes the agent from replication. For more information, see Removing a protected machine on the target core.

Viewing the Virtual Standby tab

The Virtual Standby tab lets you initiate and manage the continual export of a protected machine to one of several virtual machine (VM) host types. Available VM types include:

- ESXi
- VMware Workstation
- Hyper-V
- VirtualBox

The menu bar under the Virtual Standby heading provides actions that you can apply to any Virtual Standby machine you select. The following table includes descriptions of the various elements and capabilities on the Virtual Standby tab.

Table 5. Virtual Standby tab options

UI Element	Description
Add	Creates a Virtual Standby machine by exporting a recovery point to a virtual destination.
Force	Performs an on-demand transfer of the latest recovery points from the original protected machine to the Virtual Standby machine.
Pause	Pauses the transfer of data from the Core to the virtual machine.
Resume	Resumes data transfer after it was paused.
Remove	Terminates the connection between the Core and the virtual machine.
Show Export Queue (#)	Reveals the order in which data is scheduled to transfer for multiple virtual machines.

In the Core Console, the table under the menu bar lists the virtual machines to which the Core is connected. The following table defines the details that the tab displays for each machine.

Table 6. Virtual Standby tab table

UI Element	Description
Status	The state in which data transfer is in, such as paused.
Machine Name	The name of the Virtual Standby machine.
Destination	The name of the machine to which the Core exports the recovery points.
Export Type	The type of virtual machine host software.
Last Export	The date and time of the last export.

Viewing the Events tab

On the Events tab, you can view the jobs that occurred or are in progress on the Core. Buttons at the top of the page let you navigate to lists of jobs in each of the three categories of activities:

- **Tasks.** A job that the AppAssure must perform to operate successfully.
- **Alerts.** A notification related to a task or event that includes errors and warning.
- **Events.** A composite of all Core tasks and alerts.

The following table includes descriptions of each element on the Events tab.

Table 7. Events tab options

UI Element	Description
Search keyword	Lets you search for a specific item within each category.
From	To narrow your results, you can enter a date at which to begin searching.
To	To narrow your results, you can enter a date at which to stop searching.
Status icons	Each icon represents a different job status. Clicking one of the icons lets you filter the list by that status, thereby generating a report. Clicking the icon a second time removes the filter for that status. You can filter by more than one status. Statuses include: • **Active.** A job that is in progress. • **Queued.** A job that is waiting for another job to complete before it can initiate. • **Waiting.** A job waiting for your approval or completion, such as a seed drive. (For more information about seed drives, see Understanding replication.) • **Complete.** A job that the Core completed successfully. • **Failed.** A job that failed and did not complete.
Service icon	This button adds services jobs to the list of jobs. When you click this icon, a smaller service icon appears on each status icon, which lets you filter by service jobs that have those statuses. Examples of services jobs include deleting index files or removing a machine from protection.
Export type drop-down list	The drop-down list includes the formats to which you can export the event report. It includes the following formats: • PDF • XLS • XLSX • RTF • CSV

Table 7. Events tab options

UI Element	Description
Export icon	Converts the event report to the format you selected.
Page selection	Event reports can include several jobs across multiple pages. The numbers and arrows at the bottom of the Events tab let you navigate the additional pages of the report.

The Events tab displays all events in a table. The table includes the following information for each item.

Table 8. Events tab table information

UI Element	Description
>	The bracket expands the job to reveal the following details: • **Start Time.** When the job began. • **End Time.** When the job ended. • **Elapsed Time.** The amount of time for which the job was active. • **Rate.** The rate at which the Core transfers the data. • **Progress.** When a job is in progress, it shows how much the Core completed in real time. When the job is complete, it shows how much of the job the Core completed. • **Phase.** When a job is in progress, this detail shows in which stage the progress is (for example, Transferring). • **Cancel.** When a job is in progress, a link here lets you cancel the job. • **Total Work.** When a job is complete, it shows how much of the job the Core completed. • **Child Task.** Lists the sub tasks associated with this job.
Job	The type of job the Core performed, such as transfer of volumes, maintaining repository, or rolling up.
Status	The status of the job, such as Succeeded for a successfully completed job or Error for a job that failed due to errors.
Start Time	The day and time at which the job began.
End Time	The day and time at which the job ended.
Details	Opens the Monitor Active Task window. It includes the job details you see when you expand the job, as well as the following details for the child jobs: • Start Time • End Time • Elapsed Time • Rate • Progress • Total Work

Viewing the Tools tab

The Tools tab shows system information by default, and lets you access tools to help manage the core. To access these tools, select an option from the Tools menu on the left, or from the drop-down menu on the Tools tab. These tools are described in the following table.

Table 9. Information about tools accessible to the Core

UI Element	Description
System Info	Shows information about the AppAssure Core, including system information, local and mounted volumes, and Replay engine connections. For more information, see Modifying transfer queue settings.
Boot CDs	From the Boot CDs pane, you can manage (create or delete) boot CD ISO images required to perform a bare metal restore (BMR). The information shown for any existing boot CD images includes the path of the ISO image, the recovery environment for which it was created, and the date the image was created. For information on managing boot CD images, see Managing a Windows boot image or Managing a Linux boot image, respectively.
Mounts	From the Local Mounts pane, you can view or dismount volumes mounted locally. From the Remote Mounts pane, you can view or dismount volumes mounted using the Local Mount Utility. For information on dismounting volumes, see Dismounting recovery points. For information on mounting a recovery point locally, see Mounting a recovery point or Mounting a recovery point volume on a Linux machine, respectively.
Bulk Deploy	From the Deploy Agent to Machines pane, you can deploy the Agent software from the Core to multiple Windows machines simultaneously. For more information on performing this, see Understanding bulk deploy.
Downloads	The Downloads page includes three sections: Agent, Local Mount Utility, and Other files. Each section includes a link you can use to download and install the relevant files. For example: • The Agent section includes a link to download the web installer for the AppAssure Agent software (and also includes a link to the License Portal). • The Local Mount Utility includes a link to download the web installer for the LMU. • The Other files section includes a link to download the SNMP MIB file.
Archive	The Archive page is where you can manage scheduled and mounted archives. The Archive menu offers the following options: • **Scheduled.** This option takes you to the main Scheduled Archives page. • **Create.** Lets you create an archive of recovery points. • **Check Archive.** Opens the Check Archive window and lets you scan an archive file for index files mapping offsets, structure integrity, and checksum integrity. • **Import.** Use this option to import an archive of recovery points so that you can browse the file and restore archived data. For more information, see Understanding archives.
Clouds	The Clouds page lets you add your existing cloud provider accounts to the AppAssure Core Console and so that you can use them as archive destinations. For more information, see Archiving to a cloud. For managing cloud accounts, see Managing cloud accounts.

Table 9. Information about tools accessible to the Core

UI Element	Description
Diagnostics	The Diagnostics page is there for your convenience in case you should ever encounter an issue for which you want to contact AppAssure Support. Here, you can use the Download and Upload menu options to download the Core log or upload a Core log. For more information, see Downloading the Core logs and Uploading logs.
Reports	The Reports page lets you generate reports about the Core's activity and export them in a variety of formats. Report options include the following: • **Compliance.** Provides the status of jobs performed by the Core. • **Summary.** Provides information about the repositories associated with the Core. • **Failure.** A subset of the Compliance report, it lists only the failed jobs attempted by the Core. For more information, see Generating and viewing reports.

Viewing the Configuration tab

The Configuration tab shows settings and configuration options that let you adjust and customize AppAssure protection to suit your business needs. To access these options, select an option from the menu on the Configuration page, or from the drop-down menu on the Configuration tab. These options are described in the following table.

Table 10. Configuration tab options

UI Element	Description
Repositories	The Repositories tab lets you add, edit, and remove repositories and storage locations. For more information, see Understanding repositories.
Security	The Security tab is where you can add encryption keys for safeguarding your data. For more information, see Understanding encryption keys.
Events	Unlike the Events tab, the Configuration tab Events menu option lets you manage and modify the notifications you and your colleagues receive about a wide range of topics, such as replication, archive, and log truncation. On this tab, you can add and edit groups, manage email settings, and filter the amount of email alerts sent by enabling repetition reduction. For more information about the Events option, see Understanding email notifications, Configuring notification groups, and Configuring event retention.
Retention Policy	The Retention Policy tab lets you determine how long a recovery point should be maintained before you archive it. This option uses the rollup feature to combine recovery points after a preset amount of time, deduplicate the data, and reduce the size of the data in your repository, thereby conserving space. For more information, see Managing retention policies.
Attachability	The Attachability tab is where you can modify the attachability check settings for your protected SQL Server machines. An attachability check determines whether the SQL database is safe for recovery. For more information, see Configuring SQL attachability settings.

Table 10. Configuration tab options

UI Element	Description
Settings	The options available on the Settings page of the Configuration tab apply to the Core and all of its protected machines. It includes the ability to modify settings in the following categories: • **General.** The General settings include the Core display name, Web server port, and service port. For more information, see Understanding system information. • **Updates.** The Updates section lets you determine when to check for new AppAssure updates and when to install those updates. It also displays the status of available updates and the last time AppAssure checked for new updates Configuring update settings. • **Nightly Jobs.** Nightly jobs are regular maintenance tasks scheduled during off-hours so as not to disrupt your most intensive business hours. They include tasks such as downloading logs from the protected machines, the attachability check job, rollup, and log truncation. For more information, see Understanding nightly jobs. • **Transfer Queue.** The Transfer Queue settings determine how many data transfers should occur at one time, and how many times AppAssure should attempt a transfer in the case of network issues. For more information, see Modifying transfer queue settings.
Licensing	On the Licensing page, you can change the type of license you use as well as view information about licensing constraints, license pools, and license server. The License Server section indicates communication status between the Core and the License Portal. For more information, see Managing licenses.
Backup/Restore	The Backup/Restore option lets you back up and restore your core settings. For more information, see Backing up and restoring Core settings.
Job Settings	The Job Settings page is where you can add jobs and modify the settings for jobs scheduled to occur on the Core. Jobs include the ChecksumCheckJob, MountabilityCheckJob, and ReplicationJob, among others. For more information, see Configuring Core job settings.

Using the Error dialog box

When an error occurs in the AppAssure Core Console user interface, such as trying to enter an invalid parameter, an Error dialog box appears.

The dialog box typically indicates the cause of the error, includes some links to provide more information about the error, and includes a Close button. You must close the Error dialog box before you continue, but you may want to view more information about the error.

In the Error dialog box, choose from the following options:

Table 11. Error dialog box options

UI Element	Description
Show details	If you select this link, the dialog box expands to include additional details about the error. This option only appears if details are not already displayed.
Hide details	If you select this link, the dialog box contracts and hides error details. This option only appears if error details are already displayed.
Show stack trace	If you select this link, the highest level of detail about the error displays, including the exception chain and the full stack trace error
Hide stack trace	If you select this link, the dialog box contracts and hides error stack trace and details. This option only appears if stack trace details are already displayed.

Table 11. Error dialog box options

UI Element	Description
Search Knowledge Base	If you select this link, a browser window opens showing the Dell AppAssure Support knowledge base (KB). Details about the error are automatically passed as parameters to search the knowledge base by key word. If any KB articles exist pertaining to the error, those articles appear on the page.
Close	If you click this button, the Error dialog box closes, letting you correct the erroneous information entered into the AppAssure Core Console when the error appeared.

User interface errors that cause the Error dialog box to appear are not tracked in AppAssure events tab, since they are simply validation or data entry errors. However, when you click the Search Knowledge Base option for any error, then the URL link provided for that error is recorded to the CoreAppRecovery.log file. You can search the log for the text string “KB article url generated” to see the URL for each error that was viewed in a browser. For more information on downloading or viewing Core error logs, see the topics Downloading the Core logs or Viewing the Core logs, respectively.

Understanding the Quick Start Guide

The Quick Start Guide is a new feature available with AppAssure release 5.4 and later. This feature provides you with a guided flow of suggested tasks to configure and use AppAssure.

The Quick Start Guide appears automatically the first time you upgrade to or install the AppAssure Core and navigate to the Core Console. Click **Start Guide** on the Welcome page of the guide to see the various suggested configuration tasks. Navigate through the guide using the **Skip Step** and **Back** options. When you have seen the last suggested task, click **Finish** to close the guide.

You can launch the Quick Start Guide again at any time. You can also choose to hide the Welcome page in the Quick Start Guide. For more information about these options, see Launching the Quick Start Guide.

Unless you hide it, the Quick Start Guide will reappear each time you log in to the AppAssure Core Console and access the Home tab. For more information, see Hiding the Quick Start Guide.

You are not required to perform the steps suggested by the guide. You can simply view the suggested tasks, navigating through them using the **Skip Step** and **Back** options. Optionally, to hide the guide at any point, click **Exit Guide**.

If you choose to perform any configuration tasks suggested by the Quick Start Guide, follow the prompts indicated in any step of the guide, and the appropriate wizard or relevant area in the user interface appears. Procedures to complete each task suggested by the guide is described in this document, as indicated in the table below.

NOTE: Not all configuration tasks suggested by the Quick Start Guide are required for all users. You must understand which tasks you want to accomplish for your specific needs.

The Quick Start Guide addresses the following configuration tasks:

Table 12. Quick Start Guide configuration tasks

Function	Short Description	Result of Selecting Task, Link to Procedure
Protection	Protecting an agent machine, protecting a server cluster, or protecting multiple machines using bulk protect	Click Protect or select **Protect Machine** from the drop-down menu to open the Protect Machine Wizard. For information on completing the Protect Machine Wizard, see Protecting a machine. Select **Protect Cluster** from the drop-down menu to open the Connect to Cluster dialog box. For more information on protecting a cluster, see Protecting a cluster. Select **Bulk Protect** from the drop-down menu to open the Protect Multiple Machines Wizard. For information on completing the Protect Multiple Machines Wizard, see Protecting multiple machines.
Replication	Setting up replication from a primary (source) core to a secondary (target) core	Click **Replication** to open the Replication tab. Prompts you to add a target core using the Replication Wizard. For information on using the Replication Wizard to set up replication on a self-managed core, see Replicating to a self-managed target core. For general information on replication, see Configuring replication.
Virtual Export	Performing a one-time or establishing continual export from a protected agent machine to a virtual machine	Click **Export** to perform an export of data from your protected machine to a virtual machine. You can either perform a one-time export, or set up virtual standby for continual export to a VM. For information on virtual exports, see Exporting data to a Windows-based virtual machine.
Configuration	Allows you to set up additional configuration for the AppAssure Core	Click **Configuration** to see prompts for security (setting up encryption keys), notifications (setting up event notification), and retention policies (establishing criteria for rollup of aging recovery points).
Configuration: Encryption	Setting up encryption key that you can use for one or more agents	Click **Security** to open the Security page on the Configuration tab. Prompts you to add an encryption key or import one. Once you do this, you may apply it to one or more agents. Encryption is described in the topic Understanding encryption keys.
Configuration: Notifications	Setting up notifications for events, warnings and alerts	Click **Events** to specify notification groups for events, warnings, and alerts. To send these by email, you must also establish SMTP server settings. For more information on managing events, see the topic Managing events, including the topics Configuring notification groups and Configuring an email server.
Configuration: Retention	Viewing or changing the default retention policy for the Core	Click **Retention Policy** to open the Retention Policy page on the Configuration tab. From here you can define how long to keep a recovery point before rolling it up. For conceptual information about retention policies, see the topic Retention and archiving. For procedural information, see Managing retention policies.
Restore	Restoring data from a recovery point on the Core	Click **Restore** to open the Restore Machine Wizard. For information on restoring data, see the topic Restoring volumes from a recovery point.

Launching the Quick Start Guide

The Quick Start Guide appears automatically the first time you upgrade to or install the AppAssure Core. Unless you hide it, the guide reappears each time you access the Home tab on the Core Console.

Use the procedure below to open the Quick Start Guide at any time from the Core Console.

To launch the Quick Start Guide

1 Navigate to the AppAssure Core Console.

2 From the Help menu, select **Quick Start Guide**.

The Quick Start Guide appears.

Hiding the Quick Start Guide

The Quick Start Guide appears automatically the first time you upgrade to or install the AppAssure Core.

It also appears when you select Quick Start Guide from the Help drop-down menu, and each time you access the Home tab on the Core Console.

Use the procedure below to hide the Quick Start Guide.

To hide the Quick Start Guide from the Welcome page

- From the AppAssure Core Console, if you are viewing the Welcome page of the Quick Start Guide, do the following:
  - If you want to hide the Welcome page of the Quick Start Guide, select **Don't show again**.

    NOTE: This option will hide the Welcome page the next time the Start Guide is opened, and every time, until you upgrade the AppAssure Core.

    If you choose to hide this page, and want to access advanced options in the future, then select **Back** in the wizard to see this hidden page.

  - If you want to hide the Quick Start Guide for this session, then click **Close**.

    The Quick Start Guide closes. The next time you access the Home tab on the Core Console, the Quick Start Guide will reappear.

    You can also open the Quick Start Guide from the Help menu, as described in Launching the Quick Start Guide.

To hide the Quick Start Guide from within any step of the Quick Start Guide

- From any page in the Quick Start Guide click **Exit Guide.**

  The Quick Start Guide closes. If you select this option, you can still open the Quick Start Guide from the Help menu, as described in Launching the Quick Start Guide.

Viewing the protected machines menu

In the AppAssure user interface, a Protected Machines menu appears in the left navigation area. By default, this menu is fully expanded, and shows a list of any machines that are protected by this Core. If you have any server clusters protected, then these are included in this list.

You can collapse or expand the view for protected machines and server clusters in your Core by clicking the arrow on the left side of this menu.

The Protected Machines menu includes a drop-down menu on the right side which lists functions that can be performed on all protected machines. Click the arrow to the right of Protected Machine to see the menu and to perform any of the following:

- Force an incremental snapshot for all machines
- Force a base image for all machines
- Pause protection for all machines (if protection is active)

- Resume protection for all machines (if protection is paused)
- Refresh metadata for all protected machines

Each machine listed under the Protected Machines menu also has a drop-down menu that controls functions only for that machine. From the drop-down menu for any machine, you can perform the following:

- Force a snapshot for the selected machine (you can choose volumes on the machine and you can choose either an incremental snapshot or a base image)
- Pause protection for the selected machine (if protection is active)
- Resume protection (if protection is paused)
- Refresh metadata
- Navigate to the Summary tab for the selected machine
- Navigate to the Recovery Points tab for the selected machine
- Navigate to the Events tab for the selected machine
- Navigate to the Tools tab for the selected machine (or select any function
- Navigate to the Configuration tab for the selected machine
- Create a custom label that displays in the Protected Machines list

If you are managing server clusters from the AppAssure Core, the cluster also appears in the left navigation menu. From the drop-down menu for any cluster, you can also:

- Navigate to the Protected Nodes tab for the selected cluster

Using expandable drop-down sub-menus, you can quickly navigate to certain functions for a machine.

- You can access all functions on the Tools tab for a machine from the expandable drop-down Tools sub-menu.
- You can access all functions on the Configuration tab for a machine from the expandable drop-down Configuration sub-menu.

If you click the arrow to the left of the Protected Machines menu, the list of protected machines and server clusters contracts, and no machines are listed. Clicking again on this arrow causes the list of machines to expand again.

Clicking on any machine in the Protected Machines menu opens the Summary tab for that machine. For more information on what you can accomplish on the Summary tab, see Viewing the Summary tab.

Finally, clicking directly on the Protected Machines menu causes the Machines tab, or the Protected Machines page, to appear in the main navigation area, replacing the Home, Replication, Virtual Standby, Events, Tools, and Configurations tabs.

NOTE: From the Machines tab, you can return to the multiple tab view by clicking the Home icon in the icon bar from the left navigation area.

From the Machines tab, from the Actions menu, you can protect one machine, two or more machines simultaneously (bulk protect), or protect a cluster; you can deploy the Agent software to one machine or multiple machines (bulk deploy); or launch the Restore Machine Wizard.

When you select any of the protected machines, then from the machine-specific configuration menu, you can force a snapshot, force truncation of SQL log files, export to a virtual machine, mount a recovery point, view recovery points for that machine, restore the selected machine, or remove the selected machine from protection.

Viewing the Machines tab for a protected machine

The Machines tab appears as the only tab in the AppAssure Core Console when you click directly on the Protected Machines menu. It contains a Protected Machines pane.

If your Core replicates any machines from another AppAssure Core, then the Machines tab contains a Replicated Machines pane.

Viewing the Protected Machines pane

The Protected Machines pane contains information about all machines protected on this AppAssure Core. For each protected machine (if any are protected yet), you see listed in the grid the information described in the following table.

Table 13. Information about each protected machine

UI Element	Description
Status indicator	Colored circles in the Status column show whether a machine is online or unreachable. If you hover the cursor over the colored circle, the status condition is displayed. Status conditions include green (online and protected), yellow (paused protection), red (authentication error), and gray (offline or unreachable).
Display name	The display name of the protected machine.
Encryption status	The lock icon indicates encryption status for the selected agent machine. An open lock indicates no encryption; a closed lock indicates that encryption keys are established. Click the lock to configure encryption. For more information on encryption, see Understanding encryption keys.
Version	The version of the AppAssure Agent software loaded on the machine.
Snapshot	The next scheduled snapshot for that protected machine. If protection is paused, that status will be indicated instead.

If any of the machines protected in this Core are configured for virtual standby, then you will see additional information as described in the following table.

Table 14. Information about protected machines configured for virtual standby

UI Element	Description
Last export	The date and time of the last virtual export.
Destination	The destination for saving the protected machine as a virtual machine. For example, ESXi, VMware Workstation, Hyper-V, or VirtualBox.
Status	Status of machine configured for virtual standby. Status conditions include “In Sync,” “Paused,” and “Not enabled.”

From the Actions drop-down menu of the Protected Machines pane, you can perform the actions described in the following table.

Table 15. Actions available in the Protected Machines pane

UI Element	Description
Protect Machine	Launches the Protect Machine Wizard. This wizard enables you to protect a single machine on the AppAssure Core. If using advanced options, you can also define a repository on AppAssure Core and establish encryption for the protected agent data. For more information, see Protecting a machine.
Protect Cluster	Opens the Connects to Cluster dialog box, which allows you to connect to a server cluster to specify for protection in the AppAssure Core. For more information, see Protecting a cluster.
Bulk Protect	Launches the Protect Multiple Machines Wizard. This Wizard enables you to establish protection of two or more machines on the AppAssure Core simultaneously by connecting to an Active Directory server or (for virtual machines) a vCenter or ESXi host. For more information, see Protecting multiple machines.

Table 15. Actions available in the Protected Machines pane

UI Element	Description
Deploy Agent	Opens the Deploy Agent dialog box, which allows you to deploy the AppAssure Agent software to a specific single machine and determine the protection settings for that agent machine. For more information, see Deploying the Agent (push install).
Bulk Deploy	Allows you to deploy the latest Agent software available from the Core to multiple machines simultaneously. Opens the Deploy Agent to Machines pane (accessible by selecting Bulk Deploy from the Tools menu). You can bulk deploy using one of several options: • By connecting to an Active Directory server • By connecting to a vCenter or ESXi virtual host. • By specifying the local Core, you can deploy the latest Agent software to the machines currently protected by that core. • By entering a list of machines manually (including hostname or IP address, login credentials and port). • By defining a new individual machine to connect to and protect. For more information, see Deploying to multiple machines.
Restore	Launches the Restore Machine Wizard. This process lets you restore data from recovery point on the Core to a protected machine. For more information, see Restoring volumes from a recovery point.

You can perform actions on two or more of the machines listed in the protected machines grid. To perform actions on multiple machines, select the checkbox for each protected machine, and then, from the expanded Actions drop-down menu, you can perform any of the actions described in the following table.

Table 16. Additional actions available in the Protected Machines pane when machines are selected

UI Element	Description
Protection > Pause or Resume	Lets you pause protection for the selected machines (if protection is active), or lets you resume protection for the selected machines (if protection is paused). For more information, see Pausing and resuming protection.
Force Snapshot	Lets you force an incremental snapshot or a base image for all protected volumes on the selected agent machines. For more information, see Forcing a snapshot.
Replication	Lets you enable, force, copy, pause, or resume replication. For more information, see Managing replication settings.
Remove Machines	Removes the selected machine from protection on the AppAssure Core, letting you choose to either delete or retain the recovery points already on the AppAssure Core. For more information, see Removing a machine.
Cancel	Lets you cancel all currently active operations for the selected machines, or lets you cancel snapshots only that are currently taking place for the selected machines. This does not affect operations scheduled for the future.

From the Configuration drop-down menu for each protected machine, you can perform the actions listed in the following table. Some options appear only for an Exchange server or SQL server, as indicated.

Table 17. Actions available in the Protected Machines pane

UI Element	Description
Force Snapshot	Lets you force an incremental snapshot or a base image for one or more volumes on the selected machine. For more information, see Forcing a snapshot.
Force Log Truncation for Exchange	For an Exchange Server agent machine, forces truncation of the Exchange logs, which frees up space on the Exchange server. For more information, see Forcing log truncation for a SQL machine.

Table 17. Actions available in the Protected Machines pane

UI Element	Description
Force Log Truncation for SQL	For a SQL Server agent machine, forces truncation of the SQL Server logs, which identifies free space on the SQL server. For more information, see Forcing log truncation for a SQL machine.
Export	Launches the Export Wizard. This Wizard let you export recovery point data from a protected machine to a virtual machine in any supported VM format. You can perform a one-time export or set up virtual standby for continual exports. For more information, see Exporting protected data from Windows machines to virtual machines.
Mount	Opens the Mount Recovery Point dialog box, which allows you to browse through snapshot data saved the AppAssure Core to mount a specific recovery point. For more information, see Mounting a recovery point or Mounting a recovery point volume on a Linux machine respectively.
Recovery Points	Opens the Recovery Points tab for the selected agent machine. For more information, see Managing snapshots and recovery points.
Restore	Launches the Restore Machine Wizard. This process lets you restore data from recovery point on the Core to a protected machine. For more information, see Restoring volumes from a recovery point.
Remove Machine	Removes the selected machine from protection on the AppAssure Core, letting you choose to either delete or retain the recovery points already on the AppAssure Core. For more information, see Removing a machine.

Viewing the Replicated Machines pane

If any of the machines protected in this Core are being replicated, then you will see the Replicated Machines pane. This section contains information about all machines replicated on this AppAssure Core, as described in the following table.

Table 18. Information about replicated protected machines

UI Element	Description
Status indicator	Status of replication. Colored circles in the Status column show whether a replicated machine is online or unreachable. If you hover the cursor over the colored circle, the status condition is displayed. Status conditions include green (replication established and online), yellow (replication paused), red (authentication error), and gray (offline or unreachable).
Machine Name	The display name of the replicated machine.
Last Replicated Snapshot	The date and time of the last snapshot for that replicated machine.
Virtual Standby	Indicates whether Virtual Standby has been enabled, signifying parameters have been defined for ongoing export of the replicated machine to a virtual machine.
Replicated from	Indicates the source core from which this protected machine is being replicated.

From the Actions drop-down menu of the Replicated Machines pane, you can perform the actions described in the following table.

Table 19. Actions available in the Replicated Machines pane

UI Element	Description
Export	Lets you export data from the replicated machine to a virtual machine. For more information, see Managing exports.
Mount	Allows you to mount a recovery point. For more information, see Mounting a recovery point.

Table 19. Actions available in the Replicated Machines pane

UI Element	Description
Recovery Points	Opens a view of all recovery points on this replicated machine. For more information, see Viewing recovery points.
Restore	Lets you restore data from a recovery point. For more information, see Restoring data from recovery points.
Delete	Removes the replicated machine from this Core.

You can perform actions on two or more of the machines listed in the replicated machines grid. To perform actions on multiple machines, select the checkbox for each protected machine, and then, from the expanded Actions drop-down menu, select the action you want to perform.

Viewing the Summary tab

The Summary tab appears as the first tab in the AppAssure Core Console when you select a protected agent machine. It contains, at minimum, a Summary pane and a Volumes pane.

Viewing the Summary pane

The Summary pane contains summary information about the protected machine, including the host name, date and time of the last snapshot, date and time of the next scheduled snapshot, encryption key information, and the version of the Agent software loaded on that machine. There is an Actions menu described below.

If you have one or more protected Exchange servers, you will also see an Exchange Server Information pane that contains information about your protected Exchange server. If you have one or more protected SQL servers, you will also see a SQL Server Information pane that contains information about your protected SQL servers.

For all protected machines, from the Actions drop-down menu of the Summary pane, you can do the following:

- **Export a protected machine to a virtual machine**, using a one-time export or continual updates to an exported virtual machine. For more information, see Exporting protected data from Windows machines to virtual machines.
- Pause protection for that machine. For more information, see Pausing and resuming protection.
- **Refresh the metadata for that machine**. To refresh metadata for a machine, from the Actions drop-down menu, click **Refresh Metadata**.
- **Remove that machine from protection**, either choosing to delete or retain the recovery points. More information, see Removing a machine.

The options that appear in the Actions drop-down menu of the Summary pane may differ based on the type of machine you selected. For example, SQL Servers include a SQL option with related functions in the Actions menu; Exchange Servers include an Exchange option in the actions menu.

For protected SQL Server machines, from the Actions drop-down menu of the Summary pane, you can do the following:

- **Force log truncation.** For a SQL Server machine, you can force truncation of the SQL Server logs, which identifies free space on the SQL server. For more information, see Forcing log truncation for an Exchange machine.
- **Set SQL Server credentials.** For a SQL Server machine, you can set default credentials for all instances, or set instance credentials for a single instance. For more information, see Setting credentials for a SQL Server machine.

For protected Exchange Server machines, from the Actions drop-down menu of the Summary pane, you can do the following:

- **Force log truncation.** For an Exchange Server machine, you can force truncation of the Exchange logs, which frees up space on the Exchange server. For more information, see Forcing log truncation for an Exchange machine.

- **Set Exchange Server credentials.** For an Exchange Server machine, you can set or modify Exchange Server settings, including forcing truncation of the Exchange Server logs, or setting credentials for an exchange server instance. For more information, see Setting credentials for an Exchange Server machine.

Viewing the Volumes pane

For any agent machine, from the Summary tab, in the Volumes pane, you can perform the following actions for any of the volumes listed:

- **Set or modify a protection schedule for a selected volume**. Protection schedules are typically established when you first protect a machine. For more information about modifying a protection schedule, see Modifying protection schedules.
- **Force a base image or snapshot.** Snapshots typically occur based on the protection schedule. However, at any time, you can force a base image or an incremental snapshot for selected volumes. For more information, see Forcing a snapshot.

Viewing the Recovery Points tab

The Recovery Points tab shows a list of the recovery points collected for that protected machine as well as pertinent machine and repository data. On this page, you can mount, export, and restore specific recovery points, as well as delete recovery points.

The tab is divided into two panes: Summary and Recovery Points. The Summary pane does not include any actionable links. It displays the following data for the machine.

Table 20. Recovery Point tab Summary pane data

UI Element	Description
Total Recovery Points	The number of recovery points collected for this particular protected machine.
Total Protected Data	The amount of data from the protected machine that is stored in the repository.
Repository	The name of the repository in which AppAssure stores the recovery points for this protected machine.
Repository Status	The progress bar displays the percentage of the total space used in the repository. The amount of data used and the total size of the repository appear below the progress bar.

The Recovery Points pane provides a list of every recovery point in the repository for this protected machine.

Table 21. Recovery Point tab Recovery Points pane options

UI Element	Description
>	Expands the recovery point to reveal additional options.
Mount	After you expand a recovery point, this option lets you mount the recovery point so that you can explore it and recover individual items. For more information, see Mounting a recovery point.
Export	After you expand a recovery point, you can use this option to export the recovery point to a compatible virtual host and create a virtual machine. For more information, see Exporting protected data from Windows machines to virtual machines.
Restore	After you expand a recovery point, the Restore option lets you restore one or more volumes in the recovery point to your preferred location. For more information, see Restoring data or Performing a bare metal restore for Windows machines or, for restoring data on a Linux machine, see Performing a bare metal restore for Linux machines.

Table 21. Recovery Point tab Recovery Points pane options

UI Element	Description
Status	Displays a colored circle that indicates the state of checks performed on the recovery point. For more information, see
Contents	Lists the volumes and partitions included in the recovery point.
Type	Displays whether the recovery point is a base or an incremental snapshot.
Creation Date	The date and time on which AppAssure took the recovery point.
Size	The size of the recovery point in bytes.
Actions	The Actions drop-down menu provides the following options for removing recovery points for the machine: • **Delete Range.** Deletes the range of recovery points you select. • **Delete All.** Deletes all of the recovery points collected for this protected machine. When you delete all recovery points, there are no more recovery points from which to perform a restore. For more information, see Removing recovery points.

Viewing the Events tab for a protected machine

On the Events tab, you can view the jobs that occurred or are in progress the protected machine you selected. Buttons at the top of the page let you navigate to lists of jobs in each of the three categories of activities:

- **Tasks.** A job that the AppAssure must perform to operate successfully.
- **Alerts.** A notification related to a task or event that includes errors and warning.
- **Events.** A composite of all protected machine tasks and alerts.

The following table includes descriptions of each element on the Events tab.

Table 22. Events tab elements

UI Element	Description
Search keyword	Lets you search for a specific item within each category.
From	To narrow your results, you can enter a date at which to begin searching.
To	To narrow your results, you can enter a date at which to stop searching.
Status icons	Each icon represents a different job status. Clicking one of the icons lets you filter the list by that status, thereby generating a report. Clicking the icon a second time removes the filter for that status. You can filter by more than one status. Statuses include: • **Active.** A job that is in progress. • **Queued.** A job that is waiting for another job to complete before it can initiate. • **Waiting.** A job waiting for your approval or completion, such as a seed drive. (For more information about seed drives, see Understanding replication.) • **Complete.** A job that completed successfully. • **Failed.** A job that failed and did not complete.
Service icon	This button adds services jobs to the list of jobs. When you click this icon, a smaller service icon appears on each status icon, which lets you filter by service jobs that have those statuses. Examples of services jobs include deleting index files or removing a machine from protection.

Table 22. Events tab elements

UI Element	Description
Export type drop-down list	The drop-down list includes the formats to which you can export the event report. It includes the following formats: • PDF • XLS • XLSX • RTF • CSV
Export icon	Converts the event report to the format you selected.
Page selection	Event reports can include several jobs across multiple pages. The numbers and arrows at the bottom of the Events tab let you navigate the additional pages of the report.

The Events tab displays all events in a table. The table includes the following information for each item.

Table 23. Events tab table information

UI Element	Description
>	The bracket expands the job to reveal the following details: • **Start Time.** When the job began. • **End Time.** When the job ended. • **Elapsed Time.** The amount of time for which the job was active. • **Rate.** The rate at which the Core transfers the data. • **Progress.** When a job is in progress, it shows how much the Core completed in real time. When the job is complete, it shows how much of the job the Core completed. • **Phase.** When a job is in progress, this detail shows in which stage the progress is (for example, Transferring). • **Cancel.** When a job is in progress, a link here lets you cancel the job. • **Total Work.** When a job is complete, it shows how much of the job the Core completed. • **Child Task.** Lists the sub tasks associated with this job.
Job	The type of job the Core performed, such as transfer of volumes, maintaining repository, or rolling up.
Status	The status of the job, such as Succeeded for a successfully completed job or Error for a job that failed due to errors.
Start Time	The day and time at which the job began.
End Time	The day and time at which the job ended.
Details	Opens the Monitor Active Task window. It includes the job details you see when you expand the job, as well as the following details for the child jobs: • Start Time • End Time • Elapsed Time • Rate • Progress • Total Work

Viewing the Tools tab for a protected machine

The Tools tab shows machine information by default, and lets you access the tools to help manage this particular machine. To access these tools, select an option from the Tools menu on the left, or from the drop-down menu on the Tools tab. These tools are described in the following table.

Table 24. Information about tools accessible to a protected machine

UI Element	Description
System Info	Shows information about the protected machine, system information, volumes, processors, network adapters, and IP addresses for this machine. For more information, see Viewing system information for a machine.
Mounts	From the Local Mounts pane, you can view or dismount volumes mounted locally. From the Remote Mounts pane, you can view or dismount volumes mounted using the Local Mount Utility. For information on dismounting volumes, see Dismounting recovery points. For information on mounting a recovery point locally, see Mounting a recovery point or Mounting a recovery point volume on a Linux machine, respectively.
Diagnostics	The Diagnostics page is there for your convenience in case you should ever encounter an issue for which you want to contact AppAssure Support. Here, you can use the View Log menu option to download the log for this protected machine. For more information, see Accessing protected machine diagnostics.
Reports	The Reports page lets you generate reports about activity for this protected machine and export them in a variety of formats. Report options include the following: • **Compliance.** Provides the status of jobs performed for this machine. • **Failure.** A subset of the Compliance report, it lists only the failed jobs associated with this machine. For more information, see Generating and viewing reports.

Viewing the configuration tab for a protected machine

The Configuration tab for a protected machine (agent) shows settings and configuration options that let you adjust and customize the protection of the selected machine. To access these settings, select an option from the menu on the Configuration page, or from the drop-down menu on the Configuration tab. These options are described in the following table.

Table 25. Configuration tab options

UI Element	Description
Settings	The options available on the Settings page of the Configuration tab apply to the Core and all of its protected machines. It includes the ability to modify settings in the following categories: • **Settings.** The Settings include the display name, host name, port, encryption key (if applicable), and assigned repository for the protected machine. For more information, see Viewing and modifying configuration settings. • **Nightly Jobs.** Nightly jobs are regular maintenance tasks scheduled during off-hours so as not to disrupt your most intensive business hours. They include tasks such as downloading logs from the protected machines, the attachability check job, rollup, and log truncation. For more information, see Customizing nightly jobs for a protected machine.
Events	The Events page of the machine Configuration tab is where you can create custom notification groups for this agent. For more information, see Configuring notification groups for system events.
Retention Policy	On the Retention Policy tab, you can opt to use the default retention policy determined by the Core, or create and use a retention policy customized for this agent. For more information about custom retention policies, see Customizing retention policy settings for a protected machine.
Licensing	On the Licensing page, you can view the expiration date of the license, the license status, type of license, and agent type (virtual or physical). For more information, see Managing licenses.
Transfer Settings	The Transfer Settings page lets you manage the data that transfers between the Core and the agent. You can set the priority, maximum concurrent writes, transfer timeout, and snapshot timeout, and so on, to determine the order in which types of data transfer, how much data can transfer at once time, and how long the Core should attempt a transfer before it stops and moves on to the next transfer job in the queue. For more information, see Modifying transfer settings.

Viewing the replicated machines menu

In the AppAssure user interface, a replicated machines menu appears in the left navigation area. By default, this menu is fully expanded, and shows a list of any machines that are replicated by this Core.

You can collapse or expand the view of replicated machines by clicking the arrow on the left side of this menu.

The Replicated Machines menu includes a drop-down menu on the right side which includes functions you can perform on all replicated machines simultaneously. Click the arrow to the right of Replicated Machines to see the menu and to perform any of the following:

- Force replication. For more information, see Forcing replication.
- Resume replication (if replication has been paused). For more information, see Pausing and resuming replication.
- Pause replication (if replication is currently active). For more information, see Pausing and resuming replication.

Viewing the Recovery Points Only menu

The Recovery Points Only menu appears in the left navigation area if your AppAssure Core retains some recovery points from a machine that was previously protected.

You can collapse or expand the view of recovery points-only machines by clicking the arrow on the left side of this menu.

The menu includes a drop-down menu on the right side which lists functions that can be performed on all recovery points-only machines simultaneously. In this case, the only function you can perform is to remove recovery points from the Core.

> △ **CAUTION: This action removes all of the recovery points-only machines in your AppAssure core, permanently deleting them and precluding you from restoring information from those recovery points from this Core.**

Viewing the Custom Groups menu

The custom groups menu appears in the left navigation area only if you have defined one or more custom groups.

You can collapse or expand the view of items in this menu by clicking the arrow on its left side.

The custom groups menu includes a drop-down menu on the right side which lists functions that can be performed simultaneously on all of the like items in that group.

For more information, see Understanding custom groups.

3

Configuring the AppAssure Core

This chapter describes how to configure the Core in your AppAssure environment in preparation for initial use, including managing licenses and repositories. This chapter also describes how to view information on the Core based on specific configuration, such as with custom groups, and how to back up and restore your Core configuration settings. It includes the following sections:

- Managing licenses
- Understanding repositories
- Managing a repository
- About the repository Integrity Check job
- Understanding custom groups

Before you can use AppAssure, you must configure the AppAssure Core.Initially configuring the AppAssure Core involves understanding certain concepts and performing certain initial operations. At minimum, this includes:

- **Managing your license.** The Core requires a license key or file, which communicates with the Dell AppAssure License Portal. For information on managing your license from the AppAssure Core, see Managing licenses. For more information, see the *Dell AppAssure License Portal Guide*.
- **Creating and managing your repository.** The AppAssure Core uses a repository to store data, which includes defining storage locations to volumes on which your backup data is stored.
    - For general information about repositories, see Understanding repositories.
    - For steps to manage a repository, see Managing a repository.
    - If using replication, and if you want to have different retention policies on source and target cores, then users upgrading from a target core prior to release 5.3.6 must perform an integrity check job. For more information, see About the repository Integrity Check job.

Optionally, you can also configure a custom group of objects available in your Core. This group, which appears in the Protected Machines menu, can be heterogeneous; it can include one or more protected machines, replicated machines, or server clusters, in any combination. If using a custom group, you can perform group actions that apply to all members of that custom group, as described in Performing group actions. For more information on custom groups, see Understanding custom groups.

Other configuration tasks may be required, based on your business requirements. For example:

- **Configuring encryption keys**. The AppAssure Core can encrypt agent snapshot data within the repository, using encryption keys. For more information securing your data by configuring encryption keys, see Understanding encryption keys.
- **Configure event notification**. The AppAssure Core can send notification of specific predefined sets of events using several methods that you can configure. For more information on configuring event notifications, see Managing events. If you want to include email as one notification method, you must configure an email server in the AppAssure Core. For more information, see Configuring an email server.

You can configure your repository, define event notification, and define encryption keys individually, or you can use the Quick Start Guide to guide you through Core configuration processes. The Quick Start Guide allows you to launch wizards and configuration steps, or just to see the recommended flow to accomplish configuration tasks.

If configuring using the Quick Start Guide, you first protect a machine using the Protect Machine Wizard. During this step, you are prompted to establish storage locations, including creating and sizing a repository. This guide will also allow you to configure event notification, configure encryption keys, set up replication, export to a virtual machine, and view or change your retention policy. If you have recovery points on the Core, this guide

will allow you to restore data. For information about the Quick Start Guide, including how to launch or hide this feature, see Understanding the Quick Start Guide.

If performing configuration tasks individually, then you must perform certain initial configuration tasks, such as creating and configuring the repository for storing backup snapshots, optionally defining encryption keys for securing protected data, and optionally setting up notifications for tasks, alerts, and events.IAfter you complete the initial configuration of the AppAssure Core, you can then protect one or more agent machines, and perform recovery. AppAssure includes a default retention policy, which you may want to configure to your custom requirements. If protecting SQL databases, you can configure SQL attachability. You can also create custom groups,

Additional configuration of the AppAssure Core includes the following operations:

- **Protect a single agent machine.** For more information on protecting an agent machine using the Protect Machine Wizard, see Protecting a machine.
- **Protect multiple agent machines.** For more information on protecting multiple agent machines in one step using the Protect Machine Wizard, see Protecting multiple machines.
- **Restoring data from recovery points.** For more information on restoring data from recovery points using the Restore Wizard, see Restoring data from recovery points.
- **Configure retention policy.** For more information on configuring retention policies, see Managing retention policies.
- **Configure SQL attachability.** For more information on configuring SQL attachability, see Configuring SQL attachability settings.

Understanding repositories

A repository is used to store the snapshots that are captured from your protected workstations and servers. The repository can reside on different storage technologies such as Storage Area Network (SAN), Direct Attached Storage (DAS), or Network Attached Storage (NAS).

When you create a repository, the AppAssure Core pre-allocates the storage space required for the data and metadata in the specified location. You can create up to 255 independent repositories on a single core that span across different storage technologies; in addition, you can further increase the size of a repository by adding new file extents or specifications. An extended repository can contain up to 4096 extents that span across different storage technologies.

Key repository concepts and considerations include:

- The repository is based on the AppAssure Scalable Object File System.
- All data stored within a repository is globally deduplicated.
- The Scalable Object File System can deliver scalable I/O performance in tandem with global data deduplication, encryption, and retention management.

NOTE: AppAssure repositories should be stored on primary storage devices. Archival storage devices such as Data Domain are not supported due to performance limitations. Similarly, repositories should not be stored on NAS filers that tier to the cloud as these devices tend to have performance limitations when used as primary storage.

Managing a repository

Before you can use AppAssure, you need to set up one or more repositories on the AppAssure Core server. A repository stores your protected data; more specifically, it stores the snapshots that are captured from the protected servers in your environment.

When you configure a repository, you can perform a variety of tasks such as specifying where to locate the data storage on the Core server, how many locations should be added to each repository, the name of the repository, how many current operations the repositories support, and so on.

When you create a repository, the Core pre-allocates the space required for storing data and metadata in the specified location. You can create up to 255 independent repositories on a single core. To further increase the size of a single repository, you can add new storage locations or volumes.

Managing a repository involves creating, configuring, and viewing a repository and includes the following operations:

- **Access the Core Console.** For more information on how to access the AppAssure Core Console, see Understanding the AppAssure Core Console.
- **Create a repository.** For more information about creating a repository, see Creating a repository.
- **View repository details.** For more information about viewing repository details, see Viewing details about a repository.
- **Modify repository settings.** For more information about modifying repository settings, see Modifying repository settings.
- **Add a new storage location.** For more information on adding a new storage location, see Adding a storage location to an existing repository.
- **Check a repository.** For more information about checking a repository, see Checking a repository.
- **Delete a repository.** For more information about deleting a repository, see Deleting a repository.

Creating a repository

Complete the following steps to create a repository.

To create a repository

1 From the AppAssure Core Console, do one of the following:

- On the Home page, in the Repositories pane, click **Add New Repository**.
- Click the Configuration tab, and from the Repositories page, click **Add new**.

The Add New Repository dialog box displays.

2 Enter the information as described in the following table.

Table 26. Add New Repository settings

Text Box	Description
Repository Name	Enter the display name of the repository. By default, this text box consists of the word *Repository* and an index number, which corresponds to the number of the new repository. You can change the name as needed. You can enter up to 40 characters.
Concurrent Operations	Define the number of concurrent requests you want the repository to support. By default the value is *64*.
Comments	Optionally, enter a descriptive note about this repository. You an enter up to 254 characters.

3 Click **Add Storage Location** to define the specific storage location or volume for the repository.

WARNING: If the AppAssure repository that you are creating in this step is later removed, all files at the storage location of your repository will be deleted. If you do not define a dedicated folder to store the repository files, then those files will be stored in the root; deleting the repository will also delete the entire contents of the root, resulting in catastrophic data loss.

NOTE: AppAssure repositories should be stored on primary storage devices. Archival storage devices such as Data Domain are not supported due to performance limitations. Similarly, repositories should not be stored on NAS filers that tier to the cloud as these devices tend to have performance limitations when used as primary storage.

The Add Storage Location dialog box appears.

4 In the Storage Location area, specify how to add the file for the storage location. You can choose to add the file on the local disk or on CIFS share.

- Select **Add file on local disk** to specify a local machine, and then enter the information as described in the following table.

Table 27. Local disk settings

Text Box	Description
Data Path	Enter the location for storing the protected data. For example, type `X:\Repository\Data`. The same limitations to the path apply; use only alphanumeric characters, hyphen, or period, with no spaces or special characters.
Metadata Path	Enter the location for storing the protected metadata. For example, type `X:\Repository\Metadata`. When specifying the path, use only alphanumeric characters, the hyphen, and the period (only to separate host names and domains). The letters a to z are case-insensitive. Do not use spaces. No other symbols or punctuation characters are permitted.

- Or, select **Add file on CIFS share** to specify a network share location, and then enter the information as described in the following table.

Table 28. CIFS share credentials

Text Box	Description
UNC Path	Enter the path for the network share location. If this location is at the root, define a dedicated folder name (for example, `Repository`). The path must begin with \\. When specifying the path, use only alphanumeric characters, the hyphen, and the period (only to separate host names and domains). The letters a to z are case-insensitive. Do not use spaces. No other symbols or punctuation characters are permitted.
User Name	Specify a user name for accessing the network share location.
Password	Specify a password for accessing the network share location.

5 In the Storage Configuration area, click **More Details** and enter the details for the storage location as described in the following table.

Table 29. Storage location details

Text Box	Description
Size	Set the size or capacity for the storage location. The default is 250 MB. You can choose from the following: • MB • GB • TB **NOTE:** The size that you specify cannot exceed the size of the volume. If the storage location is a New Technology File System (NTFS) volume using Windows XP or Windows 7, the file size limit is 16 TB. If the storage location is a NTFS volume using Windows 8, 8.1 or Windows Server 2012, 2012 R2, the file size limit is 256 TB. **NOTE:** For AppAssure to validate the operating system, Windows Management Instrumentation (WMI) must be installed on the intended storage location.
Write Caching Policy	The write caching policy controls how the Windows Cache Manager is used in the repository and helps to tune the repository for optimal performance on different configurations. Set the value to one of the following: • On • Off • Sync If set to *On*, which is the default, Windows controls the caching. **NOTE:** Setting the write caching policy to *On* could result in faster performance. If you are using a version of Windows Server prior to Server 2012, the recommended setting is *Off*. If set to *Off*, AppAssure controls the caching. If set to *Sync*, Windows controls the caching as well as the synchronous input/output.
Bytes per Sector	Specify the number of bytes you want each sector to include. The default value is *512*.
Average Bytes per Record	Specify the average number of bytes per record. The default value is *8192*.

6 Click **Save**.

The Repositories screen displays to include the newly added storage location.

7 Optionally, repeat Step 3 through Step 6 to add additional storage locations for the repository.

8 Click **Create** to create the repository.

The Repository displays in the Configuration tab.

Viewing details about a repository

Complete the following step to view the details for a repository.

To view details about a repository

1 In the AppAssure Core Console, click the Configuration tab.

The Repositories page displays.

2 Click the right angle bracket > symbol next to the Status column of the repository for which you want to view the details.

3 From the Settings icon next to the Compression Ratio column, you can perform the following actions for a repository:

- Modify Settings
- Add a Storage Location
- Check a Repository
- Delete a Repository

4 Details also display for the repository to include the storage locations and statistics. Storage location details include:

- Metadata Path
- Data Path
- Size

The statistical information available for you to view consist of:

- Deduplication
- Record I/O
- Storage Engine

The level of detail available for Deduplication is reported as the number of block dedupe hits, block dedupe misses, and block compression rate.

The detail rendered for Record I/O consists of the rate (MB/s), read rate (MB/s), and write rate (MB/s).

The storage engine details are include the rate (MB/s), read rate (MB/s), and write rate (MB/s),

Modifying repository settings

After you add a repository, you can modify the repository settings such as the description or the maximum concurrent operations. You can also add a new storage location for the repository. For more information on adding a new storage location, see Adding a storage location to an existing repository.

To modify repository settings

1 In the AppAssure Core Console, click the Configuration tab.

The Repositories page displays.

2 Click the Settings icon next to the Compression Ratio column below the Actions button, and then **Settings**.

The Repository Settings dialog box displays.

3 Edit the repository information as described in the following table.

Table 30. Repository information

Text Box	Description
Repository Name	Represents the display name of the repository. By default, this text box consists of the word Repository and an index number, which corresponds to the number of the repository. **NOTE:** You cannot edit the repository name.
Description	Optionally, enter a descriptive note about the repository.
Maximum Concurrent Operations	Define the number of concurrent requests you want the repository to support.
Enable Deduplication	Clear this checkbox to turn off deduplication, or select this checkbox to enable deduplication. **NOTE:** Changing this setting only applies to backups taken after the setting has been made. Existing data, or data replicated from another core or imported from an archive, will retain the deduplication values in place at the time the data was captured from an agent.
Enable Compression	Clear this checkbox to turn off compression, or select this checkbox to enable compression. **NOTE:** This setting applies only to backups taken after the setting has been changed. Existing data, or data replicated from another core or imported from an archive, will retain the compression values in place at the time the data was captured from an agent.

4 Click **Save**.

Opening an existing repository

Complete the following procedure to open an existing repository.

To open an existing repository

1 On the AppAssure Core Console, click the Configuration tab.

2 On the Repositories page of the Configuration tab, click **Open existing**.

3 On the dialog box, enter the following information for the repository you want to open:

- Path. The path for the repository (for example, D:\work\machine for a local path, or \\servername\sharename for a network path).
- User name. If the repository has a network path, enter the user name for logging in to the network share.
- Password. If the repository has a network path, enter the password for logging in to the network share.

4 Click **Open**.

Adding a storage location to an existing repository

Adding a storage location lets you define where you want the repository or volume to be stored. Complete the steps in the following procedure to specify the storage location for the repository or volume.

To add a storage location to an existing repository

1 Click the Settings icon next to the Compression Ratio column below the Actions button and then **Add Storage Location**.

The Add Storage Location dialog box displays.

2 Specify how to add the file for the storage location. You can choose to add the file on the local disk or on CIFS share.

- Select **Add file on local disk** to specify a local machine and then enter the information as described in the following table.

Table 31. Local disk settings

Text Box	Description
Metadata Path	Enter the location for storing the protected metadata. For example, type `X:\Repository\Metadata`. When specifying the path, use only alphanumeric characters, the hyphen, and the period (only to separate host names and domains). The letters a to z are case-insensitive. Do not use spaces. No other symbols or punctuation characters are permitted.
Data Path	Enter the location for storing the protected data. For example, type `X:\Repository\Data`. The same limitations to the path apply; use only alphanumeric characters, hyphen, or period, with no spaces or special characters.

- Or, select **Add file on CIFS share** to specify a network share location and then enter the information as described in the following table.

Table 32. CIFS share credentials

Text Box	Description
UNC Path	Enter the path for the network share location. If this location is at the root, define a dedicated folder name (for example, `Repository`). The path must begin with \\. When specifying the path, use only alphanumeric characters, the hyphen, and the period (only to separate host names and domains). The letters a to z are case-insensitive. Do not use spaces. No other symbols or punctuation characters are permitted.
User Name	Specify a user name for accessing the network share location.
Password	Specify a password for accessing the network share location.

3 In the Storage Configuration pane, click **More Details** and enter the details for the storage location as described in the following table.

Table 33. Storage location details

Text Box	Description
Size	Set the size or capacity for the storage location.The default size is 250 MB. You can choose from the following: • MB • GB • TB **NOTE:** The size that you specify cannot exceed the size of the volume. If the storage location is a New Technology File System (NTFS) volume using Windows XP or Windows 7, the file size limit is 16 TB. If the storage location is a NTFS volume using Windows 8, 8.1 or Windows Server 2012, 2012 R2, the file size limit is 256 TB. **NOTE:** For AppAssure to validate the operating system, Windows Management Instrumentation (WMI) must be installed on the intended storage location.
Write Caching Policy	The write caching policy controls how the Windows Cache Manager is used in the repository and helps to tune the repository for optimal performance on different configurations. Set the value to one of the following: • On • Off • Sync If set to *On*, which is the default, Windows controls the caching. **NOTE:** Setting the write caching policy to *On* could result in faster performance. If you are using a version of Windows Server prior to Server 2012, the recommended setting is *Off*. If set to *Off*, AppAssure controls the caching. If set to *Sync*, Windows controls the caching as well as the synchronous input/output.
Bytes per Sector	Specify the number of bytes you want each sector to include. The default value is *512*.
Average Bytes per Record	Specify the average number of bytes per record. The default value is *8192*.

4 Click **Save**.

The Repositories screen displays to include the newly added storage location.

5 Optionally, repeat Step 3 through Step 6 to add additional storage locations for the repository.

6 Click **OK**.

Checking a repository

AppAssure provides the ability to perform a diagnostic check of a repository volume when errors occur. Core errors could be the result of it being improperly shut down, a hardware failure, or other environmental, lower IP stack factors that can be exposed in AppAssure functionality.

NOTE: This procedure should only be performed for diagnostic purposes; for example, in the event of hardware failure, improper shutdown of the Core, failure when importing a repository, and so on.

To check a repository

1 Click the Settings icon next to the Compression Ratio column below the Actions button, and then **Check**.

The Check Repository dialog box appears.

2 In the Check Repository dialog box, click **Check**.

NOTE: When you perform a check, all active tasks associated with this repository will be cancelled. Before you the check begins, a message asking you to acknowledge proceeding with the check will display. It is advised and encouraged to rebuild recovery points cache at this point to bring it up to date as the failure of a check will result in you having to restore the repository from an archive.

Deleting a repository

Complete the steps in this procedure to delete a repository.

To delete a repository

1 Click the Settings icon next to the Compression Ratio column below the Actions button, and then **Delete**.

2 In the Delete Repository dialog box, click **Delete**.

A warning message appears to confirm deletion.

3 Click **Yes** to confirm the deletion of the repository.

CAUTION: When a repository is deleted, the data contained in the repository is discarded and cannot be recovered.

About the repository Integrity Check job

In previous releases, replication included the process of copying recovery points from the source core to the target core on a regular basis. Rollup of aging recovery points occurred only at the source core. Combined older recovery points were synchronized daily when running the nightly job.

Starting with version 5.4.1, AppAssure includes the ability to set disparate retention policies between source and target cores, providing the source and target cores are the same version (5.4.1 or later).This allows AppAssure administrators to configure rollup on a target core at a different (presumably faster) rate than on the source core. Similarly, you can now define a custom retention policy for any replicated agent, rolling up recovery points at a faster rate and with less granularity in the target core than on the source core, saving space. For more information, see Customizing retention policy settings for a protected machine.

Some customers have experienced inconsistencies in recovery points that were replicated to a target core prior to AppAssure release 5.3.6. To address this issue, AppAssure release 5.4.1 and later includes a new Integrity

Check job that must be run on each repository before you can configure dissimilar retention policies between the source core and a target core, or configure a custom retention policy on a replicated agent.

When you run the Integrity Check job, the system verifies the integrity of all data stored in the specified repository, ensuring you can recover data from each snapshot or base image. If the integrity check discovers any issue with data in your repository, the job ceases immediately. The event details for that job on the core prompt you to contact Dell AppAssure Support, so you can schedule time to work with a Dell representative to perform additional procedures to identify and remediate data inconsistencies.

> **CAUTION: Running this job is expected to take an extended period of time based on the data in your repository, and the underlying storage system. While the job is running, no other transactions can be performed in that repository, including transfers (snapshot and base image backups, and replication), nightly jobs, and so on.**

You can perform other operations in other repositories while the Integrity Check job is running.

> **NOTE:** This job checks the integrity of *all of the contents* within a repository. For information about the Repository Check job, which you can use to check to ensure a repository is mountable and usable, see Checking a repository.

This is an ad hoc job available to run on each repository. Dell recommends performing the Integrity Check job a single time on each repository on a replicated target core if upgrading from release 5.3.x.

You do *not* need to run this job:

- On a new repository on a target core created in release 5.4.1 or later.
- On a source core.
- If you have already run the Integrity Check Job on this repository.
- If you have not used replication.

For instructions on how to perform this check, see the procedure Running the Integrity Check job on a repository.

Running the Integrity Check job on a repository

Perform this procedure to check the integrity of the entire repository. This is recommended for replicated target cores when upgrading from AppAssure 5.3.x to release 5.4. During the execution of the integrity check, which can be lengthy, no other actions can be performed in the repository.

If you have multiple repositories for a target core, perform this process once for each repository.

> **NOTE:** If you have another repository on the target core for which the Integrity Check job has already been completed, or if you create a new additional repository for this target core, you can perform operations in that secondary repository while the integrity check job is running on your primary repository.

To run the repository integrity check job

1. Navigate to the AppAssure Core Console, click the **Configuration** tab, and then click **Repositories**.

   The Repositories page appears, displaying the list of repositories associated with this Core.

2. From the drop-down menu for the repository you want to check, select **Integrity Check**.

   A confirmation message appears.

> **CAUTION: Before you confirm that you want to perform the job, you should carefully consider the duration of time required. While the job is running, no other transactions can be performed in that repository, including transfers (snapshot and base image backups, and replication), nightly jobs, and so on.**

3. From the Integrity Check Repository dialog box, to perform the integrity check, click **Yes**.

   The Check Repository dialog box closes, any jobs that were queued or that are in progress are canceled, and the integrity check job beings.

4 To monitor the progress of the Integrity Check job for a repository, including a determination of whether additional steps are required after the check, click the **Events** tab.

5 From the Events tab, click **Show details** for the job to view more information about the job status.

- If you see an error in any child tasks for this job, note the error and provide the information to a Dell technical support representative.
- If the Integrity Check Job completes all child tasks successfully, you can then establish a custom retention policy for this repository.

Understanding custom groups

The AppAssure Core shows a Protected Machines menu in the left navigation area. This includes all machines added to protection on your AppAssure Core. If you have protected clusters, these also appear in the Protected Machines menu. Beneath this, if you have replicated machines, these appear in a replicated machines menu under the name of the replicated Core. You can perform group actions for all machines arranged under one of these menus, by selecting the arrow to the right of the menu name to access the drop-down menu.

In the same manner, you can create a custom group to display in the left navigation area, as described in Managing licenses.

The act of creating a group always adds one group member (a protected machine, server cluster, or replicated agent, based on how you initiated the group creation) to the new custom group. Ideally, you would then add additional members to the group. Thereafter, you can perform group actions that apply to all members of that custom group, as described in Performing group actions.

Custom groups can include protected agents, replicated agents, and server clusters. Server clusters behave the same as protected agents, with the exception that a server cluster and its nodes behave as a single entity. If you attempt to add a node from a server cluster to a group, the entire cluster is added.

A custom group may contain similar or dissimilar members. For groups of similar members, all group actions apply to all members of the group. For example, if you force a snapshot for a custom group of protected agents, each agent will be backed up. For groups with dissimilar members (for example, protected agents and replicated agents), if you apply a group action such as forcing replication, this will only apply to the replicated agents.

You can create one or more groups. A single agent or replicated machine can be included in one or more groups. This way, you can group machines on your core in any way you choose, and can perform actions on that specific group.

Each custom group appears in the left navigation area, with a label you designate. Groups with standard protected agents appear first in the custom group, and replicated agents appear below, as applicable.

Including a machine in a group does not remove it from its original location. For example, if you have three protected machines called Agent1, Agent2, and Agent3, and you add Agent1 to CustomGroup1, then Agent1 appears in both locations.

For more information, see the following topics:

- Managing licenses
- Modifying custom group names
- Removing custom groups
- Performing group actions
- Viewing all machines in a custom group on one page

Creating custom groups

When you scroll your cursor over the name of any machine in the Protected Machines or replicated machines menu, you will see an arrow that opens a drop-down menu. From this menu, you can create a custom label.

Use the procedure below to create a custom group.

To create a custom group

1 Navigate to the AppAssure Core Console.

2 From the Protected Machines or replicated machines menu, do the following:

   a Click on a machine in the menu.

   b Click on the drop-down menu for that machine.

   c Scroll down and select **Label as**, and then click **New label**.

   A new menu appears, with a blank text box next to a label icon. The machine you selected is listed under the custom group.

3 Enter an appropriate label for your custom group.

   Use a descriptive name that communicates the purpose of the group. For example, to group agent machines by department, type Accounting Department. You can rename a group later.

   NOTE: Labels must be 50 or fewer characters. You can include a single space between words. You must provide a label for your custom group.

4 When you are satisfied with the label name, click the green check mark to save the name.

   The page refreshes, showing your custom group in the navigation area.

5 Optionally, to add other agents to this group, navigate to the agent name in the appropriate menu, click to open the drop-down menu, scroll down and select **Label as**, and then click the name of the custom group.

   You can now perform group actions on this group. For more information, see Performing group actions.

Modifying custom group names

When you modify the name of a custom group, only the label changes. The machine names remain the same.

Use the procedure below to modify a custom group name.

To modify a custom group name

1 Navigate to the AppAssure Core Console.

2 In the Protected Machines menu, scroll your cursor over the custom group you want to modify.

3 Click on the drop-down menu for that group, and then click **Edit**.

   The name of the custom group becomes editable.

4 Enter a new label or your custom group. When you are satisfied with the label name, click the green check mark to save the name. You can edit this name later.

   Use a descriptive name that communicates the purpose of the group. For example, to group agent machines by department, type Accounting Department.

   NOTE: Labels must be 50 or fewer characters. You can include a single space between words. You must provide a label for your custom group.

5 Optionally, to add other agents to this group, navigate to the agent name in the appropriate menu, click to open the drop-down menu, scroll down and select **Label as**, and then click the name of the custom group.

Removing custom groups

When you remove a custom group, you delete that group from the Protected Machines menu. The machines that were in the group are not removed, and can still be found in the appropriate standard menu.

Use the procedure below to remove a custom group.

To remove a custom group

1 Navigate to the AppAssure Core Console.

2 In the Protected Machines menu, scroll your cursor over the custom group you want to remove.

3 Click on the drop-down menu for that group, and then click **Remove label**.

You see a message asking to confirm the removal of the group. Confirm the action.

The page refreshes, and the custom group is removed from the navigation area.

Performing group actions

You can perform group actions on any group appearing in the left navigation area of the AppAssure Core Console. If the group contains dissimilar members (for example, standard agent machines and replicated agents), then the actions you request will only be performed on the relevant group members.

Use the procedure below to perform group actions on a custom group.

To perform group actions

1 Navigate to the AppAssure Core Console.

2 In the Protected Machines menu, scroll your cursor over the custom group for which you want to perform a group action.

3 Click on the drop-down menu for that group, and then select an action as follows:

- To force an incremental snapshot or base image for a protected agent, click Force Snapshot or Force Base Image, as appropriate. For more information, see Forcing a snapshot.
- To pause protection for a protected agent, click **Pause Protection** and then specify resumption parameters. For more information, see Pausing and resuming replication.
- To resume protection for an agent for which protection has been paused, click Resume Protection and then confirm that you want to resume. For more information, see Pausing and resuming replication.
- To refresh the information displayed, click **Refresh Metadata**.
- To pause replication for a replicated agent on the target core, under Replication, click **Pause**. For more information, see Pausing and resuming replication.
- To resume replication for a replicated agent for which replication has been paused on the target core, under Replication, click **Resume**. For more information, see Pausing and resuming replication
- To force replication for a replicated agent machine, click **Force**. For more information, see Forcing replication.
- For custom groups only, to modify the label for the custom group, select **Edit**. For more information, see Modifying custom group names.
- For custom groups only, to remove the custom group, select **Remove label**. For more information, see Removing custom groups.

Viewing all machines in a custom group on one page

Clicking the name of a custom group takes you to a Machines tab that lists all the machines in that custom group. You can then perform some functions on all machines from the Actions menu, or you can perform functions individually by selecting commands from each individual machine.

Backing up and restoring the Core configuration

When you configure an AppAssure Core, you can then back up all of the core settings that are stored in the Windows registry. If you perform this backup, these settings are then available in the future to restore to an AppAssure Core from the configuration file. This is useful in the case where you have restored or replaced a core machine due to upgrade or after a data loss.

Optionally, when restoring a backed-up configuration file, you can also choose to restore configuration information for the repositories. Finally, you can restore the path associated with data or metadata storage. This is useful if those settings have been inadvertently changed.

NOTE: This process restores the configuration settings only, not the data.

Security information (such as authentication credentials) is not stored in the configuration file. There is no security risk to saving a Core configuration file.

This section includes the following topics:

- Backing up the Core configuration
- Restoring a backed-up Core configuration

Backing up the Core configuration

You can back up settings from your Core configuration, to be restored at a future time. Complete the steps in the following procedure to back up Core configuration information to an XML file.

CAUTION: If your Core configuration settings change, then you should back up this configuration information after those changes to maintain the latest configuration information.

Dell recommends you move the backup file to a known location on another machine, a network drive, or on storage media retained in an offsite storage location.

To back up the Core configuration

1. Navigate to the AppAssure Core, click the Configuration tab, and then click **Backup/Restore**.

   The Backup Core Configuration pane appears.

2. In the Backup Core Configuration pane, in the Local path text field, type the path of a local directory on the server with the AppAssure Core, and then click **Backup**.

   For example, type C:\Users\administrator\Documents\Config and then click Backup.

   The file is saved as AppRecoveryCoreConfigurationBackup.xml in the directory you specified.

CAUTION: Dell recommends you move the backup file to a known location on another machine, a network drive, or on storage media retained in an offsite storage location.

Restoring a backed-up Core configuration

If you have a Core configuration file backed up, you can restore this XML file to an AppAssure Core. This is useful in the case where you have restored or replaced a core machine due to upgrade or after a data loss. After you restore the backup file, then configuration settings in the backup file are restored to the AppAssure Core. This includes the names of protected machines, replication relationships, virtual standby information, and encryption keys, as applicable.

Complete the steps in the following procedures to restore a Core configuration file.

NOTE: To restore, the configuration file must be stored locally on the Core machine.

To restore a backed-up Core configuration

1. If restoring the backup file from a network drive or other location, copy the AppRecoveryCoreConfigurationBackup.xml file to a local drive on the Core machine and note the location.
2. Navigate to the AppAssure Core, click the Configuration tab, hover over **Backup/Restore,** and click **Restore**.

   The Restore Core Configuration pane appears.
3. In the Restore Core Configuration pane, in the Local path text field, type the local path of the backup file.
4. Optionally, if you want to restore configuration information for any repositories backed up to the configuration file, select **Restore Repositories**.
5. Note the cautions indicated. If the restore process does not complete as expected, check each of these factors.
6. Then click **Restore**.

   Your configuration is restored to the AppAssure Core.

4

Managing AppAssure Core settings

This chapter describes how to manage and change the settings for your AppAssure Core from the Configuration tab or icon. It includes the following sections:

- Understanding system information
- Configuring update settings
- Understanding nightly jobs
- Modifying transfer queue settings
- Modifying transfer queue settings
- Adjusting client timeout settings
- Understanding deduplication cache size and storage locations
- Configuring Replay engine settings
- Configuring deployment settings
- Accessing diagnostics for the Core
- Configuring database connection settings
- Changing a license
- Backing up and restoring Core settings
- Configuring Core job settings

The AppAssure Core settings are configured by default for optimum performance for most users. These settings affect the performance of the AppAssure Core, or in some cases the display of information in the AppAssure Core Console. You can access most Core settings from the Configuration tab by clicking Settings. From the same tab, you can also access Licensing, Backup/Restore, or Job Settings, as appropriate. The comprehensive set of AppAssure Core settings that you can configure is described in the following table.

Table 34. AppAssure Core configurable settings

Configuration Setting	Description
System Information	System information includes data about the AppAssure Core server. You can see the host name, OS, architecture and memory for the Core. You can see or change the name displayed on the Core Console. You can also view the fully qualified domain name of the Core on your network, and the path for your cache metadata and deduplication caches. System information is available from the Tools menu. For more information about changing the display name, see Understanding system information. For more information about deduplication cache, see Understanding deduplication cache size and storage locations. For information on adjusting the settings, see Configuring deduplication cache settings.
General	General settings include configuration options that apply generally to the AppAssure Core, including display options and ports for the web server and for the AppAssure service. You access general settings by selecting General settings include: • Display name • Web server port • Service port • Language For more information about the general settings for AppAssure Core, including how to configure these settings, see Understanding system information.
Updates	Update settings controls aspects of the Automatic Update feature, which checks for updated versions of AppAssure software. Update settings include: • Check for new updates • Install updates • Status • Last check For more information about settings for updating the AppAssure Core, including how to configure these settings, see Configuring update settings.
Nightly jobs	Nightly jobs settings are automated tasks which the Core performs on a daily basis. You can configure the time the jobs begin and which jobs are performed. Dell recommends scheduling the jobs outside of normal business hours to reduce load on the system when demand is high. For more information about nightly jobs, see Understanding nightly jobs. For more information about how to configure these settings for the Core, see Configuring nightly jobs for the Core. For more information about configuring nightly jobs for specific protected machines, see Customizing nightly jobs for a protected machine.
Transfer queue	Transfer queue settings apply to the number of times transfer operations are attempted if jobs fail due to unavailability of resources. You can establish the maximum number of concurrent transfers and the maximum number of retries for transferring data. For more information about transfer queue settings, see Modifying transfer queue settings.
Client timeout	Client timeout settings determine the length of time before that specific connection requests or read and write operations should be attempted before timing out. For more information about client timeout settings, see Adjusting client timeout settings.

Table 34. AppAssure Core configurable settings

Configuration Setting	Description
Deduplication cache	AppAssure deduplicates your data globally across a repository. Deduplication ensures that unique blocks of information are stored only once in your repository, creating references to repeated data blocks. The references are stored in a deduplication cache. If encryption keys are used, then deduplication occurs within each encryption domain. The deduplication cache settings let you configure the size and specify the locations for the primary and secondary cache as well as the location for the metadata cache. For more information about deduplication cache, see Understanding deduplication cache size and storage locations. For information on adjusting the settings, see Configuring deduplication cache settings.
Replay Engine	Replay engine settings control information regarding the communication channel for the Replay engine. Settings include IP addresses, port numbers, automatic port assignment, buffer sizes for sending and receiving information, and timeout settings to help adjust the performance specific to your network needs. For more information about engine settings for AppAssure, see Configuring Replay engine settings.
Deploy	Deploy settings let you set options for deploying the AppAssure Agent software from your Core to the machines you want to protect. AppAssure lets you specify the name of the installer file, Core address, timeout settings, and limit the number of concurrent installations (to prevent overload of your network). You can also specify whether protected machines reboot, and whether machines are automatically protected after the AppAssure Agent software is deployed. For more information about configuring deployment settings, see Configuring deployment settings.
Database connection	AppAssure displays information about Core tasks, events, and alerts on the Events tab. AppAssure stores this transactional information in a MongoDB service database that is installed locally by default on the Core machine. You can configure these settings to change how long information is retained in the database, or to change the connection pool size to allow for more or fewer connections concurrently. If using a second AppAssure Core, you can configure the database connection settings on the first Core to point to the second Core machine. In this way, the event data for both Cores will be stored in the MongoDB on the second Core. Alternatively, you can configure the database connection settings on the Core to point to another machine that has a separately installed MongoDB which is accessible over the network to the AppAssure Core. The event transaction data for your Core is then saved to that service database, not locally. For more information about establishing or modifying database connection settings for the service database, see Configuring database connection settings.
SMTP server	The AppAssure Core monitors tasks, events and alerts. This information is visible on the Events tab. If you configure simple mail transfer protocol (SMTP) server settings for the Core, you can also send this information by email. These settings include the SMTP server name, originating email address (“From” address), login credentials, port, timeout and security settings for that email server. For more information about configuring an SMTP email server, see Configuring an email server. **NOTE:** To send event information by email, you must also configure notification group settings. For more information on specifying events to receive email alerts, see Configuring notification groups.

Table 34. AppAssure Core configurable settings

Configuration Setting	Description
Trace logs	Trace log settings can be configured to record a significantly higher amount of events and states in the AppAssure Core than standard logs. This functionality is specifically intended to be used for problem resolution only. **NOTE:** Dell does not recommend changing these settings unless requested by Dell Support. Maintaining a high level of logging for a long period can significantly impact storage space and may impact performance.
Cloud configuration	The Cloud Configuration settings let you specify configuration settings for various cloud storage accounts, including Microsoft Azure, Amazon S3, and managed cloud providers using OpenStack open source technology. These settings do not create cloud accounts. Instead, they associate your existing cloud accounts with your AppAssure Core to facilitate actions such as archiving AppAssure information. For more information about managing cloud storage account information in the AppAssure Core, see Managing cloud accounts.
Licensing	From the Core console, AppAssure lets you change the license associated with your Core, view license pool information, and contact the license server. For more information about managing licenses from the Core, see Managing licenses. For more information about managing licenses, see the Dell AppAssure License Portal Guide. **NOTE:** The Dell AppAssure License Portal has a different release cycle than AppAssure software. For the latest product documentation, see the Dell Technical Documentation website.
Core settings backup and restore	AppAssure lets you back up Core configuration settings to an XML file. If you have a backup file, you can use it to restore or migrate Core settings. For more information about backing up and restoring Core settings, see Backing up and restoring Core settings.
Jobs	Core jobs are automatically created whenever you initiate operations such as replication. The Job Settings section of the Configuration tab lets you determine the settings for each job. You can configure the number of jobs to execute at one time. In case network or other communication errors prevent any job from succeeding the first time, you can set how many times a job should be attempted. For more information about Core jobs, which jobs are available, and how to configure them, see Configuring Core job settings.
Diagnostics (Core Logs)	For diagnosing possible issues, you can download and view logs for your AppAssure Core, or you can upload logs for your Core an any protected machines. These features are available from the Tools menu. For information about downloading or uploading logs, see Accessing diagnostics for the Core.

Local database	AppAssure displays information about Core tasks, events, and alerts on the Events tab. AppAssure stores this transactional information in a MongoDB service database that is installed locally on the same machine as the AppAssure Core. You can configure credential information (username and password) for the local Mongo service database using the Local database settings.
SNMP configuration	Simple Network Management Protocol (SNMP) is a protocol for managing devices on an IP network. SNMP is used primarily to monitor devices on a network for conditions that require attention. This protocol uses software components (agents) to report information to administrative computers (managers). An SNMP agent handles the manager's requests to get or set certain parameters. The SNMP agent can also sends traps (notifications about specific events) to the manager). Data objects managed by SNMP agents are organized into a Management Information Base (MIB) that contains Object Identifiers (OIDs). Each OID identifies a variable that can be read or set using SNMP. You can configure the AppAssure Core as an SNMP agent. The Core then can report information such as alerts, repository status, and protected machines. This information can be read by an SNMP host using a standalone application called an SNMP browser. You can install the SNMP browser on any machine accessible over the network to the AppAssure Core. Use the SNMP settings to control communication between the Core and the SNMP browser. You can also download a MIB file from the AppAssure Core. From the Tools tab, select Downloads and download the MIB from the Other Files pane. This file can be read by an SNMP browser in a more user-friendly fashion than data it receives directly from the Core.
Reports	Report settings include a single configuration parameter that allows you to select the font used when a report is generated from the

Understanding system information

AppAssure lets you view information about the AppAssure Core that includes system information, local and mounted volumes, and Replay engine connections.

In the System Info pane, you can see the information described in the following table.

Table 35. System information

UI Element	Description
Host name	The machine name of your AppAssure core.
OS version	The version of the operating system installed on the AppAssure core.
OS architecture	Lists the underlying structure and design of the machine hosting your AppAssure core. Potentially includes chipset and lists 64-bit system.
Memory (physical)	Lists the amount of Random Access Memory installed on the Core machine.
Display name	Shows the display name of the Core, which is configurable (see Understanding system information).
Fully qualified domain name	Shows the fully qualified domain name for the Core machine.
Metadata cache location	Shows the path of the metadata cache location. For more information, see Understanding deduplication cache size and storage locations.
Primary cache location	Shows the path of the primary deduplication cache location. For more information, see Understanding deduplication cache size and storage locations.
Secondary cache location	Shows the path of the secondary deduplication cache location. For more information, see Understanding deduplication cache size and storage locations.

The Volumes pane includes the following information about storage volumes for the Core machine: Name, device ID, file system, raw capacity, formatted capacity, used capacity, and mount points.

The Replay Engine Connections pane displays detailed information about currently mounted recovery points. You can view the ID, local end point, remote end point, mounted image agent ID, mounted image ID, and the mounted image display name. You can see if the mount is writable, view the authenticated user, bytes read, and bytes written.

You can dismount recovery points that are mounted locally on a core from the Mount option on the Tools tab. For more information about dismounting recovery points, see Dismounting recovery points.

Viewing system information

Complete the steps in this procedure to view system information.

To view system information

1. Navigate to the AppAssure Core, and then select the Tools tab.
2. From the Tools option, click **System Info**.

Configuring the Core general settings

Complete the steps in this procedure to update the general settings for the AppAssure Core, which include the Core display name, the web server port, service port, and the display language for the AppAssure Core.

To configure the Core general settings

1 Navigate to the AppAssure Core Console and click the Configuration tab, and then **Settings**.

The General Settings dialog box appears.

2 In the General pane, click **Change**.

The General Settings dialog box displays.

3 Enter the configuration information as described in the following table.

Table 36. General Settings information

Text Box	Description
Display name	Enter a new display name for the Core. This is the name that will display in the AppAssure Core Console. You can enter up to 64 characters.
Web server port	Enter a port number for the Web server. The default port is *8006*.
Service port	Enter a port number for the AppAssure Core service. The default port is *8006*.
Language	From the Language drop-down list, select the language you want to display. International versions of AppAssure support English, French, German, Japanese, Korean, Portuguese, Simplified Chinese, and Spanish.

4 Click **OK**.

5 If changing the languages, confirm the message indicating that the AppAssure Core service must restart before the updated language can display in the Core Console.

Configuring update settings

AppAssure includes the Automatic Update feature. When installing the AppAssure Core, you can choose whether to automatically update the AppAssure Core software when new updates are available, and how frequently the system should check for updates.

NOTE: For information on installing AppAssure Core software, see the *Dell AppAssure Installation and Upgrade Guide*.

You can view and change the settings the system uses to check for updates at any time.

CAUTION: When using replication, configuring your system to install updates automatically could result in upgrading the source core before the target core, which may result in replication failure or the inability to set up new replication between cores. For replication users, Dell recommends administrators apply automatic upgrades only to the target core, and then manually upgrade the source core, and lastly upgrade the protected machines.

Complete the steps in this procedure to configure update settings.

To configure update settings

1 Navigate to the AppAssure Core Console and click the Configuration tab, and then **Settings**.

2 In the Updates pane, click **Change**.

The Update Settings dialog box displays.

3 In the Check for new updates text box, select how frequently AppAssure checks for and installs updates after the nightly jobs are complete. You may choose from the following options:

- Never
- Daily
- Weekly
- Monthly

4 Specify the handling of available updates by choosing one of the following options:

- **Install them automatically (recommended)** or
- **Notify me but do not install them**

5 Click **OK**.

Understanding nightly jobs

Nightly jobs are daily automated tasks that occur at a predetermined time outside of normal business hours. These jobs are memory-intensive, and include various integrity checks and data consolidation tasks that are best conducted when the AppAssure Core is less active.

Nightly jobs can be managed at the Core level, or for each protected machine. All of the nightly jobs, and the scope for which they can be applied, are described in the following table.

Table 37. Nightly jobs

Job Name	Scope	Description
Attachability check	Protected Machine	Checks the integrity of recovery points containing SQL databases. Process: • Mount the latest recovery point for protection groups containing databases. • Connect to the database from SQL Server. • Open the database. • Close the database. • Dismount the recovery point. To enable this nightly check, specify a SQL Server instance to use to perform attachability checks for SQL Server databases on protected machines. **NOTE:** This option does not appear if you are not protecting a SQL Server in your Core.
Check recovery points integrity	Core or Protected Machine	Checks the integrity of recovery points for each protected machine. By default, the Check recovery points integrity option is not enabled. Process: • Mount the latest recovery point for every protection group. • Enumerate the files and folders for each volume. • Examines the recovery points to ensure that they are valid. • Dismount the recovery point.

Table 37. Nightly jobs

Job Name	Scope	Description
Checksum check	Protected Machine	Checks the integrity of recovery points containing Exchange Message Databases (MDBs). **NOTE:** This option does not appear if you are not protecting an Exchange Server in your Core.
Deleting old events and jobs	Core	Maintains the scale of the events database by removing old events. The number of days is configurable, defaulting to 30 days.
Downloading the logs from the protected machines	Core	Downloads logs for protected machines to the Core, so they can be sent to the logging server.
Log repository statistics	Core	Sends repository statistics to the logging server.
Rollup	Core	Maintains the retention policy by combining backup images. The retention policy is defined on a per-Core basis, but can be overridden per protected machine. The rollup job is run for the whole Core.
Log truncation	Protected Machine	Maintains the size of SQL Server logs by truncating the database transaction log to match the last recovery point. **NOTE:** This option does not appear if you are not protecting a SQL Server in your Core.
Truncate Exchange logs	Protected Machine	Maintains the size of Exchange logs, by truncating the exchange database transaction log to match the last recovery point. **NOTE:** This option does not appear if you are not protecting an Exchange Server in your Core.

The following procedures apply only to the Core-level nightly jobs. For more information about nightly jobs specific to protected machines, see Customizing nightly jobs for a protected machine.

Configuring nightly jobs for the Core

When any nightly job option is enabled on the AppAssure Core, the selected job executes once daily at the time specified for all machines that are protected by the Core.

Conversely, if you disable any nightly job at the Core level, the specified job no longer executes for all machines protected by the Core.

NOTE: If the scope of a nightly job includes protected machines, you can apply that nightly job only for specific protected machines individually. The scope for each nightly job is described in the topic Understanding nightly jobs. For more information about applying nightly job settings specific to a protected machine, see Customizing nightly jobs for a protected machine.

Because nightly jobs are memory-intensive, Dell recommends configuring your Core to execute them during a time of low activity. The default schedule to run nightly jobs is 12:00 am. If another time is more suitable, change this setting in the Nightly Jobs Time field using this procedure.

To configure nightly jobs for the Core

1. Navigate to the AppAssure Core Console and click the Configuration tab, and then **Settings**.
2. In the Nightly Jobs pane, click **Change**.

   The Nightly Jobs dialog box displays.
3. If you want to change the time nightly jobs execute, enter a new time in the Nightly Job Times text box.

4 Select the nightly jobs option you want to set for the Core. You can choose to select all or individually select only those that apply.

5 Click **OK**.

Modifying transfer queue settings

Transfer queue settings are core-level settings that establish the maximum number of concurrent transfers and the maximum number of retries for transferring data.

Complete the steps in this procedure to modify transfer queue settings.

To modify the transfer queue settings

1 Navigate to the AppAssure Core Console and click the Configuration tab, and then **Settings**.

2 In the Transfer Queue pane, click **Change**.

The Transfer Queue dialog box displays.

3 In the Maximum Concurrent Transfers text box, enter a value to update the number of concurrent transfers.

Set a number from 1 to 60. The smaller the number, the lesser the load on network and other system resources. As the number of agents that are processed increases, so does the load on the system.

4 In the Maximum Retries text box, enter a value to update the maximum number of retries.

5 Click **OK**.

Adjusting client timeout settings

Client timeout settings control the length of time that various operations are attempted before the operation times out.

NOTE: Dell recommends leaving default timeout settings unless you experience specific issues in your environment, and you are advised by a Dell Support representative to modify the settings.

These settings let you configure the parameters described in the following table.

Table 38. Timeout settings

Setting	Description
Connection Timeout	Controls the timeout for the connection between the Core and protected machines when sending data across the hypertext transfer protocol (http).
Read/Write Timeout	Controls the timeout for the connection between the Core and protected machines when reading or writing stream data across http. An example is receiving changed data blocks from a protected machine to the Core for an incremental snapshot.
Connection UI Timeout	Controls the timeout for the connection between the graphic user interface and the Core service across http.
Read/Write UI Timeout	Controls the timeout for the connection for reading and writing data streams between the graphic user interface and the Core service across http.

Complete the steps in this procedure to adjust client timeout settings.

To adjust client timeout settings

1. Navigate to the AppAssure Core Console and click the Configuration tab, and then **Settings**.
2. In the Client Timeout Settings Configuration area, click **Change**.

   The Client Timeout Settings dialog box displays.
3. In the Connection Timeout text box, enter the number of minutes and seconds before a connection time out occurs.
4. In the Connection UI Timeout text box, enter the number of minutes and seconds before a connection UI time out occurs.
5. In the Read/Write Timeout text box, enter the number of minutes and seconds you want to lapse before a time out occurs during a read/write event.
6. In the Read/Write UI Timeout text box, enter the number of minutes and seconds that will to lapse before a read/write UI time out occurs.
7. Click **OK**.

Understanding deduplication cache size and storage locations

Global deduplication reduces the amount of disk storage space required for your backed up data. The AppAssure deduplication volume manager (DVM) combines a set of storage locations into a single repository. Each repository is deduplicated, storing each unique block once physically on disk, and using virtual references or pointers to those blocks in subsequent backups. To identify duplicate blocks, AppAssure includes a deduplication cache that holds references to unique blocks.

By default, this deduplication cache is 1.5 GB in size. This is a sufficient size for many repositories. Until this cache is exceeded, your data is deduplicated across the repository. Should the amount of redundant information be so large that the deduplication cache is full, your repository can no longer take full advantage of further deduplication across your repository for newly added data. The amount of data saved in your repository before the deduplication cache is full varies by the type of data being backed up and is different for every user.

You can increase the size of the deduplication cache by changing the deduplication cache configuration in the AppAssure Core. For more information on how to accomplish this, see the topic Configuring deduplication cache settings.

When you increase the deduplication cache size, there are two factors to consider: disk space and RAM usage.

Disk space. Two copies of the cache are stored on disk: a primary cache, and a secondary cache which is a parallel copy. So with a default cache size of 1.5 GB, 3 GB of disk storage is used in your system. As you increase the cache size, the amount of disk space used remains proportionally twice the size of the cache. To ensure proper and fault-resistent performance, the Core dynamically changes the priority of these caches. Both are required, the only difference being that the cache designated as primary is saved first.

RAM usage. When the AppAssure Core starts, it loads the deduplication cache to RAM. Memory usage for your system is therefore affected by the size of the cache. The total amount of RAM used by the Core depends on many factors including which operations are running, the number of users, the number of agents, as well as the size of the deduplication cache. Each operation performed by Core (transfer, replication, rollup, and so on) consumes more RAM. Once an operation is finished, memory consumption decreases accordingly. However, administrators should consider the highest RAM load requirement for efficient operations.

Default settings for the AppAssure Core place the primary cache, secondary cache, and the metadata cache in the AppRecovery directory used by AppAssure.

NOTE: Depending on your settings, the AppRecovery directory may not be visible on the AppAssure Core. To see this directory, you may need to change the Folder Options control panel to show hidden files, folders, and drives.

Assuming the AppAssure Core is installed on the C drive, these locations are typically as follows:

Table 39. Default storage locations for deduplication cache settings

Setting	Default Storage Location
Primary Cache Location	C:\ProgramData\AppRecovery\RepositoryMetaData\PrimaryCache
Secondary Cache Location	C:\ProgramData\AppRecovery\RepositoryMetaData\SecondaryCache
Metadata Cache Location	C:\ProgramData\AppRecovery\RepositoryMetaData\CacheMetadata

You have the ability to change the storage location of these caches. For example, for increased fault tolerance, you can change location of your secondary cache to a different physical drive than the primary, assuming the AppAssure Core has access to the location.

For more information on how to change storage locations for any of these settings, see the topic Configuring deduplication cache settings.

Configuring deduplication cache settings

Complete the steps in this procedure to configure deduplication cache settings.

To configure deduplication cache settings

1. Navigate to the AppAssure Core Console and click the Configuration tab, and then **Settings**.
2. In the Deduplication Cache Configuration pane, click **Change**.

   The Deduplication Cache Configuration dialog box displays for you to specify the cache storage locations and cache size.

   Since the primary and secondary cache are the same size, collective storage for these two caches requires twice the amount of space as the amount allocated for the deduplication cache size. For example, if you specify the default amount of 1.5 GB for the deduplication cache size, you will need to ensure that each of the two storage locations have at least 1.5 GB. In particular, if both locations belong to the same drive (for example, the C drive), there must be at least 3.0 GB of free disk space.
3. If you want to change the primary cache location, then in the Primary Cache Location text box, type the path for a storage location accessible to the Core.
4. If you want to change the secondary cache location, then in the Secondary Cache Location text box, type the path for a storage location accessible to the Core.
5. If you want to change the metadata cache location, then in the Metadata Cache Location text box, type the path for a storage location accessible to the Core.
6. If you want to change the deduplication cache size, then do the following:
   a. In the Dedupe Cache Size text box, enter a value corresponding to the amount of space you want to allocate for the deduplication cache.

      For example, if you want to increase the deduplication cache size to 3 GB, enter 3 in this field.
   b. From the unit size drop-down field, select either GB (gigabytes) or TB (terabytes), as appropriate, to specify the unit of measurement for the value in the Dedupe Cache Size text box.

   NOTE: The minimum cache size setting is 1.5 GB. Additionally, the cache size cannot exceed 50 percent of the installed RAM.
7. Click **OK**.

   NOTE: You must restart the AppAssure Core service for the changes to take effect.

Configuring Replay engine settings

You can configure information regarding the Replay engine, which is the communication channel for AppAssure. These settings determine Core settings to provide effective communication.

In general, Dell recommends using default settings. In some cases, you may be directed by Dell Support to modify these settings to help adjust the performance specific to your network needs.

Complete the steps in this procedure to configure Replay engine settings.

To configure Replay engine settings

1 Navigate to the AppAssure Core Console and click the Configuration tab, and then **Settings**.

2 In the Replay Engine Configuration pane, click **Change**.

The Replay Engine Configuration dialog box displays.

3 Enter the configuration information as described in the following table.

Table 40. Replay engine settings

Text Box	Description
IP Address	Specify the IP address by choosing one of the following: • Click **Automatically Determined** to use the preferred IP address from your TCP/IP. • Or, click **Use a specific address** to manually enter an IP address.
Preferable Port	Enter a port number or accept the default setting. The default port is *8007*. The port is used to specify the communication channel for the Replay engine.
Port in use	Represents the port that is in use for the Replay engine configuration.
Allow port auto-assigning	Click for allow for automatic TCP port assignment.
Admin Group	Enter a new name for the administration group. The default name is *BUILTIN\Administrators*.
Minimum Async I/O Length	Enter a value or choose the default setting. It describes the minimum asynchronous input/output length. The default setting is *65536*.
Receive Buffer Size	Enter an inbound buffer size or accept the default setting. The default setting is *8192*.
Send Buffer Size	Enter an outbound buffer size or accept the default setting. The default setting is *8192*.
Read Timeout	Enter a read timeout value or choose the default setting. The default setting is *00:05:00*.
Write Timeout	Enter a write timeout value or choose the default setting. The default setting is *00:05:00*.
No Delay	It is recommended that you leave this check box unchecked as doing otherwise will impact network efficiency. If you determine that you need to modify this setting, contact Dell Support for guidance in doing so.

4 Click **OK**.

The Replay Engine Configuration dialog box closes, and your new job settings are applied.

Configuring deployment settings

AppAssure lets you download installers from the AppAssure Core to machines you want to protect.

You can configure settings related to the deployment of the AppAssure Agent software from your Core to the machines you want to protect.

Complete the steps in this procedure to configure deployment settings.

To configure deployment settings

1 Navigate to the AppAssure Core Console and click the Configuration tab, and then **Settings**.

2 In the Deploy Settings pane, click **Change**.

The Deploy Settings dialog box appears.

3 Enter deployment settings as described in the following table.

Table 41. Deployment settings

Text Box	Description
Agent installer name	Specify the name of the Core Web Installer executable file. This file streams a download of the latest version of the AppAssure Core installer, which runs directly from the Web and lets you pause and resume the process as needed. The default filename is Agent-Web.exe.
Core address	Enter the address of your Core server. This typically consists of the protocol, the name of your core server and port, and the directory where the Core files reside. For example, if your server is Sample, this setting is https://sample:8006/apprecovery/admin/Core
Failed receive timeout	The amount of time deployment of the Agent software should be attempted before timing out. The default setting is 20 minutes.
Maximum parallel install	This setting controls the maximum number of deployments of the Agent software for the Core to attempt at one time. The default setting is 100.
Automatic reboot after install	This checkbox controls whether machines receiving the AppAssure Agent software automatically reboot after installation. Restarting after installation is required to protect the machine using the newly installed software. The default setting is for this option to be enabled.
Protect after deploy	This checkbox controls whether machines should begin transmitting backup information to the AppAssure Core immediately after deploying and restarting. If disabled, machines will not be protected until the user forces a backup or until the next scheduled backup time.

4 Click **Save**.

The Deploy Settings dialog box closes, and your new deployment settings are applied.

Configuring database connection settings

You can view system events related to the AppAssure Core on the Events tab. The AppAssure Core stores this transactional information in a MongoDB service database. By default, this database is installed locally on the Core machine.

NOTE: For more information about viewing event information from the AppAssure Core, see Viewing tasks, alerts, and events.

Customers can choose to specify installation of the MongoDB service database on another machine accessible on the network to the AppAssure Core. If the service database for your AppAssure Core is installed on a machine other than the machine hosting the AppAssure Core, you must provide database credentials (a user name and password) in these settings.

Complete the steps in this procedure to modify the database connection settings for the service database used by the AppAssure Core.

To modify database connection settings

1 Navigate to the AppAssure Core Console and click the Configuration tab, and then **Settings**.

2 In the Database Connection Settings pane, perform one of the following:

- Click **Restore Default**.
- Or, click **Change**.

The Database Connection Settings dialog box displays.

3 Enter the settings for modifying the database connection as described in the following table.

Table 42. Database connection settings

Text Box	Description
Host name	Enter a host name for the database connection. **NOTE:** When localhost is the parameter specified as the host, the MongoDB is installed locally on the machine hosting the Core.
Port	Enter a port number for the database connection. **NOTE:** The default setting is 27017.
User name	Enter the name of a user with administrative privileges to the MongoDB service database. **NOTE:** If the host name parameter is localhost, this field is not required.
Password	Enter the password associated with the user name you specified. **NOTE:** If the host name parameter is localhost, this field is not required.
Retain event and job history for ___ days	Enter the number of days to retain the event and job history in the service database.
Maximum connection pool size	Sets the maximum number of database connections cached to allow dynamic reuse. **NOTE:** The default setting is 100.
Minimum connection pool size	Sets the minimum number of database connections cached to allow dynamic reuse. **NOTE:** The default setting is 0.

4 Click **Test Connection** to verify your settings.

5 Click **Save**.

Managing licenses

AppAssure lets you manage licenses directly from the AppAssure Core Console. You can change the license key or file associated with your Core, and contact the license server. You can also access the Dell AppAssure License Portal directly from the Licensing page in the console.

The Licensing page includes the following information:

License details:

- **License type.** Types of licenses include Trial, Subscription, or Enterprise. For more information, see the topic About AppAssure Software License Types in the *Dell AppAssure License Portal User Guide*.
- **License status.** Indicates the status of the license. An active status ensures snapshots can continue as scheduled. If the license is blocked, or expired, or if the Core has not been able to communicate with the Dell AppAssure License Portal past the grace period, snapshots are paused until the license status is corrected.

License pool.

- **License pool size.** The license pool is the number of non-trial licenses available to allocate across groups and subgroups in the Dell AppAssure License Portal. The size of the pool determines how many licenses can be allocated. For more information, see the topic About License Pools in the *Dell AppAssure License Portal User Guide*.
- **Protected by this Core.** Indicates the number of machines from the license pool that are protected by this core.
- **Total protected in group.** Indicates the total number of machines protected within the same license group as this Core.

License server. These settings apply to standard (phone home) licenses. These settings are not applicable for appliances and other non-phone-home licenses:

- **License server address.** Displays an active URL for the license server associated with this Core.
- **Last response from the licensing server.** Indicates whether the last attempted communication with the license server portal was successful.
- **Last contact with licensing server.** Displays the date and time of the last successful contact with the licensing server.
- **Next attempt to contact the licensing server**. Indicates the next scheduled date and time to attempt communication with the licensing server.

For more information, see the *Dell AppAssure License Portal User Guide*.

You can also view licensing information for a single protected machine. For more information, see Viewing license information on a machine.

Changing a license

Many users start with a trial license, which has limited capabilities. A trial license is valid for 14 days, and can be extended one time by the group administrator to a 28-day license. Once the trial period expires, the AppAssure Core stops taking snapshots until you obtain a valid license.

Once you have upgraded or purchased your AppAssure license, you will receive either a license file or a license key.

Complete the steps in this procedure to upgrade your trial license, or to change your license from within the AppAssure Core Console.

NOTE: For information about obtaining a license key, see the *Dell AppAssure License Portal User Guide*.

To change a license key or file

1 Navigate to the AppAssure Core Console and then select the Configuration tab.

2 Click **Licensing**.

The Licensing page appears.

3 From the License Details pane, click **Change License**.

The Change License dialog box appears.

4 In the Change License dialog box, to enter a license key, skip to Step 6.

5 To upload a license file, do the following:

a Select **Upload License File** and click **Browse**.

The File Upload dialog box appears.

b Navigate to the location where your license file is stored, and click to select the file.

c In the File Upload dialog box, click **Open**.

The license file is authenticated. Skip to Step 7.

6 In the Change License dialog box, select **Enter License Key**, and then, in the license key text field, enter the license key.

7 Click **Continue**.

The Change License dialog box closes.

8 In the License Server pane, click **Contact Now**.

Once the license is applied to the license server, any associated protected machines will automatically update with the new license.

Contacting the Dell AppAssure License Portal server

The AppAssure Core Console frequently contacts the portal server to remain current with any changes made in the Dell AppAssure License Portal.

For non-trial licenses, the AppAssure Core contacts the license portal every 3 days. If the Core cannot reach the license portal after that grace period, the Core stops taking snapshots.

Typically, communication with the portal server occurs automatically at designated intervals; however, you can initiate communication on-demand.

Complete the steps in this procedure to contact the portal server.

To contact the portal server

1 Navigate to the AppAssure Core Console and then click the Configuration tab.

2 Click **Licensing**.

The Licensing page appears.

3 From the License Server option, click **Contact Now**.

Backing up and restoring Core settings

You can back up core setting information to a file, and later restore these settings if you have problems with the Core machine or if you want to migrate those settings to a different machine. Information that gets backed up includes repository metadata (such as the repository name, data path, and metadata path); machines protected in the Core; replication relationships (targets and sources); which machines are configured for virtual standby; and information about encryption keys.

CAUTION: **You must first back up core setting information before you can use this process to restore core settings.**

Use this procedure to backup and restore Core settings.

To back up and restore Core settings

1 Navigate to the AppAssure Core Console.

2 Click the Configuration tab, and then click Backup/Restore.

The Backup Core Configuration page appears.

3 If you want to back up core settings, proceed to Step 4. If you want to restore settings, proceed to Step 5.

4 To back up the current settings in an XML file, in the Local path text box, type a directory path accessible locally to the Core machine where you want to store core settings as an XML file, and then click Backup.

For example, type C:\Users\Your_User_Name\Documents\AA5CoreSettings and then click Backup.

A file named AppRecoveryCoreConfigurationBackup.xml is saved to the local destination you specified.

5 To restore Core settings from a backup XML file saved previously using this method, perform the following steps.

CAUTION: **When you restore the Core configuration settings, the AppAssure Core service restarts.**

a From the Configuration tab, select Backup/Restore and then click Restore.

The Restore Core Configuration page appears.

b In the local path text box, enter the local path of the location where you stored the core configuration settings.

For example, type C:\Users\Your_User_Name\Documents\AA5CoreSettings.

c If you do not want to restore repository information, proceed to Step f**.**

d Optionally, if you want to restore repository information as configured in the backup file, select Restore Repositories and then click **Restore**.

The page refreshes.

If you choose to restore repository information from the backed-up configuration data, then any repositories configured when the Core settings were saved appear for verification. By default, each repository is selected.

e Verify the repository information you want to restore. If multiple repositories appear in the list for verification, and you only wish to restore information for some of them, then clear the selection for each repository you do not want.

f Click **Restore**.

The restore process begins. An alert appears indicating that the repository service configuration has changed.

g If any configuration settings could not be restored you will see an error message. Review the details of the error to see if any action is required on your part. For more information, see **Viewing tasks, alerts, and events**. To continue, click **Close** to clear the error dialog box.

h After restoring the configuration, follow guidance in the user interface. For example:

- Unlock all encryption keys
- Verify virtual standby credentials. If you have continual export established to a virtual machine, you must specify your credentials before a successful synchronization.
- If replication is set up and you want to restore to a target core, verify the target core settings on the source core.

Configuring Core job settings

Core jobs are automatically created whenever you initiate operations such as replication. The Job Settings section of the Configuration tab lets you determine the settings for each job, including how many jobs should occur at once and how many times a job should be tried, in case a network or other communication error prevents the job from succeeding the first time.

- AppAssure can perform the following Core jobs:
- AdHocDeleteRecoveryPointsJob
- AutoUpdateJob
- BackupJob
- BootCdBulderJob
- CheckAgentRecoveryPointsJob
- CheckRepositoryIntegrityJob
- DeleteAllRecoveryPointsJob
- DeleteOrphanVolumeImagesChainJob
- DeleteRecoveryPointsChainJob
- DownloadExchangeDllsJob
- ExportJob
- ImportJob
- LocalMountRecoveryPointJob
- MaintainRepositoryJob
- NightlyAttachabilityJob
- PushInstallJob
- RecoveryPointAttachabilityJob
- RollbackJob
- RollupJob
- UploadCoreLogsJob

Editing Core job settings

Complete the steps in the following procedure to edit the settings of a job.

To edit Core job settings

1. On the AppAssure Core Console, do one of the following:
   - Click the Configuration tab, and then click **Job Settings**.
   - Click the drop-down menu on the Configuration tab, and then click **Job Settings**.
   - From the icon bar, click the drop-down menu next to the Configuration icon and then click Jobs settings.
2. On the Job Settings page, select a job you want to change. From the drop-down configuration menu for that job, select **Edit**.
3. The Job Settings: [JobName] dialog box opens.
4. To change the maximum number of jobs for the Core to attempt at one time, in the Maximum concurrent jobs text box, enter a new value between 1 to 50.
5. To change the setting for the number of attempts the Core should make before abandoning the job, in the Try count text box, enter a new value between 1 and 10.
6. Click **Save**.

   The Job Settings dialog box closes, and your new job settings are applied.

Adding Core jobs

Complete the steps in the following procedure to add a job to the Core.

To add Core jobs

1. On the AppAssure Core Console, do one of the following:
   - Click the Configuration tab, and then click **Job Settings**.
   - Click the drop-down menu on the Configuration tab, and then click **Job Settings**.
   - From the icon bar, click the drop-down menu next to the Configuration icon and then click Jobs settings.
2. On the Job Settings page, click **Add**.

   The Job Settings dialog box appears.
3. In the Job Settings dialog box, from the Jobs field, select the name of a job you want to add.

   These jobs are described in the topic Configuring Core job settings.
4. To set the maximum number of jobs for the Core to attempt at one time, in the Maximum concurrent jobs text box, enter a new value between 1 to 50.
5. To set the number of attempts the Core should make before abandoning the job, in the Try count text box, enter a new value between 1 and 10.
6. Click **Save**.

   The Job Settings dialog box closes, and your new job settings are applied.

Accessing diagnostics for the Core

In AppAssure, diagnostic information is available for you to download and upload log data for the Core. Additionally, you can download and view diagnostic information for individual protected machines. For more information about machine logs, see Accessing protected machine diagnostics. The ability to access logs can be useful when troubleshooting an issue or working with AppAssure Support. To access core logs, see the following procedures:

- Downloading the Core logs
- Viewing the Core logs
- Uploading logs

Downloading the Core logs

If you encounter any errors or issues with the Core, you can download the Core logs to view them or upload them to AppAssure Support.

To download the Core logs

1. Navigate to the AppAssure Core Console.
2. Click the Tools drop-down menu or tab, click **Diagnostics**, and then click **View Log**.
3. On the Download Core Log page of the Tools tab, click **Click here to begin the download**.
4. In the Opening CoreAppRecovery.log dialog box, select **Save File**.
5. Click **OK**.

   The CoreAppRecovery.log file saves to your Downloads folder.

Viewing the Core logs

If you encounter any errors or issues with the Core, it may be useful to view the logs.

To view the Core logs

1. Navigate to the AppAssure Core Console.
2. Click the Tools drop-down menu or tab, click **Diagnostics,** and then click **View Log**.
3. On the Download Core Log page of the Tools tab, click on the link **Click here to begin the download**.
4. In the Opening CoreAppRecovery.log dialog box, select **Open with**, and then use the drop-down list to select a program with which you want to open the file; for example, Notepad.
5. Click **OK**.

   The file opens in the program you selected.

Uploading logs

The following procedure lets you upload logs that you may have downloaded for the Core or for protected machines. Complete the step in the following procedure to upload machine logs.

To upload logs

1. Navigate to the AppAssure Core Console.
2. Click the Tools tab, and then click **Diagnostics**.

3 Click **Upload Log**.

4 On the Upload Core Log page of the Tools tab, click **Click here to begin the upload**.

The Events tab displays for you to view the progress of the upload of log information for the Core and all protected machines.

5

Understanding encryption keys

This chapter describes the process of securing data in your environment using encryption keys. It includes the following sections:

- Applying or removing encryption from a protected machine
- Managing encryption keys

The AppAssure Core can encrypt snapshot data for all volumes within any repository using encryption keys that you define and manage.

Instead of encrypting the entire repository, AppAssure lets you specify an encryption key for one or more machines protected on a single AppAssure Core. Each active encryption key creates an encryption domain. There is no limit to the number of encryption keys you can create on the Core. In a multi-tenant environment (when a single Core hosts multiple encryption domains), data is partitioned and deduplicated within each encryption domain. As a result, Dell recommends using a single encryption key for multiple protected machines if you want to maximize the benefits of deduplication among a set of protected machines.

Because you manage the encryption keys, loss of the volume cannot leak the keys.

You can also share encryption keys between Cores using one of three methods. One method is to export an encryption key as a file from one AppAssure Core and import it to another Core. A second method is to archive data secured with an encryption key, and then import that archived data into another AppAssure Core. The third method is to replicate recovery points from a protected machine using an encryption key. After you replicate protected machines, the encryption keys used in the source Core appear as replicated encryption keys in the target Core.

In all cases, once imported, any encryption key appears in the Core with a state of Locked. To access data from a locked encryption key, you must unlock it. For information about importing, exporting, locking or unlocking encryption keys, see the topic Managing encryption keys.

Key security concepts and considerations include:

- Encryption is performed using 256 Bit Advanced Encryption Standard (AES) in Cipher Block Chaining (CBC) mode that is compliant with SHA-3.
- Deduplication operates within an encryption domain to ensure privacy.
- Encryption is performed without impact on performance.
- You can apply a single encryption key to any number of protected machines, but any protected machine can only have one encryption key applied at a time.
- You can add, remove, import, export, modify, and delete encryption keys that are configured on the AppAssure Core.

△ **CAUTION: AppAssure takes a new base image whenever you apply an encryption key to a protected machine, or when you edit the name, description, or passphrase for an encryption key that is applied to one or more protected machines. A new base image is also triggered after you disassociate an encryption key for a protected machine. The resulting base image is captured upon the next scheduled snapshot (or immediately if you force a snapshot).**

Encryption keys generated from the AppAssure Core are text files that contain four parameters, as described in the following table:

Table 43. Components of an encryption key

Component	Description
Name	This value is equivalent to the key name given when adding a key in the AppAssure Core Console.
Key	This parameter consists of 107 randomly generated English alphabetic, numeric, and mathematical operator characters.
ID	The key ID consists of 26 randomly generated upper-case and lower-case English characters.
Comment	The comment contains the text of the key description entered when the key was created.

Applying or removing encryption from a protected machine

You can secure the data protected on your Core at any time by defining an encryption key and applying it to one or more protected machines in your repository. You can apply a single encryption key to any number of protected machines, but any protected machine can only use one encryption key at a time.

The scope of deduplication in AppAssure is limited to protected machines using the same repository and encryption key. Therefore, to maximize the value of deduplication, Dell recommends applying a single encryption key to as many protected machines as is practical. However, there is no limit to the number of encryption keys you can create on the Core. Thus, if legal compliance, security rules, privacy policies, or other circumstances require it, you can add and manage any number of encryption keys. You could then apply each key to only one protected machine, or any set of machines in your repository.

Any time you apply an encryption key to a protected machine, or modify the properties of an encryption key that is in use for one or more protected machines (including removing encryption), AppAssure takes a new base image for that machine upon the next scheduled or forced snapshot. The data stored in that base image (and all subsequent incremental snapshots taken while an encryption key is applied) is protected by a 256-bit advanced encryption standard. There are no known methods for compromising this method of encryption.

Once an encryption key is created and applied to a protected machine, there are two concepts involved in removing that encryption. The first is to disassociate the key from the protected machine. Optionally, once the encryption key is disassociated from all protected machines, it can be deleted from the AppAssure Core.

This section includes the following topics:

- Associating an encryption key with a protected machine
- Applying an encryption key from the Machines tab
- Disassociating an encryption key from a protected machine

Associating an encryption key with a protected machine

You can apply an encryption key to a protected machine using either of two methods:

- **As part of protecting a machine.** When using this method, you can apply encryption to one or multiple machines simultaneously. This method lets you add a new encryption key, or apply an existing key to the selected machine or machines.

  To use encryption when first defining protection for a machine, you must select the advanced options in the relevant Protect Machines Wizard. This selection adds an Encryption page to the wizard workflow. From this page, select **Enable encryption**, and then select an existing encryption key or specify parameters for a new key. For more information, see Protecting a machine or Protecting multiple machines, respectively.

- **By modifying the configuration settings for a machine.** This method applies an encryption key to one protected machine at a time. There are two approaches for modifying configuration settings for a machine in the AppAssure UI:
  - Modify the Core settings in the Configuration tab for a specific protected machine. The encryption key you want to use for this approach must already exist on the AppAssure Core. For more information, see Viewing and modifying configuration settings.
  - Click the encryption icon on the Machines tab. Using this approach you can create and apply a new encryption key, or assign an existing key to the specified protected machine. For more information, see Applying an encryption key from the Machines tab.

Applying an encryption key from the Machines tab

Once an encryption key has been added to an AppAssure Core, it can be used for any number of protected machines.

If you select an encryption key during the initial protection of one or more machines, that key is automatically applied to any machines you protect using that wizard. In such cases, this procedure is not required.

Perform this procedure if you added an encryption key using the process described in the topic Adding an encryption key.

△ **CAUTION: After you apply an encryption key to a protected machine, AppAssure takes a new base image for that machine upon the next scheduled or forced snapshot.**

To apply an encryption key

1 Navigate to the AppAssure Core and click **Protected Machines**.

The Machines tab appears, listing all the machines protected by this Core. An open lock appears for any machine that does not have an encryption key applied. A closed lock indicates that a protected machine has encryption applied.

2 In the Protected Machines pane, click the lock icon for the protected machine you want to configure.

The Encryption Configuration dialog box appears.

3 Do one of the following:
  - If you want to apply an existing encryption key to this machine, from the Select Encryption Key drop-down menu, select the appropriate key.
  - If you want to create a new encryption key and apply it to this protected machine, click **Add New Encryption Key**. Then enter the details for the key as described in the following table.

Table 44. New encryption key details

Text Box	Description
Name	Enter a name for the encryption key. Encryption key names must contain between 1 and 130 alphanumeric characters. Do not use prohibited characters or prohibited phrases.
Description	Enter a comment for the encryption key. This information appears in the Description field when viewing encryption keys from the Configuration tab of the AppAssure Core Console. Descriptions may contain up to 454 characters. Best practice is to avoid using prohibited characters and prohibited phrases.

Table 44. New encryption key details

Text Box	Description
Passphrase	Enter a passphrase used to control access. Best practice is to avoid using prohibited characters. Record the passphrase in a secure location. Dell Support cannot recover a passphrase. Once you create an encryption key and apply it to one or more protected machines, you cannot recover data if you lose the passphrase.
Confirm Passphrase	Re-enter the passphrase. It is used to confirm the passphrase entry.

4 Click **OK**.

The dialog box closes. The encryption key you specified has been applied to future backups for this protected machine, and the lock now appears as closed.

Optionally, if you want the encryption key applied immediately, force a snapshot. For more information, see Forcing a snapshot.

△ **CAUTION: AppAssure uses AES 256-bit encryption in the Cipher Block Chaining (CBC) mode with 256-bit keys. While using encryption is optional, Dell recommends that you establish an encryption key, and that you protect the passphrase you define. Store the passphrase in a secure location as it is critical for data recovery. Without a passphrase, data recovery is not possible.**

Disassociating an encryption key from a protected machine

Once an encryption key is applied to a protected machine, all subsequent snapshot data stored in the AppAssure Core is encrypted.

You can disassociate an encryption key from a protected machine. This action does not decrypt the existing backup data, but does result in a new base image for that machine at the time of the next scheduled or forced snapshot.

ⓘ **NOTE:** If you want to remove an encryption key from the Core, as described in the topic Removing an encryption key, you must first disassociate that encryption key from all protected machines.

Perform this procedure to disassociate an encryption key from a specific protected machine.

To disassociate an encryption key from a protected machine

1 Navigate to the AppAssure Core and click **Protected Machines**.

The Machines tab appears, listing all the machines protected by this Core. An open lock appears for any machine that does not have an encryption key applied. A closed lock indicates that a protected machine has encryption applied.

2 In the Protected Machines pane, click the closed lock icon for the protected machine you want to configure.

The Encryption Configuration dialog box appears.

3 From the Select Encryption Key drop-down menu, select **(None)** and then click **OK**.

4 If you want to remove this encryption key from AppAssure Core, first repeat this procedure for all protected machines using this key. Then perform the procedure described in the topic Removing an encryption key.

Managing encryption keys

To manage encryption keys for the AppAssure Core, from the Configuration tab, click **Security**. The Encryption Keys pane appears. For each encryption key added to your AppAssure Core (if any have been defined yet), you see the information described in the following table.

Table 45. Information about each encryption key

UI Element	Description
Name	The name associated with the encryption key.
Thumbprint	This parameter is a 26-character alphabetic string of randomly generated English upper and lower case letters that helps uniquely identify each encryption key.
Status	Status describes the origin point of an encryption key and its ability to be applied. An encryption key can contain one of two possible status conditions: **Universal**. Universal status is the default condition when you create an encryption key. A key with a status of Universal, combined with a state of Locked, indicates that the key can be applied to a protected machine. You cannot manually lock a key with a status of Universal; instead, you must first change its status as described in the procedure Changing encryption key status. **Replication.** When a protected machine in a source Core has encryption enabled, and recovery points for that machine are replicated in a target Core, any encryption keys used in the source appear automatically in the target Core with a status of Replication. The default state after receiving a replicated key is *locked*. You can unlock an encryption key with a status of Replication by providing the passphrase. If a key has a status of Unlocked, you can manually lock it. For more information, see the topic Locking or unlocking an encryption key.
State	The state indicates whether an encryption key can be used. Two possible states include: **Unlocked.** An Unlocked state indicates that the key can be used immediately. For example, you can encrypt snapshots for a protected machine, or perform data recovery from a replicated recovery point on the target Core. **Locked.** A Locked state indicates that the key cannot be used until it is unlocked by providing the passphrase. Locked is the default state for a newly imported or replicated encryption key. If the state of an encryption key is Locked, it must be unlocked before it can be used. If you previously unlocked a locked encryption key, and the duration to remain unlocked has expired, the state changes from Unlocked to Locked. After the key locks automatically, you must unlock the key again in order to use it. For more information, see the topic Locking or unlocking an encryption key.
Description	The description is an optional field that is recommended to provide useful information about the encryption key such as its intended use or a passphrase hint.

At the top level of the Encryption Keys pane, you can add an encryption key or import a key using a file exported from another AppAssure Core.

Once an encryption key exists for a Core, you can manage the existing keys by editing the name or description properties; changing the passphrase; unlocking a locked encryption key; or removing the key from the AppAssure Core. You can also export a key to a file, which can be imported into another AppAssure Core.

When you add an encryption key from the Configuration tab, it appears in the list of encryption keys, but is not applied to a specific protected machine. For information on how to apply an encryption key you create from the Encryption Keys pane, or to delete a key entirely from the AppAssure Core, see Applying or removing encryption from a protected machine.

From the Encryption Keys pane, you can manage security for the backup data saved to the Core for any protected machine in your repository by doing the following:

- Adding an encryption key
- Importing an encryption key
- Locking or unlocking an encryption key
- Editing an encryption key
- Changing an encryption key passphrase
- Exporting an encryption key
- Removing an encryption key
- Changing encryption key status

Adding an encryption key

After an encryption key is defined, you can use it to safeguard your data. Encryption keys can be used by any number of protected machines.

This step describes how to add an encryption key from the AppAssure Core Console. This process does not apply the key to any machines currently being protected on the Core. You can also add an encryption key during the process of protecting a machine. For more information on adding encryption as part of protecting one machine, see Protecting a machine. For more information on adding encryption to two or more machines while initially protecting them, see Protecting multiple machines.

Complete the steps in this procedure to add an encryption key.

To add an encryption key

1 Navigate to the AppAssure Core, click the Configuration tab, and then select **Security**.

The Encryption Keys page appears.

2 From the Actions drop-down menu, select **Add Encryption Key**.

The Create Encryption Key dialog box appears.

3 In the Create Encryption Key dialog box, enter the details for the key as described in the following table.

Table 46. Create encryption key details.

Text Box	Description
Name	Enter a name for the encryption key. Encryption key names must contain between 1 and 130 alphanumeric characters. Do not use prohibited characters or prohibited phrases.
Description	Enter a comment for the encryption key. This information appears in the Description field when viewing encryption keys from the Core Console. You can enter up to 254 characters. Best practice is to avoid using prohibited characters and prohibited phrases.
Passphrase	Enter a passphrase used to control access. Best practice is to avoid using prohibited characters. Record the passphrase in a secure location. Dell Support cannot recover a passphrase. Once you create an encryption key and apply it to one or more protected machines, you cannot recover data if you lose the passphrase.
Confirm Passphrase	Re-enter the passphrase. It is used to confirm the passphrase entry.

4 Click **OK**.

The dialog box closes and the encryption key you created is visible on the Encryption Keys page.

5 If you want to apply the encryption key to a protected machine, see Applying an encryption key from the Machines tab.

CAUTION: AppAssure uses AES 256-bit encryption in the Cipher Block Chaining (CBC) mode with 256-bit keys. While using encryption is optional, Dell recommends that you establish an encryption key, and that you protect the passphrase you define. Store the passphrase in a secure location as it is critical for data recovery. Without a passphrase, data recovery is not possible.

Importing an encryption key

You can import an encryption key from another AppAssure Core and use that key to encrypt data for a protected machine in your Core. To import the key, you must be able to access it from the Core machine, either locally or through your network.

Complete the steps in this procedure to import an encryption key.

NOTE: This procedure does not apply the key to any protected machines. For more information on applying the key, see Applying an encryption key from the Machines tab.

To import an encryption key

1 Navigate to the AppAssure Core, click the Configuration tab, and then select **Security**.

The Encryption Keys page appears.

2 From the Actions drop-down menu, select **Import**.

The Import Key dialog box appears.

3 In the Import Key dialog box, click **Browse** to locate the encryption key you want to import.

The key filename starts with "EncryptionKey-," followed by the key ID, and ending in the file extension .key. For example, a sample encryption key name is **EncryptionKey-RandomAlphabeticCharacters.key**.

4 Select the key you want to import, and then click **Open**.

5 In the Import Key dialog box, click **OK**.

The dialog box closes and the encryption key you imported is visible on the Encryption Keys page. If the encryption key was used to protect a volume before it was exported, the state of the key is Locked.

Locking or unlocking an encryption key

Encryption keys may contain a state of unlocked or locked. An unlocked encryption key can be applied to a protected machine to secure the backup data saved for that machine in the repository. From an AppAssure Core using an unlocked encryption key, you can also recover data from a recovery point.

When you import an encryption key into an AppAssure Core, its default state is Locked. This is true regardless of whether you explicitly imported the key, or whether the encryption key was added to the AppAssure Core either by replicating encrypted protected machines or by importing an archive of encrypted recovery points.

For encryption keys added to the AppAssure Core by replication only, when you unlock a key you can specify a duration of time (in hours, days, or months) for the encryption key to remain unlocked. Each day is based on a 24-hour period, starting from the time the unlock request is saved to the AppAssure Core. For example, if the key is unlocked at 11:24 AM on Tuesday and the duration selected is 2 days, the key automatically re-locks at 11:24 AM that Thursday.

CAUTION: You cannot use a locked encryption key to recover data or to apply to a protected machine. You must first provide the passphrase, thus unlocking the key.

You can also lock an unlocked encryption key, ensuring that it cannot be applied to any protected machine until it is unlocked. To lock an encryption key with a state of Universal, you must first change its status to Replicated.

If an unlocked encryption key is currently being used to protect a machine in the Core, you must first disassociate that encryption key from the protected machine before you can lock it.

Complete the steps in this procedure to unlock a locked encryption key.

To unlock an encryption key

1 Navigate to the AppAssure Core, click the Configuration tab, and then select **Security**.

The Encryption Keys page appears. The State column indicates which encryption keys are locked.

2 From the Configuration drop-down menu for the encryption key that you want to unlock, select **Unlock**.

The Unlock Key dialog box appears.

3 In the Unlock Key dialog box, in the Passphrase field, enter the passphrase to unlock this key.

4 To specify the length of time that the key remains unlocked, in the Duration option, do one of the following:

- To specify that the key remains unlocked until you explicitly lock it, AppAssure select **Until explicitly forgotten**.

  This option is available for unlocking any encryption key.

- To specify that the key remains locked for a duration which you configure:
  - Select the number field and, by typing or using the up and down arrow controls, specify an integer.
  - In the duration field, select hours, days, or months, respectively.
  - Then click **OK**.

  This option is available for encryption keys added by replication.

The dialog box closes and the changes for the selected encryption key are visible on the Encryption Keys page.

Complete the steps in this procedure to lock an encryption key.

To lock an unlocked encryption key

1 Navigate to the AppAssure Core, click the Configuration tab, and then select **Security**.

The Encryption Keys page appears. The State column indicates which encryption keys are unlocked, and shows the status for each key.

2 If the status of the encryption key that you want to unlock is Universal, then from the Configuration drop-down menu, select **Change the encryption status to Replicated**.

The Change Encryption Key Status dialog box appears.

3 In the Change Encryption Key Status dialog box, confirm that you want to change the status of the key to Replicated.

4 If you successfully changed the encryption key status to Replicated, then from the Configuration drop-down menu for the encryption key that you want to lock, select **Lock**.

The Lock Key dialog box appears.

5 In the Lock Key dialog box, confirm that you want to lock the key.

The dialog box closes, and the state of the selected encryption key is now locked.

NOTE: This option is available for encryption keys added by replication.

Editing an encryption key

After an encryption key is defined, you can edit the name of the encryption key or the description of the key. These properties are visible when you view the list of encryption keys in the Encryption Keys pane.

Complete the steps in this procedure to edit the name or description of an existing encryption key.

> △ **CAUTION: After you edit the name or description an encryption key that is used to protect one or more machines, AppAssure takes a new base image.That base image snapshot occurs for that machine upon the next scheduled or forced snapshot.**

To edit an encryption key

1. Navigate to the AppAssure Core, click the Configuration tab, and then select **Security**.

   The Encryption Keys page appears.

2. From the Configuration drop-down menu for the encryption key that you want to modify, select **Edit**.

   The Edit Encryption Key dialog box appears.

3. In the Edit Encryption Key dialog box, edit the name or the description for the encryption key, and then click **OK**.

   The dialog box closes and the changes for the selected encryption key are visible on the Encryption Keys page.

Changing an encryption key passphrase

To maintain maximum security, you can change the passphrase for any existing encryption key. Complete the steps in this procedure to change the passphrase for an encryption key.

> △ **CAUTION: After you edit the passphrase for an encryption key that is used to protect one or more machines, a new base image is taken for that machine upon the next scheduled or forced snapshot.**

To change an encryption key passphrase

1. Navigate to the AppAssure Core, click the Configuration tab, and then select **Security**.

   The Encryption Keys page appears.

2. From the Configuration drop-down menu for the encryption key that you want to modify, select **Change Passphrase**.

   The Change Passphrase dialog box appears.

3. In the Change Passphrase dialog box, enter the new passphrase for the encryption and then re-enter the passphrase to confirm what you entered.

4. Click **OK**.

   The dialog box closes and the passphrase is updated.

> △ **CAUTION: AppAssure uses AES 256-bit encryption in the Cipher Block Chaining (CBC) mode with 256-bit keys. It is recommended that you protect the passphrase you define. Store the passphrase in a secure location as it is critical for data recovery. Without a passphrase, data recovery is not possible.**

Exporting an encryption key

You can export an encryption key from any AppAssure Core with the express purpose of using it in another Core. When you perform this procedure, the key is saved to the Downloads folder for the active Windows user account.

Complete the steps in this procedure to export an encryption key.

To export an encryption key

1. Navigate to the AppAssure Core, click the Configuration tab, and then select **Security**.
2. From the Configuration drop-down menu for the encryption key that you want to export, select **Export**.

   The Export Key dialog box appears.
3. In the Export Key dialog box, click **Save File** to save and store the encryption keys in a secure location, and then click **OK**.

Removing an encryption key

When you remove an encryption key on the Configuration tab, the key is deleted from the AppAssure Core.

You cannot remove an encryption key that is already associated with any protected machine. You must first view the encryption settings for each protected machine using the key, and disassociate the encryption key you want to remove. For more information, see the topic Disassociating an encryption key from a protected machine.

Complete the steps in this procedure to remove an encryption key.

To remove an encryption key

1. Navigate to the AppAssure Core, click the Configuration tab, and then select **Security**.
2. From the Configuration drop-down menu for the encryption key that you want to remove, select **Remove**.

   You see a message confirming the action to remove the encryption key.
3. In the Remove Key dialog box, confirm that you want to remove the encryption key.

   NOTE: Removing an encryption key does not make the data un-encrypted.

   The dialog box closes and the encryption key you removed no longer appears on the Encryption Keys page.

Changing encryption key status

Encryption keys list one of two possible status conditions on the Encryption Keys pane: Universal or Replication. The status indicates the likely origin of the encryption key, and determines whether you can change its details or passphrase. You can modify these attributes only if the status is Universal. If you need to modify these attributes for a key with Replicated status, you must change its status to Universal using this procedure. When you change the status of an encryption key to Universal, it is unlocked manually and can be used to encrypt other protected machines.

CAUTION: You must know the passphrase to change the status from Replicated to Universal.

Encryption keys also have two possible states: Locked or Unlocked. The state controls your ability to apply an encryption key to a protected machine, or to restore data from a recovery point with encryption. You can change the status of an encryption key manually only if the state is Unlocked.

When you first create an encryption key, its status is Universal, and its state is Unlocked. You can use such a key immediately (for example, to encrypt backups for a protected machine). However, a key with Universal status

cannot be locked manually. If you want to manually lock an encryption key with a status of Universal, you must change the status to Replicated using this procedure.

Follow this procedure to change the status of an encryption key.

To change encryption key status

1 Navigate to the AppAssure Core, click the Configuration tab, and then select **Security**.

Any encryption keys accessible to the Core appear in the Encryption Keys pane. Each lists a status of universal or replicated.

2 To change the status from Universal, from the Configuration drop-down menu for the encryption key that you want to change, select **Change the status to replicated**.

You see a message confirming the action to change the encryption key status.

3 To change the status from Replicated, from the Configuration drop-down menu for the encryption key that you want to change, select **Change the status to universal**.

You see a message confirming the action to change the encryption key status.

4 Provide the encryption key passphrase and then click OK.

The dialog box closes and the encryption key status is updated on the Encryption Keys page.

6

Protecting machines using the AppAssure Core

This chapter describes how to protect, configure, and manage the protected machines in your AppAssure environment. It includes the following sections:

- Dynamic and basic volumes support limitations
- About the Agent Installer
- Deploying the Agent (push install)
- Understanding bulk deploy
- Understanding protection schedules
- Protecting a machine
- Protecting multiple machines

To protect your data using AppAssure, you need to add the workstations and servers for protection in the AppAssure Core Console; for example, your Exchange server, SQL Server, Linux server, and so on.

In the AppAssure Core Console, you can identify the machine on which the AppAssure Agent software is installed, and specify which volumes, for example, a Microsoft Windows Storage Space, to protect. You can define the schedules for protection, add additional security measures such as encryption, and much more. For more information on how to access the AppAssure Core Console to protect workstations and servers, see Protecting a machine.

△ **CAUTION: AppAssure does not support bare metal restores (BMRs) of Linux machines with EXT2 boot partitions. Any BMR performed on a machine with this type of partition results in a machine that does not start. If you want to be able to perform a BMR on this machine in the future, you must convert any EXT2 partitions to EXT3 or EXT4 before you begin protecting and backing up the machine.**

Dynamic and basic volumes support limitations

AppAssure supports taking snapshots of all dynamic and basic volumes. AppAssure also supports exporting simple dynamic volumes that are on a single physical disk. As their name implies, simple dynamic volumes are not striped, mirrored or spanned volumes.

Dynamic disks (except simple dynamic disks as previously described) are not available for selection in the Export Wizard. Non-simple, dynamic volumes have arbitrary disk geometries that cannot be fully interpreted. AppAssure therefore does not support the export of complex or non-simple dynamic volumes.

Notification appears in the user interface to alert you that exports are limited and restricted to simple dynamic volumes. If you attempt to export anything other than a single dynamic volume, the export job will fail.

About the Agent Installer

AppAssure lets you download installers from the AppAssure Core. From the Tools tab, you can choose to download the Agent Installer or the Local Mount Utility (LMU). For more information about the LMU, see Understanding the Local Mount Utility.

NOTE: For access to the Agent Installer, see Downloading the Agent Installer. For more information about deploying the Agent Installer, see the *Dell AppAssure Installation and Upgrade Guide*.

The Agent installer is used to install the AppAssure Agent application on machines that are intended to be protected by the AppAssure Core. If you determine that you have a machine that requires the Agent Installer, you can download the web installer from the Tools tab in the AppAssure Core.

NOTE: Downloading of the Core is performed from the Dell AppAssure License Portal. For more information or to download the AppAssure Core installer, visit https://licenseportal.com.

Downloading the Agent Installer

You can download and deploy the AppAssure Agent Installer on any machine that will be protected by the AppAssure Core. Complete the steps in this procedure to download the web installer.

To download the AppAssure Agent installer

1. Download the AppAssure Agent installer file from the Dell AppAssure License Portal or from the AppAssure Core. For example:

   `Agent-X64-5.2.1.xxxxx.exe`

2. Click **Save File**.

   For more information about installing agents, see the *Dell AppAssure Installation and Upgrade Guide*.

Deploying the Agent (push install)

AppAssure lets you deploy the AppAssure Agent Installer to individual Windows machines for protection. Complete the steps in the following procedure to push the installer to an agent.

To deploy agents to multiple machines at the same time, see Deploying to multiple machines.

NOTE: Agents must be configured with a security policy that makes remote installation possible.

To deploy the agent

1. In the AppAssure Core Console in the left navigation area, click Protected Machines to open the Machines page.
2. In the **Actions** drop-down menu, click **Deploy Agent**.

   The Deploy Agent dialog box appears.
3. In the Deploy Agent dialog box, enter the logon settings as described in the following table.

   Table 47. Machine logon settings

Text Box	Description
Machine	Enter the host name or IP address of the agent machine that you want to deploy.
User name	Enter the user name to connect to this machine; for example, Administrator or, if the machine is in a domain, [domain name]\Administrator).

Table 47. Machine logon settings

Text Box	Description
Password	Enter the password to connect to this machine
Automatic reboot after install	Select to specify whether the Core should start upon the completion of the deployment and installation of the AppAssure Agent Installer.

4 Click **Verify** to validate the credentials you entered.

The Deploy Agent dialog box displays a message to indicate that validation is being performed. Click **Abort** if you want to cancel the verification process. After the verification process is complete, a message indicating that verification has been completed displays.

5 Click **Deploy**.

A message indicating that the deployment has started displays. You can view the progress in the Events tab. Click **Show details** to view more information about the status of the agent deployment.

6 Click **OK**.

Modifying deploy settings

Complete the steps in this procedure to modify deploy settings.

To modify deploy settings

1 Navigate to the AppAssure Core Console and click the Configuration tab, and then **Settings**.

2 In the Deploy Settings pane, click **Change**.

The Deploy Settings dialog box displays.

3 In the Agent Installer Name text box, enter the name of the agent executable file. The default is Agent-web.exe.

4 In the Core Address text box, enter the address for the core.

5 In the Failed Receive Timeout text box, enter the number of minutes to wait without activity to timeout.

6 In the Max Parallel Installs text box, enter a number for the maximum installations that can be installed in parallel.

7 Select either or both of the following optional settings:

- **Automatic reboot after install**
- **Protect After Deploy**

8 Click **OK**.

Understanding bulk deploy

This topic describes the tasks that administrators perform to deploy the AppAssure Agent software simultaneously to multiple Windows machines. This batch processing capability is known as the bulk deploy feature.

To deploy and protect multiple agents, perform the following tasks:

1 Deploy AppAssure Agent to multiple machines. See Deploying to multiple machines.

2 Monitor the activity of the batch deployment. See Verifying the deployment to multiple machines.

3 Protect multiple machines. See Protecting multiple machines.

> **NOTE:** This step can be skipped if you selected the **Protect Machine After Install** option during deployment.

4 Monitor the activity of the batch protection. See Monitoring the protection of multiple machines.

Deploying to multiple machines

You can simplify the task of deploying the AppAssure Agent software to multiple Windows machines by using the bulk deploy feature of Dell AppAssure. From within the Core Console, you can specifically bulk deploy to:

- Machines on an Active Directory domain
- Machines on a VMware vCenter/ESX(i) virtual host
- Machines on any other host

The bulk deploy feature automatically detects machines on a host and allows you to select those to which you want to deploy. Alternatively, you can manually enter host and machine information.

> **NOTE:** You can use the bulk deploy feature to deploy the Agent software to as many as 50 agent machines. The machines to which you are deploying must have access to the Internet to download and install bits, since the AppAssure Core uses the Web version of the AppAssure Agent Installer to deploy the installation components. If access to the Internet is not available, you will need to manually download the AppAssure Agent Installer from the Dell AppAssure License Portal and deploy it to the individual machines that you want to protect.
>
> For more information, see the *Dell AppAssure License Portal User Guide*.

Deploying to machines on an Active Directory domain

Before starting this procedure, you must have the domain information and logon credentials for the Active Directory server.

To deploy to machines on an Active Directory domain

1 On the AppAssure Core Console, click the Tools tab, and then click **Bulk Deploy**.

2 On the Deploy Agent to Machines pane, click **Active Directory**.

3 In the Connect to Active Directory dialog box, enter the domain information and logon credentials as described in the following table.

Table 48. Domain information and credentials

Text Box	Description
Host	The host name or IP address of the Active Directory domain.
User name	The user name used to connect to the domain; for example, Administrator or, if the machine is in a domain, [domain name]\Administrator).
Password	The secure password used to connect to the domain.

4 Click **Connect**.

5 On the Add Machines from Active Directory dialog box, select the machines to which you want to deploy the AppAssure Agent software, and then click **Add**.

The machines you added appear on the Deploy Agent to Machines pane. The system verifies each machine you added automatically.

6 Optionally, you can verify any machine by selecting the machine and then clicking **Verify** in the toolbar.

7 Check the status icon and message for each machine on the Deploy Agents to Machines page. These reflect the readiness of each machine for deployment, as follows:

- Green icon - AppAssure is able to connect to the machine and it is ready to be deployed.
- Yellow icon - AppAssure is able to connect to the machine; however, the AppAssure Agent on the machine is already paired with an AppAssure Core.
- Red icon - AppAssure cannot connect to the machine. This may be because the logon credentials are incorrect, the machine is shut down, the firewall is blocking traffic, or another problem.

  NOTE: The Agent software will not be deployed to any machine with a red status icon.

- If the status message for each machine indicates that it is ready to deploy, skip to Step 9.
- Optionally, if the status message shows a Details link, click on **Details** to determine if you can correct the issue.

8 To correct issues connecting to any machine showing a red status, or to enable automatic protection, change port or authentication information, change the display name, specify a repository or establish an encryption key, click **Settings** on the toolbar or click the **Edit** link next to the machine, and then do the following:

a In the Edit Settings dialog box, specify the settings as described in the following table.

Table 49. Active Directory settings

Text Box	Description
Host name	Automatically provided from Step 3.
User name	*Automatically provided from* Step 3.
Password	*Enter the password for the machine.*
Automatic reboot after install	Mandatory. Reboots the machine after deployment, which is required before protecting any agent machine.
Protect machine after install	Selected by default. If selected, the system will automatically protect the machine after deployment. (This allows you to skip Protecting multiple machines.)
Display name	Automatically assigned based on the host name provided in Step 3. This is the name that appears in the Core Console for the selected machine.
Port	The port number on which the AppAssure Core communicates with the agent on the machine. The default port is 8006.
Repository	Use the drop-down list to select the repository on the AppAssure Core where the data from the specified machines should be stored. This option is only available when you select **Protect machine after install**.
Encryption key	(Optional) Use the drop-down list to specify whether encryption should be applied to the data on the specified machine. The encryption key is assigned to all machines that are being protected. **NOTE:** This option is only available when you select **Protect machine after install**.

b Click **Save**.

9 After machines are successfully verified, select each machine to which you want to deploy the AppAssure Agent, and then click **Deploy**.

If you proceed with protecting an agent that is already protected by another Core, that protection will be stopped and the machine will now be protected by this Core.

If you chose the **Protect machine after install** option, after deployment is successful, the machines automatically reboot and protection is enabled.

Deploying to machines on a VMware vCenter/ESX(i) virtual host

Before starting this procedure, you must have the host location information and logon credentials for the VMware vCenter/ESX(i) virtual host.

NOTE: All virtual machines must have VM Tools installed; otherwise, AppAssure cannot detect the host name of the virtual machine to which to deploy. In lieu of the host name, AppAssure uses the virtual machine name, which may cause issues if the host name is different from the virtual machine name.

To deploy to machines on a vCenter/ESX(i) virtual host

1 Navigate to the AppAssure Core Console, click the Tools tab, and then click **Bulk Deploy**.

2 In the Deploy Agent to Machines window, click **vCenter/ESX(i)**.

3 In the Connect to vCenter Server/ESX(i) dialog box, enter the host information and logon credentials as follows and click **Connect**.

Table 50. vCenter/ESX(i) connection settings

Text Box	Description
Host	The name or IP address of the VMware vCenter Server/ESX(i) virtual host.
Port	The port used to connect to the virtual host. The default setting is 443.
User name	The user name used to connect to the virtual host; for example, Administrator or, if the machine is in a domain, [domain name]\Administrator).
Password	The secure password used to connect to this virtual host.

4 In the Add Machines from VMware vCenter Server/ESX(i) dialog box, or the following:

a Under Options, enter the user name and password for the virtual machines you select.

b Drill through the machines, expanding or collapsing through the clusters, resource pools, vApps and virtual machines by clicking the arrows next to each node.

c Select the machines to which you want to deploy and then click **Add**.

The machines you added appear on the Deploy Agent to Machines window. The system verifies each machine you added automatically.

5 Optionally, you can verify any machine, check the status of the machine, or edit connection settings for a machine, as described in Step 6 to Step 8 of the procedure Deploying to machines on an Active Directory domain.

6 After machines are verified successfully, check the box next to each machine to which you want to deploy the AppAssure Agent, and then click **Deploy**.

If you proceed with protecting an agent that is already protected by another Core, that protection will be stopped and the machine will now be protected by this Core.

If you chose the **Protect machine after install** option, after deployment is successful the machines are rebooted automatically and protection is enabled.

Deploying to machines on any other host

Complete the following procedure to deploy the AppAssure Agent to multiple machines on any other type of host.

To deploy to machines on any other host

1 Navigate to the AppAssure Core Console, click the Tools tab, and then click **Bulk Deploy**.

2 On the Deploy Agent on Machines window, do one of the following:

- Click **New** to enter a new machine host, logon credentials, optional display name, repository, encryption key, and other information. For details on each setting, see Deploying to machines on an Active Directory domain.

  After you enter this information, click **OK** to add it to the Deploy Agent on Machines list, or click **OK & New** to add another machine.

  > **NOTE:** If you want to automatically protect the machine after deployment, check the **Protect Machine after Install** box. If you check the box, the machine will be rebooted automatically prior to enabling protection.

- To specify multiple machines in a list, click **Manually**, enter the machine details in the Add Machines Manually dialog box, and click **Add**. For each machine, you will need to enter the IP address or name for the machine, the user name, the password separated by a double-colon delimiter, and port as shown in the following format:

  ```
  hostname::username::password::port
  ```

  For example:

  ```
  10.255.255.255::administrator::&11@yYz90z::8006

  abc-host-00-1::administrator::99!zU$o83r::168
  ```

The machines you added appear on the Deploy Agent to Machines window. The system verifies each machine you added automatically.

3 Optionally, you can verify any machine, check the status of the machine, or edit connection settings for a machine, as described in Step 6 to Step 8 of the procedure Deploying to machines on an Active Directory domain.

4 After machines are verified successfully, check the box next to each machine to which you want to deploy the AppAssure Agent, and then click **Deploy**.

If you proceed with protecting an agent that is already protected by another Core, that protection will be stopped and the machine will now be protected by this Core.

If you chose the **Protect machine after install** option, after deployment is successful the machines are rebooted automatically and protection is enabled.

Verifying the deployment to multiple machines

Once you have bulk deployed the AppAssure Agent software to two or more machines, you can verify the success by viewing each agent machine listed under the Protected Machines menu.

You can also view information regarding the bulk deploy process from the Events tab. Complete the steps in this procedure to verify the deployment.

To verify the deployment to multiple machines

1 Navigate to the AppAssure Core Console, click the Events tab, and click **Alerts**.

2 Navigate to the AppAssure Core Home tab and then click the **Events** tab.

Alert events appear in the list, showing the time the event initiated and a message. For each successful deployment of the Agent software, you will see an alert indicating that the protected machine has been added.

3 Optionally, click on any link for a protected machine.

The Summary tab for the selected machine appears, showing pertinent information including:

- The host name of the protected machine
- The last snapshot, if applicable
- The scheduled time of the next snapshot, based on the protection schedule you selected
- The encryption key, if any, used for this protected agent.
- The version of the Agent software.

Understanding protection schedules

A protection schedule defines when backups are transferred from protected agent machines to the AppAssure Core.

Protection schedules are initially defined using the Protect Machine Wizard or the Protect Multiple Machines Wizard. You can then modify the existing schedule at any time from the Summary tab for a specific agent machine.

NOTE: For information about protecting a single machine, see Protecting a machine. For information about bulk protect (protecting multiple machines), see Protecting multiple machines. For information on customizing protection periods when protecting an agent using either of these wizards, see Creating custom protection schedules. For information about modifying an existing protection schedule, see Modifying protection schedules.

AppAssure provides a default protection schedule, which includes a single period spanning all days of the week, with a single time period defined (from 12:00 AM to 11:59 PM). The default interval (the time period between snapshots) is 60 minutes.

When protection is first enabled, the schedule is activated. Thus, using the default settings, regardless of the current time of day, the first backup will occur every hour, on the hour (12:00 AM, 1:00 AM, 2:00 AM, and so on).

The first backup transfer saved to the Core is called a base image snapshot. All data on all specified volumes (including the operating system, applications, and settings), are saved to the Core. Thereafter, incremental snapshots (smaller backups, consisting only of data changed on the protected machine since the last backup) are saved to the core regularly, based on the interval defined (for example, every 60 minutes).

You can create a custom schedule to change the frequency of backups. For example, a simple change you can make is to change the interval for the weekday period to 20 minutes, resulting in three snapshots every hour. Or you can increase the interval on weekends from 60 minutes to 180 minutes, resulting in snapshots once every three hours when traffic is low.

You can also define peak and off-peak times. For example, if your protected machines are mostly in use on weekdays, you could define Period #1 to include only weekdays, by clearing the selections for Sunday and Saturday in the Period #1 drop-down menu. Then, select Period #2, which automatically covers the time period not included in Period #1 (in this case, Saturday and Sunday). You can then increase the interval for period two, for example to 360 minutes, resulting in snapshots once every six hours on weekends, and once hourly on weekdays.

Alternatively, you can change the default schedule to define peak and off-peak times daily. To do this using the Protection Schedule Wizard, change the default start and end time to a smaller range of time (for example, 12:00 AM to 4:59 PM), and set an appropriate interval (for example, 20 minutes). This represents frequent backups during peak periods.

Then, select **Period#2,** change the time range to the remaining period (5:00 pm to 11:59 pm) and set an appropriate (presumably larger) interval (for example, 180 minutes). These settings define an off-peak period

that includes 5:00 PM to midnight every day. This results in snapshots every 3 hours from 5:00 PM through 11:59 PM, and snapshots every 20 minutes from midnight to 4:59 PM.

Other options in the Protection Schedule Wizard page include for a setting a daily protection time. This results in a single backup daily at the period defined (the default setting is 12:00 PM).

The option to initially pause protection prevents a base image from occurring (and in fact, prevents all backups) until you explicitly resume protection. When you are ready to begin protecting your machines based on the established protection schedule, you must explicitly resume protection. For more information on resuming protection, see Pausing and resuming protection. Optionally, if you want to protect a machine immediately, you can force a snapshot. For more information, see Forcing a snapshot.

Protecting a machine

This topic describes how to start protecting the data on a single machine that you specify using the Protect Machine Wizard.

NOTE: The machine must have the AppAssure Agent software installed in order to be protected. You can choose to install the Agent software prior to this procedure, or you can deploy the software to the target machine as a part of completing the Protect Machine Wizard. For more information on installing the Agent software, see "Installing the AppAssure Agent software" in the *Dell AppAssure Installation and Upgrade Guide*.

If the Agent software is not installed prior to protecting a machine, you will not be able to select specific volumes for protection as part of this wizard. In this case, by default, all volumes on the agent machine will be included for protection.

AppAssure supports the protection and recovery of machines configured with EISA partitions. Support is also extended to Windows 8 and 8.1, and Windows 2012 and 2012 R2 machines that use Windows Recovery Environment (Windows RE).

To protect multiple machines using one process simultaneously, see Protecting multiple machines.

When you add protection, you need to define connection information such as the IP address and port, and provide credentials for the machine you want to protect. Optionally, you can provide a display name to appear in the Core Console instead of the IP address. If you change this, you will not see the IP address for the protected machine when you view details in the Core Console. You will also define the protection schedule for the machine.

This process includes optional steps you can access if you select an advanced configuration. Advanced options include repository functions and encryption. For example, you can specify an existing AppAssure repository to save snapshots, or create a new repository. You can also specify an existing encryption key (or add a new encryption key) to apply to the data saved to the Core for this machine.

The workflow of the wizard may differ slightly based on your environment. For example, if the AppAssure Agent software is installed on the machine you want to protect, you will not be prompted to install it from the wizard. Likewise, if a repository already exists on the Core, you will not be prompted to create one.

CAUTION: AppAssure does not support bare metal restores (BMRs) of Linux machines with EXT2 boot partitions. Any BMR performed on a machine with this type of partition results in a machine that does not start. If you want to be able to perform a BMR on this machine in the future, you must convert any EXT2 partitions to EXT3 or EXT4 before you begin protecting and backing up the machine.

To protect a machine

1. If you have already installed the AppAssure Agent software on the machine you want to protect, but have not restarted it yet, restart the machine now.
2. On the core machine, navigate to the AppAssure Core Console, and from the button bar, click **Protect**.

   The Protect Machine Wizard appears.

3 On the Welcome page, select the appropriate installation options:

- If you do not need to define a repository or establish encryption, select **Typical**.
- If you need to create a repository, or define a different repository for backups for the selected machine, or if you want to establish encryption using the wizard, select **Advanced (show optional steps)**.
- Optionally, if you do not wish to see the Welcome page for the Protect Machine Wizard in the future, select the **Skip this Welcome page the next time the wizard opens** option.

4 When you are satisfied with your choices on the Welcome page, then click **Next**.

The Connection page appears.

5 On the Connection page, enter the information about the machine to which you want to connect as described in the following table, and then click **Next**.

Table 51. Machine connection settings

Text Box	Description
Host	The host name or IP address of the machine that you want to protect.
Port	The port number on which the AppAssure Core communicates with the Agent on the machine. The default port number is 8006.
User name	The user name used to connect to this machine; for example, Administrator (or, if the machine is in a domain, [domain name]\Administrator).
Password	The password used to connect to this machine.

6 If the Upgrade Agent page appears next in the Protect Machine Wizard, this means that an older version of the AppAssure Agent software exists on the machine you want to protect. Do one of the following and then click **Next**:

- To deploy the new version of the Agent software (matching the version for the AppAssure Core), select **Upgrade the Agent to the latest version of the software**.
- To continue protecting the machine without updating the Agent software version, clear the **Upgrade the Agent to the latest version of the software** option.

7 If the Protection page appears next in the Protect Machine Wizard, skip to Step 8.

If the Install Agent page appears next in the Protect Machine Wizard, this indicates that the Agent software is not yet on installed on the designated machine. Click **Next** to install the Agent software.

> **NOTE:** The Agent software must be installed on the machine you want to protect, and that machine must be restarted, before it can back up to the Core. To have the installer reboot the protected machine, select the **After installation, restart the machine automatically (recommended)** option before clicking **Next**.

The Protection page appears.

8 Optionally, if you want a name other than the IP address to display in the AppAssure Core console for this protected machine, then in the Display Name field, type a name in the dialog box.

You can enter up to 64 characters. Do not use the special characters described in the topic prohibited characters. Additionally, do not begin the display name with any of the character combinations described in the topic prohibited phrases.

9 Select the appropriate protection schedule as described below.To use the default protection schedule, in the Schedule Settings option, select **Default protection**.

With a default protection schedule, the Core will take snapshots of all volumes on the protected machine once every hour. To change the protection settings at any time after you close the wizard, including choosing which volumes to protect, go to the Summary tab for the specific protected machine.

- To define a different protection schedule, in the Schedule Settings option, select **Custom protection**.

10 Proceed with your configuration as follows:

- If you selected a Typical configuration for the Protect Machine Wizard and specified default protection, then click **Finish** to confirm your choices, close the wizard, and protect the machine you specified.

  The first time protection is added for a machine, a base image (that is, a snapshot of all the data in the protected volumes) will transfer to the repository on the AppAssure Core following the schedule you defined, unless you specified to initially pause protection.

- If you selected a Typical configuration for the Protect Machine Wizard and specified custom protection, then click **Next** to set up a custom protection schedule. For details on defining a custom protection schedule, see Creating custom protection schedules.

- If you selected Advanced configuration for the Protect Machine Wizard, and default protection, then click **Next** and proceed to Step 14 to see repository and encryption options.

- If you selected Advanced configuration for the Protect Machine Wizard and specified custom protection, then click **Next** and proceed to Step 11 to choose which volumes to protect.

11 On the Protection Volumes page, select which volumes you want to protect. If any volumes are listed that you do not want to include in protection, click in the Check column to clear the selection. Then click **Next**.

NOTE: Typically, it is good practice to protect, at minimum, the System Reserved volume and the volume with the operating system (typically the C drive).

12 On the Protection Schedule page, define a custom protection schedule and then click **Next**. For details on defining a custom protection schedule, see Creating custom protection schedules.

13 If you already have repository information configured, the Encryption page appears. Proceed to Step 19

14 On the Repository page, do one of the following:

- If you want to store the data from this machine for protection in an existing repository, then select **Use an existing repository**, select the appropriate repository from the list, and then click **Next**.

  The Encryption page appears. Skip to Step 19 to optionally define encryption.

- If you want to create a new storage location on the Core, then do the following:

  a Select **Create a Repository**.

  b On the Repository page, in the **Name** field, specify the name of the repository you want to create.

  This is typically the word Repository and an index number, which corresponds to the number of the new repository (for example, **Repository1**). You can change the name as needed. You can enter up to 40 characters. This name must be unique for this core.

NOTE: When specifying the repository name, use only alphanumeric characters or the hyphen. No other symbols or punctuation characters are permitted. Do not use letter combinations that specify commands or reserved words (such as con, prn, aux, or null) or that represent ports (such as com or lpt).

c In the **Location** field, enter a directory path for the repository. For example, on a local computer, type **C:\Repository**. This location must be unique for this core. If storing your repository on a shared drive, enter in format \\servername\sharename.

CAUTION: If you delete your repository in the future, the Installer program will remove the entire contents of the repository path. For this reason, do not create the storage location at the root (for example, c:\), which could result in losing all data stored on that volume.

d If storing the repository on a shared volume, in the **User Name** field, enter the user name with privileges to access the shared drive, and in the **Password** field, enter the password for this user.

e In the **Metadata path** field, enter the path where you want metadata to be stored. This should be a subdirectory of the storage location. For example, if the storage location is C:\Repository, type **C:\Repository\Metadata**. This must be a unique path for this core.

15 When you have entered all required data for the Repository page, click **Next**.

The Repository Configuration page appears.

16 Specify the size of the repository.

NOTE: If the storage location is a New Technology File System (NTFS) volume using Windows XP or Windows 7, the file size limit is 16 TB.

If the storage location is a NTFS volume using Windows 8, 8.1 or Windows Server 2012, 2012 R2, the file size limit is 256 TB.

For AppAssure to validate the storage location, Windows Management Instrumentation (WMI) must be installed and accessible on the operating system hosting the AppAssure Core.

17 To specify bytes per sector, bytes per record, or control the write caching policy, select **Show advanced options**, and then enter details for the storage location as described in the following table.

Table 52. Advanced options for storage

Text Box	Description
Bytes per Sector	Specify the number of bytes you want each sector to include. The default value is *512*.
Bytes per Record	Specify the average number of bytes per record. The default value is *8192*.
Write Caching Policy	The write caching policy controls how the Windows Cache Manager is used in the repository and helps to tune the repository for optimal performance on different configurations. Set the value to one of the following: • On • Off • Sync If set to *On*, which is the default, Windows controls the caching. **NOTE:** Setting the write caching policy to *On* could result in faster performance. If you are using a version of Windows Server prior to Server 2012, the recommended setting is *Off*. If set to *Off*, AppAssure controls the caching. If set to *Sync*, Windows controls the caching as well as the synchronous input/output.

18 When you are satisfied with the repository configuration information you entered, click **Next**.

The Encryption page appears.

19 Optionally, to enable encryption, on the Encryption page, select **Enable Encryption**.

Encryption key fields appear on the Encryption page.

NOTE: If you enable encryption, it will be applied to data for all protected volumes for this agent machine.

You can change the settings later from the Configuration tab in the AppAssure Core Console.

For more information about encryption, see the topic Understanding encryption keys.

CAUTION: AppAssure uses AES 256-bit encryption in the Cipher Block Chaining (CBC) mode with 256-bit keys. While using encryption is optional, Dell highly recommends that you establish an encryption key, and that you protect the passphrase you define. Store the passphrase in a secure location as it is critical for data recovery. Without a passphrase, data recovery is not possible.

20 If you want to encrypt this protected machine using an encryption key that is already defined on this AppAssure Core, select **Encrypt data using an existing Encryption key**, and select the appropriate key from the drop-down menu.

Proceed to Step 22.

21 If you want to add a new encryption key to the Core and apply that key to this protected machine, then enter the information as described in the following table.

Table 53. Encryption key settings

Text Box	Description
Name	Enter a name for the encryption key. Encryption key names must contain between 1 and 130 alphanumeric characters. You may not include special characters such as the back slash, forward slash, pipe, colon, asterisk, quotation mark, question mark, open or close brackets, ampersand or hash.
Description	Enter a comment for the encryption key. This information appears in the Description field when viewing encryption keys from the Core Console.
Passphrase	Enter the passphrase used to control access. Best practice is to avoid special characters listed above. Record the passphrase in a secure location. Dell Support cannot recover a passphrase. Once you create an encryption key and apply it to one or more protected machines, you cannot recover data if you lose the passphrase.
Confirm Passphrase	Re-enter the passphrase you just entered.

22 Click **Finish** to save and apply your settings.

The first time protection is added for a machine, a base image (that is, a snapshot of all the data in the protected volumes) will transfer to the repository on the AppAssure Core following the schedule you defined, unless you specified to initially pause protection.

Creating custom protection schedules

When defining protection using the Protect a Machine Wizard or the Protecting Multiple Machines Wizard, you must define a protection schedule.

The default protection schedule includes a single period that includes each day of the week. The default time range is from 12:00 AM to 11:59 PM, covering a full 24-hour time period. The default interval is 60 minutes.

Using the wizard, you can customize protection schedules, choosing either periods or a daily protection time.

Selecting periods allows you to view the default protection schedule and make adjustments accordingly. Selecting a daily protection time causes AppAssure to back up the designated protected machines once daily at a time you specify.

When using periods, you can make simple changes to the default protection schedule. For example, you may want to simply change the interval in the default weekday schedule to every 20 minutes, resulting in backups three times hourly instead of once hourly, or every 180 minutes, resulting in backups once every three hours.

You can also create more complex protection schedules. For example, you can create peak periods for weekdays, and off-peak periods for weekends, as described in Understanding protection schedules.

Finally, when protecting one or multiple machines using the wizard, you can initially pause protection, which defines the protection schedule without starting protection. When you are ready to begin protecting your machines based on the established protection schedule, you must explicitly resume protection. For more information on resuming protection, see Pausing and resuming protection. Optionally, if you want to protect a machine immediately, you can force a snapshot. For more information, see Forcing a snapshot.

NOTE: For conceptual information about protection schedules, see Understanding protection schedules. For information about protecting a single machine, see Protecting a machine. For information about bulk protect (protecting multiple machines), see Protecting multiple machines. For information on customizing protection periods when protecting an agent using either of these wizards, see Creating custom protection schedules. For information about modifying an existing protection schedule, see Modifying protection schedules.

Complete the steps in this procedure to create custom schedules for protecting data on agent machines when defining protection using a wizard.

To create custom schedules

1. On the Protection Schedule page of the Protect Machine or Protect Multiple Machines Wizard, to change the interval schedule for any period, do the following:
   a. Select **Periods**.

      The existing periods display and can be modified. Editable fields include a start time, end time, and interval (in minutes) for each period.

   b. Click in the interval field and type an appropriate interval in minutes.

      For example, highlight the existing interval and replace it with the value **20** to perform snapshots every 20 minutes in this period.
2. To create a peak and off-peak period for weekdays, change the time range of the weekday period so that it does not include a 24-hour period, set an optimal interval for the peak, select **Take snapshots for the remaining time** and set an off-peak interval, by doing the following:
   a. Select **Periods**.

      The existing periods display and can be modified.

   b. Click in the **From** box to change the start time for this period.

      The Choose Time dialog box appears.

   c. Drag the Hours and Minutes slider controls as appropriate for the desired start time, and then click **Done**. To specify the current time, click **Now**.

      For example, drag the Hours control to show a time of 08:00 AM

   d. Click in the **To** box to change the end time for this period.

      The Choose Time dialog box appears.

   e. Drag the Hours and Minutes slider controls as appropriate for the desired start time, and then click **Done**. To specify the current time, click **Now**.

      For example, drag the Hours control to show a time of 04:59 PM
3. To set a single time of day for a single backup to occur daily, select **Daily protection time** and then enter a time in format HH:MM AM. For example, to do a daily backup at 9:00 PM, enter 09:00 PM.

4 To define the schedule without beginning backups, select **Initially pause protection**.

After you pause protection from the wizard, it remains paused until you explicitly resume it. Once you resume protection, backups will occur based on the schedule you established. Fore more information on resuming protection, see Pausing and resuming protection.

5 When you are satisfied with changes made to your protection schedule, click **Finish** or **Next**, as appropriate. Return to the procedure for the appropriate wizard to complete any requirements remaining.

Modifying protection schedules

A protection schedule defines when backups are transferred from protected agent machines to the AppAssure Core. Protection schedules are initially defined using the Protect Machine Wizard or the Protect Multiple Machines Wizard.

You can modify an existing protection schedule at any time from the Summary tab for a specific agent machine.

NOTE: For conceptual information about protection schedules, see Understanding protection schedules. For information about protecting a single machine, see Protecting a machine. For information about bulk protect (protecting multiple machines), see Protecting multiple machines. For information on customizing protection periods when protecting an agent using either of these wizards, see Creating custom protection schedules. For information about modifying an existing protection schedule, see Modifying protection schedules.

Complete the steps in this procedure to modify an existing protection schedule for volumes on a protected machine.

To modify protection schedules

1 Navigate to the AppAssure Core Console.

2 From the list of protected machines, select the machine with a defined protection schedule that you want to change.

The Summary tab displays for the machine.

3 Select the volumes for the protected machine that you want to change, and click **Set a schedule**. To select all volumes at once, click in the checkbox in the header row.

Initially, all volumes share the same protection schedule. Typically, it is good practice to protect, at minimum, the System Reserved volume and the volume with the operating system (typically the C drive).

The Protection Schedule dialog box appears.

4 On the Protection Schedule dialog box, if you previously created a protection schedule template and want to apply it to this agent, select the template from the drop-down list, and then go to Step 10.

5 If you want to save this new protection schedule as a template, enter a name for the template in the text box.

6 If you want to remove an existing time period from the schedule, clear the check boxes next to each time period option. Options include the following:

- **Mon - Fri.** This range of time denotes a typical five-day work week.
- **Sat - Sun.** This range of time denotes a typical weekend.

7 If the weekday start and end times are from 12:00 AM to 11:59 PM, then a single period exists. To change the start or end time of a defined period, do the following:

a Select the appropriate time period.

b Click in the **Start Time** box to change the start time for this period.

The Choose Time dialog box appears.

c Drag the Hours and Minutes slider controls as appropriate for the desired start time, and then click **Done**. To specify the current time, click **Now**.

For example, drag the Hours control to show a time of 08:00 AM

d Click in the **End Time** box to change the end time for this period.

The Choose Time dialog box appears.

e Drag the Hours and Minutes slider controls as appropriate for the desired start time, and then click **Done**. To specify the current time, click **Now**.

For example, drag the Hours control to show a time of 04:59 PM

f Change the interval according to your requirements. For example, if defining a peak period. change the interval from 60 minutes to 20 minutes to take snapshots three times hourly.

8 If you defined a period other than 12:00 AM to 11:59 PM in Step 7, then if you want backups to occur in the remaining time ranges, you must add additional periods to define protection by doing the following:

a Click **+ Add period.**

Under the appropriate category (weekdays or weekends), a new time period appears. If the first period started later than 12:00 AM, then AppAssure automatically starts this period at 12:00. Following the above example, this second period starts at 12:00 AM. You may need to adjust hours or minutes for the start and end times.

b Drag the Hours and Minutes slider controls as appropriate for the desired start and end times, as appropriate.

For example, set a start time of 12:00 AM and an end time of 07:59 AM.

c Change the interval according to your requirements. For example, if defining an off-peak period. change the interval from 60 minutes to 120 minutes to take snapshots every two hours.

9 If needed, continue to create additional periods, setting start and end times and intervals as appropriate.

NOTE: If you want to remove a period you have added, click the **X** to the far right of that period. If you remove a period in error, you can click **Cancel**.

10 When your protection schedule meets your requirements, click **Apply**.

The protection Schedule dialog box closes.

Pausing and resuming protection

When you pause protection, you temporarily stop all transfers of data from the selected machine to the AppAssure Core. When you resume protection, the AppAssure Core follows the requirements in the protection schedule, backing up your data regularly based on that schedule.

You can pause protection for any AppAssure agent machine:

- When establishing protection using the Protect Machine Wizard or the Protect Multiple Machines Wizard.
- From the Protected Machines menu in the left navigation area of the AppAssure Core (pausing protection for all agents).
- From the Protected Machines page (accessible when you click on the Protected Machines menu).
- From a specific protected machine in the Protected Machines menu.
- From the Summary tab of any protected agent machine.

If you pause protection using the Protect Machine Wizard or the Protect Multiple Machines Wizard, protection is paused until explicitly resumed.

If you pause protection outside of a wizard, you can choose whether to pause protection until resumed, or to pause it for a designated amount of time (specified in any combination of days, hours and minutes). If you pause

for a period of time, then when that time expires, the system resumes protection based on the protection schedule automatically.

You can resume protection for any paused AppAssure agent:

- From the Protected Machines menu in the left navigation area of the AppAssure Core (resuming protection for all agents).
- From a specific protected machine in the Protected Machines menu.
- From the Protected Machines page (accessible when you click on the Protected Machines menu).
- From the Summary tab of any protected agent machine.

Use the procedure below to pause or to resume protection, as appropriate.

To pause and resume protection

1 In the AppAssure Core Console, to pause or resume protection for all machines, click the **Protected Machines** drop-down menu in the left navigation area.

- If you want to pause protection, do the following:
  - a Select **Pause Protection**.

    The Pause Protection dialog box appears.
  - b Select the appropriate setting using one of the options described below, and then click **OK**.
    - If you want to pause protection until you explicitly resume it, select **Pause until resumed**.
    - If you want to pause protection for a specified period, select **Pause for** and then, in the Days, Hours, and Minutes controls, type or select the appropriate pause period as appropriate.
- If you want to resume protection, do the following:
  - a Select **Resume Protection**.

    The Resume Protection dialog box appears.
  - b In the Resume Protection dialog box, select **Yes**.

    The Resume Protection dialog box closes, and protection is resumed for all machines.

2 To pause or resume protection for a single machine from any tab, then in the left navigation area, from the list of protected machines, click the arrow to the right of the machine you want to affect.

- If you want to pause protection, do the following:
  - a Select **Pause Protection**.

    The Pause Protection dialog box appears.
  - b Select the appropriate setting using one of the options described below, and then click **OK**.
    - If you want to pause protection until you explicitly resume it, select **Pause until resumed**.
    - If you want to pause protection for a specified period, select **Pause for** and then, in the Days, Hours, and Minutes controls, type or select the appropriate pause period as appropriate.
- If you want to resume protection, do the following:
  - a Select **Resume Protection**.

    The Resume Protection dialog box appears.

b In the Resume Protection dialog box, select **Yes**.

The Resume Protection dialog box closes, and protection is resumed for the selected machine.

3 To pause or resume protection for a single machine from the Summary tab, navigate to the machine that you want to affect.

The Summary tab displays for the selected machine appears.

- If you want to pause protection, do the following:

  a In the **Actions** drop-down menu for that machine, select **Pause Protection**.

  The Pause Protection dialog box appears.

  b Select the appropriate setting using one of the options described below, and then click **OK**.

  - If you want to pause protection until you explicitly resume it, select **Pause until resumed**.
  - If you want to pause protection for a specified period, select **Pause for** and then, in the Days, Hours, and Minutes controls, type or select the appropriate pause period as appropriate.

- If you want to resume protection, do the following:

  a Select **Resume Protection**.

  The Resume Protection dialog box appears.

  b In the Resume Protection dialog box, select **Yes**.

  The Resume Protection dialog box closes, and protection is resumed for the selected machine.

Protecting multiple machines

You can add two or more Windows machines for protection on the AppAssure Core simultaneously using the Protect Multiple Machines Wizard. This feature is called bulk protection.

As with protecting individual machines, bulk protection requires the AppAssure Agent software to be installed on each machine you want to protect, and requires that machine to be restarted after installation of the Agent software. There is more than one method to deploy the Agent software to multiple machines simultaneously. For example:

- You can install the AppAssure Agent software to multiple machines using the bulk deploy feature, accessed from the Tools tab. (If you selected **Protect Machine After Install** when you deployed the Agent, you can skip this procedure.) For more information about using bulk deploy, see Deploying to multiple machines.
- You can deploy the AppAssure Agent software as part of this wizard.

NOTE: Protected machines must be configured with a security policy that makes remote installation possible.

This process includes optional steps you can access if you select an advanced configuration. Advanced options include repository functions and encryption. For example, you can specify an existing AppAssure repository to save snapshots, or create a new repository. You can also specify an existing encryption key (or add a new encryption key) to apply to the data saved to the Core for this machine.

The workflow of the Protect Multiple Machines Wizard may differ slightly based on your environment. For example, if the AppAssure Agent software is installed on the machines you want to protect, you will not be prompted to install it from the wizard. Likewise, if a repository already exists on the Core, you will not be prompted to create one.

To protect multiple machines

1 If you have already installed the AppAssure Agent software on the machines you want to protect, but have not restarted them yet, restart the machines now.

2 On the Core machine, navigate to the AppAssure Core Console, click the drop-down menu next to the Protect icon, and select **Bulk Protect**.

The Protect Multiple Machines Wizard appears.

3 On the Welcome page, select the appropriate installation options:

- If you do not need to define a repository or establish encryption, select **Typical**.
- If you need to create a repository, or define a different repository for backups for the selected machines, or if you want to establish encryption using the wizard, select **Advanced (show optional steps)**.
- Optionally, if you do not wish to see the Welcome page for the Protect Machine Wizard in the future, select the **Skip this Welcome page the next time the wizard opens** option.

4 When you are satisfied with your choices on the Welcome page, then click **Next**.

The Connection page appears.

5 Select the appropriate method to identify the machines you want to add for protection.

The machines must be on and accessible in order to connect to them. For successful connection using Active Directory, bulk protection and bulk deploy will be most successful if logged into the machines as the domain administrator.

- To identify the machines you want to protect on an Active Directory domain, select **Connect to Active Directory**, enter credentials as described in the following table, and then click **Next**. Skip to Step 7.
- To identify the machines you want to protect on a VMware vCenter/ESX(i) virtual host, select **Connect to vCenter/ESX(i)**, enter credentials as described in the following table, and then click **Next**. Skip to Step 7.

Table 54. vCenter/ESX(i) credentials

Text Box	Description
Host	The host name or IP address of the Active Directory domain or of the VMware vCenter Server/ESX(i) virtual host.
User name	The user name used to connect to the domain: for example, Administrator or, if the machine is in a domain, [domain name]\Administrator).
Password	The secure password used to connect to the domain.

NOTE: All virtual machines must have VM Tools installed; otherwise, AppAssure cannot detect the host name of the virtual machine to which to deploy the Agent software, and cannot protect that machine. In lieu of the host name, AppAssure uses the virtual machine name, which may cause issues if the host name is different from the virtual machine name.

- To add the machines manually, select **Add the machines manually** and then click **Next**.

The Machines page appears.

6 On the Machines page, to specify machines manually, type the connection details for each machine on a separate line, and then click **Next**. Use the following format:

```
hostname::username::password::port
```

NOTE: The port parameter is optional; if omitted, port 8006 will be used by default. If you wish to specify a port different than 8006, then inclusion of this parameter is required.

7 On the Machines page, to specify machines identified from an Active Directory domain or from a VMware vCenter/ESX(i) virtual host, select each appropriate machine you want to protect from the list. Ensure you clear the checkbox option for any machine you do not want to protect at this time. When satisfied, then click **Next**.

The system verifies each machine you added automatically.

8 If the Protection page appears next in the Protect Multiple Machines Wizard, skip to Step 12.

If the Agent software is not yet deployed to the machines you want to protect, or if any of the machines you specified cannot be protected for another reason, then the selected machines appear on the Machines Warnings page. The system verifies each machine you added automatically.

9 Optionally, on the Machines Warnings page, you can verify any machine by selecting the machine and then clicking **Verify** in the toolbar.

10 Optionally, on the Machines Warnings page, select **After Agent installation, restart the machines automatically**.

> **NOTE:** Dell recommends this option. You must restart agent machines before they can be protected.

11 If the status indicates that the machine is reachable, click **Next** to install the agent software.

The Protection page appears.

12 On the Protection page, select the appropriate protection schedule as described below.

- If you want to use the default protection schedule, then in the Schedule Settings option, select **Default protection (hourly snapshots of all volumes)**.
- If you want to define a different protection schedule, then in the Schedule Settings option, select **Custom protection** and then click **Next**.

13 Proceed with your configuration as follows:

- If you selected a Typical configuration for the Protect Multiple Machines Wizard, and default protection, then click **Finish** to confirm your choices, close the wizard, and protect the machines you specified.
- If you selected a Typical configuration for the Protect Multiple Machines Wizard and specified custom protection, then on the Protection page, click **Next**, and on the Protection Schedule page, set up a custom schedule as described in the topic Creating custom protection schedules, and then click **Finish** to confirm your choices, close the wizard, and protect the machines you specified.
- If you selected Advanced configuration for the Protect Machine Wizard, then click **Next** and proceed to Step 14 to see repository and encryption options.

14 On the Repository page, if you want to store the data for all of the machines you specified for protection in an existing repository, then select **Use an existing repository**, select the appropriate repository from the list, and then click **Next** and proceed to Step 19.

If you want to create a new storage location on the Core, then do the following:

a Select **Create a Repository**.

b On the Repository page, in the **Name** field, specify the name of the repository you want to create.

This is typically the word Repository and an index number, which corresponds to the number of the new repository (for example, **Repository1**). You can change the name as needed. You can enter up to 40 characters. This name must be unique for this core.

> **NOTE:** When specifying the repository name, use only alphanumeric characters or the hyphen. No other symbols or punctuation characters are permitted. Do not use letter combinations that specify commands or reserved words (such as con, prn, aux or nul) or that represent ports (such as com or lpt).

c In the **Location** field, enter a directory path for the repository. For example, on a local computer, type **C:\Repository**. This location must be unique for this core. If storing your repository on a shared drive, enter in format \\servername\sharename.

△ **CAUTION: If you delete your repository in the future, the Installer program will remove the entire contents of the repository path. For this reason, do not create the storage location at the root (for example, c:\), which could result in losing all data stored on that volume.**

d If storing the repository on a shared volume, in the **User Name** field, enter the user name with privileges to access the shared drive, and in the **Password** field, enter the password for this user.

e In the **Metadata path** field, enter the path where you want metadata to be stored. This should be a subdirectory of the storage location. For example, if the storage location is C:\Repository, type **C:\Repository\Metadata**. This must be a unique path for this core.

15 When you have entered all required data for the Repository page, click **Next**.

The Repository Configuration page appears.

16 Specify the size of the repository.

ⓘ **NOTE:** If the storage location is a New Technology File System (NTFS) volume using Windows XP or Windows 7, the file size limit is 16 TB.

If the storage location is a NTFS volume using Windows 8, 8.1 or Windows Server 2012, 2012 R2, the file size limit is 256 TB.

For AppAssure to validate the operating system, Windows Management Instrumentation (WMI) must be installed on the intended storage location.

17 To specify bytes per sector, bytes per record, or control the write caching policy, select **Show advanced options**, and then enter details for the storage location as described in the following table.

Table 55. Advanced options for storage

Text Box	Description
Bytes per Sector	Specify the number of bytes you want each sector to include. The default value is *512*.
Bytes per Record	Specify the average number of bytes per record. The default value is *8192*.
Write Caching Policy	The write caching policy controls how the Windows Cache Manager is used in the repository and helps to tune the repository for optimal performance on different configurations. Set the value to one of the following: • On • Off • Sync If set to *On*, which is the default, Windows controls the caching. **NOTE:** Setting the write caching policy to *On* could result in faster performance. If you are using a version of Windows Server prior to Server 2012, the recommended setting is *Off*. If set to *Off*, AppAssure controls the caching. If set to *Sync*, Windows controls the caching as well as the synchronous input/output.

18 When you are satisfied with the repository configuration information you entered, click **Next**.

The Encryption page appears.

19 Optionally, to enable encryption, on the Encryption page, select **Enable Encryption**.

Encryption key fields appear on the Encryption page.

NOTE: If you enable encryption, it will be applied to data for all protected volumes for the machines you have specified for protection. You can change the settings later from the Configuration tab in the AppAssure Core Console. For more information about encryption see, Understanding encryption keys.

CAUTION: AppAssure uses AES 256-bit encryption in the Cipher Block Chaining (CBC) mode with 256-bit keys. While using encryption is optional, Dell highly recommends that you establish an encryption key, and that you protect the passphrase you define. Store the passphrase in a secure location as it is critical for data recovery. Without a passphrase, data recovery is not possible.

20 If you want to encrypt these protected machines using an encryption key that is already defined on this AppAssure Core, select **Encrypt data using an existing Encryption key**, and select the appropriate key from the drop-down menu.

Proceed to Step 22.

21 If you want to add a new encryption key to the Core and apply that key to these protected machines, then enter the information as described in the following table.

Table 56. Encryption key settings

Text Box	Description
Name	Enter a name for the encryption key. Encryption key names must contain between 1 and 130 alphanumeric characters. You may not include special characters such as the back slash, forward slash, pipe, colon, asterisk, quotation mark, question mark, open or close brackets, ampersand or hash.
Description	Enter a comment for the encryption key. This information appears in the Description field when viewing encryption keys from the Core Console.
Passphrase	Enter the passphrase used to control access. Best practice is to avoid special characters listed above. Record the passphrase in a secure location. Dell Support cannot recover a passphrase. Once you create an encryption key and apply it to one or more protected machines, you cannot recover data if you lose the passphrase.
Confirm Passphrase	Re-enter the passphrase you just entered.

22 Click **Finish** to save and apply your settings.

The wizard closes. The AppAssure Agent software is deployed to the specified machines, if necessary, and the machines are added to protection on the Core.

Monitoring the protection of multiple machines

You can monitor the progress as AppAssure applies the protection polices and schedules to the machines.

To monitor the protection of multiple machines

- In the AppAssure Core Console, navigate to the AppAssure **Home** tab and then click the **Events** tab.

  The Events tab displays, broken down by Tasks, Alerts, and Events. As volumes are transferred, the status, start times, and end times display in the Tasks pane.

You can also filter tasks by status (active, waiting, completed and failed). For more information, see Viewing tasks.

NOTE: To only see tasks that are waiting to be performed, make sure that you select the Waiting Tasks icon.

As each protected machine is added, an alert is logged, which lists whether the operation was successful or if errors were logged. For more information, see Viewing alerts.

For information on viewing all events, see Viewing all events.

7

Managing Microsoft Exchange and SQL Servers

This chapter describes how to configure, and manage the protection of Microsoft Exchange and SQL Servers in your AppAssure environment. It includes the following sections:

- Configuring Exchange and SQL Server settings
- Managing SQL attachability and log truncation
- Managing Exchange database mountability checks and log truncation

Options specific to Exchange Server and SQL Server appear in the AppAssure Core Console when an instance of the software and associated databases are detected on protected servers. This section includes the following topics specific to managing protected machines that use Exchange Server or SQL Server:

This section includes the following topics:

- Modifying Exchange Server settings
- Forcing a mountability check of an Exchange database
- Forcing a checksum check of Exchange Server recovery points
- Modifying SQL Server settings
- Forcing a SQL Server attachability check
- Forcing log truncation for a SQL machine

When you protect SQL Servers and Exchange Servers, there are functions specific to these server types that you can perform. These include:

- **Forcing server log truncation.** Both SQL Servers and Exchange Servers include server logs. The process for truncating SQL logs identifies available space on the server. When you truncate Exchange server logs, in addition to identifying the available space, the process frees up more space on the server. For information on forcing log truncation, see Forcing log truncation for a SQL machine.
- **Setting credentials for the relevant server.** Exchange servers allow you to set credentials for the protected machine in the Summary tab for the protected server. SQL Servers allow you to set credentials for a single protected SQL server machine, or to set default credentials for all protected SQL Servers.
    - For information on setting credentials for Exchange servers, see Setting credentials for an Exchange Server machine.
    - For information on setting credentials for SQL servers, see Setting credentials for a SQL Server machine.
    - For information on performing other actions accessible to all protected agents from the agent Summary tab, see Viewing the Summary tab.

Configuring Exchange and SQL Server settings

If you protect data on Exchange servers or SQL servers on your AppAssure Core, you can establish credentials so the Core can authenticate to those machines. Navigate to the protected server in the AppAssure Core console to configure these server settings, as described in the following topics.

Setting credentials for an Exchange Server machine

Once you protect data on a Microsoft Exchange server, you can set login credentials in the AppAssure Core Console.

To set credentials for an Exchange Server machine

1. Once you have added the Exchange Server machine for protection, navigate to the AppAssure Core Console and select the machine in the Navigation pane.

   The Summary tab displays for the machine.

2. From the Summary tab, in the **Actions** drop-down menu, click **Exchange**, and from the context-sensitive drop-down menu, select the action you want to perform.

3. To set credentials for a single Exchange server, click **Set Credentials**, and in the Edit Exchange Credentials dialog box, do the following:

   a. In the User name text field, enter the user name for a user with permissions to the Exchange server; for example, Administrator (or, if the machine is in a domain, [domain name]\Administrator).

   b. In the Password text field, enter the password associated with user name you specified to connect to the Exchange server.

   c. Click **OK** to confirm the settings and close the dialog box.

Setting credentials for a SQL Server machine

Once you protect data on a SQL Exchange server, you can set login credentials in the AppAssure Core Console.

To set credentials for a SQL server machine

1. Once you have added the SQL Server machine for protection, from the AppAssure Core Console, select the machine in the Navigation pane.

   The Summary tab displays for the machine.

2. From the Summary tab, in the **Actions** drop-down menu, click, and from the context-sensitive drop-down menu, select the action you want to perform.

   - If you want to set default credentials for all SQL Server database instances, click **Set Default Credentials for All Instances**, and in the Edit Default Credentials dialog box, do the following:

     a. In the User name text field, enter the user name for a user with permissions to all associated SQL servers; for example, Administrator (or, if the machine is in a domain, [domain name]\Administrator).

     b. In the Password text field, enter the password associated with user name you specified to connect to the SQL server.

     c. Click **OK** to confirm the settings and close the dialog box.

- If you want to set credentials for a single SQL Server database instance, click **Set Instance Credentials**, and in the Edit Instance Credentials dialog box, do the following:
    a Select the credential type (Default, Windows, or SQL)
    b In the User name text field, enter the user name for a user with permissions to the SQL server; for example, Administrator (or, if the machine is in a domain, [domain name]\Administrator).
    c In the Password text field, enter the password associated with user name you specified to connect to the SQL server.
    d Click **OK** to confirm the settings and close the dialog box.

Understanding recovery point status indicators

Once a recovery point is created on a protected SQL or Exchange server, the application displays a corresponding color status indicator in the Recovery Points table. The color that displays is based on the check settings for the protected machine and the success or failure of those checks, as described in the following Recovery Status Point Colors for SQL Databases and Recovery Status Point colors for Exchange Database tables.

NOTE: For more information on viewing Recovery Points, see Viewing recovery points.

Recovery status point colors for SQL databases

The following table lists the status indicators that display for SQL databases.

Table 57. SQL database status indicators

Status Color	Description
White	Indicates that one of the following conditions exist: • An SQL database did not exist, • Attachability checks were not enabled, or • Attachability checks have not yet been run.
Yellow	Indicates that the SQL database was offline and a check was not possible.
Red	Indicates that the attachability check failed.
Green	Indicates that the attachability check passed.

Recovery status point colors for Exchange databases

The following table lists the status indicators that display for Exchange databases.

Table 58. Exchange database status indicators

Status Color	Description
White	Indicates that one of the following conditions exist: • An Exchange database did not exist, or • Mountability checks were not enabled. **NOTE:** This can apply to certain volumes within a recovery point.
Yellow	Indicates that the Exchange database mountability checks are enabled, but the checks have not yet been run.
Red	Indicates that either the mountability or checksum checks failed on at least one database.
Green	Indicates that the mountability check passed or that the checksum check passed.

NOTE: Recovery points that do not have an Exchange or SQL database associated with it will appear with a white status indicator. In situations where both an Exchange and SQL database exists for the recovery point, the most severe status indicator displays for the recovery point.

Modifying Exchange Server settings

If you are protecting data from a Microsoft Exchange server, you need to configure additional settings in the AppAssure Core Console.

To modify Exchange Server settings

1 Once you have added the Exchange Server machine for protection, navigate to the AppAssure Core Console and select the machine in the Navigation pane.

The Summary tab displays for the machine.

2 From the Summary tab, in the **Actions** drop-down menu, click **Exchange**, and from the context-sensitive drop-down menu, select the action you want to perform.

- If you want to truncate Exchange server logs, click **Force Log Truncation**.
- If you want to set credentials for a single Exchange server, click **Set Credentials**, and in the Edit Exchange Credentials dialog box, do the following:

  a In the User name text field, enter the user name for a user with permissions to the Exchange server; for example, Administrator (or, if the machine is in a domain, [domain name]\Administrator).

  b In the Password text field, enter the password associated with user name you specified to connect to the Exchange server.

  c Click **OK** to confirm the settings and close the dialog box.

Modifying SQL Server settings

If you are protecting data from Microsoft SQL Server, there are additional settings you need to configure in the AppAssure Core Console.

To modify SQL Server settings

1 Once you have added the SQL Server machine for protection, from the AppAssure Core Console, select the machine in the Navigation pane.

The Summary tab displays for the machine.

2 From the Summary tab, in the **Actions** drop-down menu, click, and from the context-sensitive drop-down menu, select the action you want to perform.

- If you want to truncate SQL or Exchange server logs, click **Force Log Truncation**.
- If you want to set default credentials for all SQL Server database instances, click **Set Default Credentials for All Instances**, and in the Edit Default Credentials dialog box, do the following:

  a In the User name text field, enter the user name for a user with permissions to all associated SQL servers; for example, Administrator (or, if the machine is in a domain, [domain name]\Administrator).

  b In the Password text field, enter the password associated with user name you specified to connect to the SQL server.ke as described in the following table.

  c Click **OK** to confirm the settings and close the dialog box.
- If you want to set credentials for a single SQL Server database instance, click **Set Instance Credentials**, and in the Edit Instance Credentials dialog box, do the following:

a Select the credential type (Default, Windows, or SQL)

b In the User name text field, enter the user name for a user with permissions to the SQL server; for example, Administrator (or, if the machine is in a domain, [domain name]\Administrator).

c In the Password text field, enter the password associated with user name you specified to connect to the SQL server.

d Click **OK** to confirm the settings and close the dialog box.

Managing SQL attachability and log truncation

The SQL attachability configuration enables the AppAssure Core to attach SQL database and log files in a snapshot of a SQL server using a local instance of Microsoft SQL Server. The attachability test lets the Core check for the consistency of the SQL databases and ensures that all data files (MDF and LDF files) are available in the backup snapshot. Attachability checks can be run on demand for specific recovery points or as part of a nightly job.

Attachability requires a local instance of Microsoft SQL Server on the AppAssure Core machine. This instance must be a fully licensed version of SQL Server procured from Microsoft or through a licensed reseller. Microsoft does not allow the use of passive SQL licenses.

Attachability supports SQL Server 2005, 2008, 2008 R2, 2012, and 2014. The account used to perform the test must be granted the sysadmin role on the SQL Server instance.

The SQL Server on-disk storage format is the same in both 64-bit and 32-bit environments and attachability works across both versions. A database that is detached from a server instance that is running in one environment can be attached on a server instance that runs in another environment.

Log truncation identifies the free space available in the SQL database logs, but does not minimize them. You can schedule log truncation to occur with the nightly jobs, or you can force it on demand. To force log truncation, see Forcing log truncation for a SQL machine.

NOTE: The version of SQL Server on the Core must be equal to or newer than the SQL Server version on all of the agent machines with SQL Server installed.

This section includes the following topics:

- Configuring SQL attachability settings
- Configuring nightly SQL attachability checks and log truncation for all protected machines

For more information about managing protected machines that use SQL Server, see Modifying SQL Server settings or Customizing nightly jobs for a protected machine.

Configuring SQL attachability settings

Prior to running attachability checks on protected SQL databases, you must first select a local instance of SQL Server on the Core machine that will be used to perform the checks against the agent machine.

NOTE: Attachability requires a local instance of Microsoft SQL Server on the AppAssure Core machine. This instance must be a fully licensed version of SQL Server procured from Microsoft or through a licensed reseller. Microsoft does not allow the use of passive SQL licenses.

Complete the steps in this procedure to configure SQL attachability settings.

To configure SQL attachability settings

1 Navigate to the AppAssure Core, and then click the Configuration tab.

2 Click **Settings**.

3 In the Nightly Jobs pane, click **change**.

The Nightly Jobs dialog box appears.

4 Select **Attachability Check Job** and then click **Settings**.

The Configuration dialog box appears, letting you choose the local SQL Server instance to use for performing attachability checks for the protected SQL Server databases.

5 Use the drop-down menus to select the instance of SQL Server installed on the Core from the following options:

- SQL Server 2005
- SQL Server 2008
- SQL Server 2008 R2
- SQL Server 2012
- SQL Server 2014

NOTE: The options that appear in this drop-down list are populated based on the local SQL Server instances in your environment.

6 Select the credential type. You can select from:

- Windows, or
- SQL

7 Specify the credentials with administrative privileges for the Windows or SQL Server instances, as described in the following table.

Table 59. SQL Server administrator credentials

Text Box	Description
User Name	Enter a user name for logon permissions to the SQL server.
Password	Enter a password for SQL attachability. It is used to control logon activity.

8 Click **Test Connection**.

NOTE: If you entered the credentials incorrectly, a message displays to alert you that the credentials test failed. Correct the credential information and run the connection test again.

9 Click **Save**.

Attachability checks are now available to be run on the protected SQL Server databases.

10 In the Nightly Jobs window, click **OK**.

Attachability checks are now schedule to occur with the nightly jobs.

Configuring nightly SQL attachability checks and log truncation for all protected machines

You can view, enable, or disable SQL database server settings, including attachability check and nightly log truncation from the Nightly Jobs dialog box accessed from the Core. Changes made here apply to all SQL machines protected by the Core.

Complete the steps in this procedure to have the system perform nightly attachability checks for the SQL Server recovery points.

To configure nightly SQL attachability checks and log truncation

1. Navigate to the AppAssure Core, and then click the Configuration tab.
2. Click **Settings**.
3. In the Nightly Jobs section, click **Change**.
4. Select or clear the SQL Server settings based on the needs of your organization:
   - Attachability Check Job
   - Log Truncation Job (simple recovery model only)
5. Click **OK**.

   The attachability and log truncation settings take effect for the protected SQL Server.

Forcing a SQL Server attachability check

Complete the steps in this procedure to force the system to perform an attachability check for a specific SQL server recovery point.

NOTE: To have the ability to force an attachability check, a SQL database must be present on a protected volume. If AppAssure does not detect the presence of a database, the attachability check function does not appear in the Core Console.

To force a SQL Server attachability check

1. In the left navigation area of the AppAssure Core, select the machine for which you want to force the attachability check and click the **Recovery Points** tab.
2. Click the right angle bracket **>** symbol next to a recovery point in the list to expand the view.
3. Click **Check**, and then click **Force Attachability Check**.

   The Force Attachability Check window appears for you to indicate that you want to force an attachability check.
4. Click **Yes**.

   The system performs the attachability check.

   NOTE: For information on how to view the status of the attachability checks, see Viewing tasks, alerts, and events.

Forcing log truncation for a SQL machine

Log truncation is available for machines that use SQL Server. Complete the steps in this procedure to force log truncation.

NOTE: When conducted for a SQL machine, truncation identifies the free space on a disk, but does not reduce the size of the logs.

To force log truncation for a SQL machine

1. On the AppAssure Core Console, navigate to the protected machine for which you want to force log truncation.
   - From the Summary tab of the protected machine, click the **Actions** drop-down menu, select **SQL**, and then click **Force Log Truncation**.
2. Click **Yes** to confirm that you want to force log truncation.

Managing Exchange database mountability checks and log truncation

When using AppAssure to back up Microsoft Exchange Servers, mountability checks can be performed on all Exchange databases after every snapshot. This corruption detection feature alerts administrators of potential failures and ensures that all data on the Exchange servers will be recovered successfully in the event of a failure.

Log truncation minimizes the size of the logs from an Exchange database on a daily basis when scheduled to occur with the nightly jobs. For information about forcing log truncation, see Forcing log truncation for an Exchange machine.

NOTE: The mountability checks only apply to Microsoft Exchange 2007, 2010, and 2013. Additionally, the AppAssure Agent service account must be assigned the Organizational Administrator role in Exchange.

Configuring nightly Exchange database checksum checks and log truncation

You can view, enable, or disable Exchange database server settings, including automatic mountability check, nightly checksum check, or nightly log truncation, from the Core-level Configuration tab. Changes made to the settings on this tab apply to all Exchange machines protected by the Core.

Complete the steps in this procedure to configure settings for Exchange database mountability and log truncation.

To configure Exchange database mountability and log truncation

1. Navigate to the AppAssure Core, and then click the Configuration tab.
2. Click **Settings**.
3. In the Nightly Jobs section, click **Change**.
4. Select or clear the Exchange Server settings based on the needs of your organization:
   - Checksum Check Job
   - Truncate Exchange logs
5. Click **OK**.

   The checksum and log truncation settings take effect for the protected Exchange Server.

   NOTE: For information on forcing log truncation, see Forcing log truncation for an Exchange machine.

Forcing a mountability check of an Exchange database

Complete the steps in this procedure to force the system to perform a mountability check for a specific Exchange server recovery point.

NOTE: To have the ability to force a mountability check, an Exchange database must be present on a protected volume. If AppAssure does not detect the presence of a database, the mountability check function does not appear in the Core Console.

To force a mountability check

1. In the left navigation area of the AppAssure Core Console, select the machine for which you want to force the mountability check, and then click the **Recovery Points** tab.
2. Click the right angle bracket > symbol next to a recovery point in the list to expand the view.
3. Click **Check**, and then click **Force Mountability Check**.

   A pop-up window displays asking if you want to force a mountability check.
4. Click **Yes**.

   The system performs the mountability check.

   NOTE: For instructions on how to view the status of the attachability checks, see Viewing tasks, alerts, and events.

Forcing a checksum check of Exchange Server recovery points

Complete the steps in this procedure to force the system to perform a checksum check for a specific Exchange server recovery point.

NOTE: To have the ability to force a checksum check, an Exchange database must be present on a protected volume. If AppAssure does not detect the presence of a database, the checksum check function does not appear in the Core Console.

To force a checksum check of Exchange Server recovery points

1. In the left navigation area of the AppAssure Core Console, select the machine for which you want to force the checksum check, and then click the **Recovery Points** tab.
2. Click the right angle bracket > symbol next to a recovery point in the list to expand the view.
3. Click **Check**, and then click **Force Checksum Check**.

   The Force Attachability Check window appears for you to indicate that you want to force a checksum check.
4. Click **Yes**.

   The system performs the checksum check.

   NOTE: For information on how to view the status of the attachability checks, see Viewing tasks, alerts, and events.

Forcing log truncation for an Exchange machine

Log truncation is available for machines that use Exchange Server. When log truncation is conducted for an Exchange machine, the size of the logs are reduced. Complete the steps in this procedure to force log truncation.

To force log truncation for an Exchange machine

1. On the AppAssure Core Console, navigate to the protected machine for which you want to force log truncation.
   - From the Summary tab of the protected machine, click the **Actions** drop-down menu, select **Exchange**, and then click **Force Log Truncation**.
2. Click **Yes** to confirm that you want to force log truncation.

8

Protecting server clusters

This chapter describes how to protect information on Microsoft SQL Server or Exchange Server clusters using AppAssure. It includes the following topics:

- Supported applications and cluster types
- Protecting a cluster
- Protecting nodes in a cluster
- Modifying cluster node settings
- Configuring cluster settings
- Converting a protected cluster node to a protected machine
- Viewing server cluster information
- Working with cluster recovery points
- Managing snapshots for a cluster
- Performing a restore for clusters and cluster nodes
- Replicating cluster data
- Removing a cluster from protection
- Removing cluster nodes from protection
- Viewing a cluster or node report

In AppAssure, server cluster protection is associated with the AppAssure protected machines installed on individual cluster nodes (that is, individual machines in the cluster) and the AppAssure Core, which protects those machines, all as if they were one composite machine.

You can easily configure an AppAssure Core to protect and manage a cluster. In the Core Console, a cluster is organized as a separate entity, which acts as a container that includes the related nodes. For example, in the left navigation area, under the Protected Machines menu, protected clusters are listed. Directly below each cluster, the associated individual nodes or agent machines appear. Each of these is a protected machine on which the AppAssure Agent software is installed. If you click on the cluster, the tab appears in the Core Console

At the Core and cluster levels, you can view information about the cluster, such as the list of related nodes and shared volumes. A cluster appears in the Core Console on the Protected Nodes tab, and you toggle the view (using Show/Hide) to view the nodes included in the cluster. At the cluster level, you can also view corresponding Exchange and SQL cluster metadata for the nodes in the cluster. You can specify settings for the entire cluster and the shared volumes in that cluster, or you can navigate to an individual node (machine) in the cluster to configure settings just for that node and the associated local volumes.

Supported applications and cluster types

To protect your cluster properly, you must have installed the AppAssure Agent software on each of the machines or nodes in the cluster. AppAssure supports the application versions and cluster configurations listed in the following table.

Table 60. Supported application versions and cluster configurations

Application	Application Version and Related Cluster Configuration	Windows Failover Cluster
Microsoft Exchange Server	2007 Single Copy Cluster (SCC) 2007 Cluster Continuous Replication (CCR)	2003, 2008, 2008 R2
	2010 Database Availability Group (DAG)	2008, 2008 R2
	2013 Database Availability Group (DAG)	2008 R2 SP1, 2012, 2012 R2
Microsoft SQL Server	2005	2003, 2008, 2008 R2
	2008, 2008 R2 Single Copy Cluster (SCC)	2003, 2008, 2008 R2, 2012, 2012 R2
	2012, 2014 Single Copy Cluster (SCC)	2008, 2008 R2, 2012, 2012 R2
	2012, 2014 Availability Groups	Server 2012, 2012 R2

The supported disk types include:

- GUID partition table (GPT) disks greater than 2 TB
- Basic disks

The supported mount types include:

- Shared drives that are connected as drive letters (for example, D:)
- Simple dynamic volumes on a single physical disk (not striped, mirrored, or spanned volumes)
- Shared drives that are connected as mount points

Limited support for cluster shared volumes

AppAssure cluster shared volumes (CSV) support applies only to native backup of CSVs on Windows 2008 R2 agents only.

For other operating systems, the agent service can be run on all nodes in a cluster, and the cluster can be protected as a cluster within the AppAssure Core; however, CSVs do not display in the Core Console and are not available for protection. All local disks (such as the operating system volume) are available for protection.

The following table depicts current support in AppAssure core for cluster shared volumes.

Table 61. Compatibility for cluster shared volumes in AppAssure

AppAssure Cluster Shared Volumes Support	Protect, Replicate, Rollup, Mount Archive		Restore CSV Volumes		Virtual Export to Hyper-V CSV	
AppAssure version	5.3	5.4	5.3	5.4	5.3	5.4
Windows 2008 R2	Yes	Yes	No	Yes	Yes	Yes
Windows 2012	No	No	No	No	No	No
Windows 2012 R2	No	No	No	No	No	No

Protecting a cluster

This topic describes how to add a cluster for protection in AppAssure. When you add a cluster to protection, you need to specify the host name or IP address of the cluster, the cluster application, or one of the cluster nodes or machines that includes the AppAssure Agent software.

NOTE: A repository is used to store the snapshots of data that are captured from your protected nodes. Before you start protecting data in your cluster, you should have set up at least one repository that is associated with your AppAssure Core.

For information about setting up repositories, see Understanding repositories.

To protect a cluster

1. In the Core Console, navigate to the Home tab, and then click the **Protect** button drop-down menu and then click **Protect Cluster**.
2. In the Connect to Cluster dialog box, enter the following information, and then click **Connect**.

   Table 62. Connect to Cluster settings

Text Box	Description
Host	The host name or IP address of the cluster, the cluster application, or one of the cluster nodes.
Port	The port number on the machine on which the AppAssure Core communicates with the Agent. The default port is 8006.
User name	The user name of the domain administrator used to connect to this machine: for example, domain_name\administrator or administrator@domain_name.com **NOTE:** The domain name is mandatory. You cannot connect to the cluster using the local administrator user name.
Password	The password used to connect to this machine.

3. In the Protect Cluster dialog box, select a repository for this cluster.
4. If you want to secure the recovery points for this cluster, select an encryption key.
5. If you do not want protection to begin immediately after completing this procedure, select **Initially pause protection**.
6. To protect the cluster based on default settings, select the nodes for default protection, and then skip to Step 8.

   NOTE: The default settings ensure that all volumes are protected with a schedule of every 60 minutes.

7. To enter custom settings for the cluster (for example, to customize the protection schedule for the shared volumes), click **settings** next to the node whose protection you want to customize, and then see Creating custom protection schedules.

   For more information on customizing nodes, see Protecting nodes in a cluster.

8. In the Protect Cluster dialog box, click **Protect**.

Protecting nodes in a cluster

This topic describes how to protect the data on a cluster node or machine that has an AppAssure Agent installed. When you add protection, you need to select a node from the list of available nodes as well as specify the host name and the user name and password of the domain administrator.

To protect nodes in a cluster

1 In the AppAssure Core Console, once you have added a cluster, navigate to the cluster that you want to protect.

The Summary tab for the selected cluster displays.

2 Click the Protected Nodes tab, and then from the **Actions** menu, select **Protect Cluster Node**.

3 In the Protect Cluster Node dialog box, select or enter as appropriate the following information, and then click **Connect** to add the machine or node.

Table 63. Protect Cluster Node settings

Text Box	Description
Host	A drop-down list of nodes in the cluster available for protection.
Port	The port number on which the AppAssure Core communicates with the Agent on the node.
User name	The user name of the domain administrator used to connect to this node; for example, example_domain\administrator or administrator@example_domain.com.
Password	The password used to connect to this machine.

4 Click **Protect** to start protecting this machine with default protection settings.

NOTE: The default settings ensure that all volumes on the machine are protected with a schedule of every 60 minutes.

5 To enter custom settings for this machine, (for example, to change the Display name, add encryption, or customize the protection schedule), click **Show Advanced Options**.

6 Edit the following settings as needed, as described in the following table.

Table 64. Advanced Options settings

Text Box	Description
Display Name	Enter a new name for the machine to be displayed in the Core Console.
Repository	Select the repository on the AppAssure Core in which the data from this machine should be stored.
Encryption	Specify whether encryption should be applied to the data for every volume on this machine to be stored in the repository. **NOTE:** The encryption settings for a repository are defined under the Configuration tab in the AppAssure Core Console.
Schedule	Select one of the following options. • Protect all volumes with default schedule • Protect specific volumes with custom schedule. Then, under Volumes, select a volume and click **Edit**. For more information on setting custom intervals, see Step 7 in Protecting a cluster.

Modifying cluster node settings

Once you have added protection for cluster nodes, you can easily modify basic configuration settings for those machines or nodes (for example, display name, host name, and so on), protection settings (for example, changing the protection schedule for local volumes on the machine, adding or removing volumes, and pausing protection), and more.

To modify cluster node settings, you must perform the following tasks:

1 In the AppAssure Core Console, navigate to the cluster that contains the node you want to modify, and select the machine or node that you want to modify.

2 To modify and view configuration settings, see Configuring notification groups for system events.

3 To configure notification groups for system events, see Viewing and modifying configuration settings.

4 To customize retention policy settings, see Customizing retention policy settings for a protected machine.

5 To modify the protection schedule, see Modifying protection schedules.

6 To modify transfer settings, see Modifying transfer settings.

Configuring cluster settings

Configuring cluster settings involves performing the following tasks:

- **Modify cluster settings.** For more information about modifying cluster settings, see Modifying cluster settings.
- **Configure cluster event notifications.** For more information about configuring cluster event notifications, see Configuring cluster event notifications.
- **Modify the cluster retention policy.** For more information about modifying the cluster retention policy, see Modifying the cluster retention policy.
- **Modify the cluster protection schedules.** For more information about modifying the cluster protection schedules, see Modifying cluster protection schedules.
- **Modify the cluster transfer settings.** For more information about modifying cluster transfer settings, see Modifying cluster transfer settings.

Modifying cluster settings

Once you have added a cluster, you can easily modify basic settings (for example, display name), protection settings (for example, protection schedules, adding or removing volumes, and pausing protection), and more.

To modify cluster settings

1 In the AppAssure Core Console, navigate to the cluster that you want to modify,

2 Click the Configuration tab.

The Settings page displays.

3 Click **Edit** to modify the settings on this page for the cluster as described in the following table.

Table 65. Cluster settings

Text Box	Description
Display Name	Enter a display name for the cluster. The name for this cluster displays in the AppAssure Core Console. By default, this is the host name for the cluster. You can change this to something more descriptive, if needed.
Repository	Enter the Core repository associated with the cluster. NOTE: If snapshots have already been taken for this cluster, this setting is listed here for informational purposes only and cannot be modified.
Encryption Key	Edit and select an encryption key if necessary. This specifies whether encryption should be applied to the data for every volume on this cluster to be stored in the repository.

Configuring cluster event notifications

You can configure how system events are reported for your cluster by creating notification groups. These events could be system alerts or errors. Complete the steps in this procedure to configure notification groups for events.

To configure cluster event notifications

1 In the AppAssure Core Console, navigate to the cluster that you want to modify,

2 Click the Configuration tab, and then click **Events**.

3 Select one of the options described in the following table.

Table 66. Event notification options

Option	Description
Use Core alert settings	This adopts the settings used by the associated core: • Click **Apply** and then perform Step 5.
Use Custom alert settings	This lets you configure custom settings: • Proceed to Step 4.

4 If you selected Custom alert settings, do the following:

a Click **Add Group** to add a new notification group for sending a list of system events.

The Add Notification Group dialog box opens.

b Add the notification options as described in the following table.

Table 67. Event notification settings

Text Box	Description
Name	Enter a name for the notification group.
Description	Enter a description for the notification group.

Table 67. Event notification settings

Text Box	Description
Enable Events	Select the events for notification, for example, Clusters. You can also choose to select by type: • Error • Warning • Info NOTE: When you choose to select by type, by default, the appropriate events are automatically enabled. For example, if you choose Warning, the Attachability, Jobs, Licensing, Archive, CoreService, Export, Protection, Replication, and Rollback events are enabled.
Notification Options	Select the method to specify how to handle notifications. You can choose from the following options: • **Notify by Email**. Specify the email addresses to which to send the events in the To, CC, and BCC text boxes. • **Notify by Windows Event log**. The Windows Event log controls the notification. • **Notify by syslogd.** Specify the host name and port to which to send the events. • **Notify by Toast alerts.** Select this option if you want the alert to appear as a pop-up in the lower-right corner of your screen.

c Click **OK** to save your changes, and then click **Apply**.

5 To edit an existing notification group, next to a notification group in the list, click **Edit**.

The Edit Notification Group dialog box displays for you to edit the settings.

Modifying the cluster retention policy

The retention policy for a cluster specifies how long the recovery points for the shared volumes in the cluster are stored in the repository. Retention policies are used to retain backup snapshots for longer periods of time and to help with management of these backup snapshots. The retention policy is enforced by a rollup process that helps in aging and deleting old backups.

To modify the cluster retention policy

1 In the AppAssure Core Console, navigate to the cluster that you want to modify,

2 Click the Configuration tab, and then click **Retention Policy**.

3 Select one of the options in the following table.

Table 68. Retention policy options

Option	Description
Use Core default retention policy	This adopts the settings used by the associated core. • Click **Apply**.
Use Custom retention policy	This lets you configure custom settings.

NOTE: If you selected Custom alert settings, follow the instructions for setting a custom retention policy as described in Customizing nightly jobs for a protected machine.

Modifying cluster protection schedules

In AppAssure, you can modify the protection schedules only if your cluster has shared volumes.

To modify cluster protection schedules

1. In the AppAssure Core Console, navigate to the cluster that you want to modify, and then select the cluster.
2. Follow the instructions for modifying the protection settings as described in Modifying protection schedules.

Modifying cluster transfer settings

In AppAssure, you can modify the settings to manage the data transfer processes for a protected cluster.

NOTE: You can modify cluster transfer settings only if your cluster has shared volumes.

There are three types of transfers in AppAssure:

- **Snapshots.** Backs up the data on your protected cluster.
- **VM Export.** Creates a virtual machine with all of the backup information and parameters as specified by the schedule defined for protecting the cluster.
- **Restore**. Restores backup information for a protected cluster.

To modify cluster transfer settings

1. In the AppAssure Core Console, navigate to the cluster that you want to modify.
2. Click the Configuration tab, and then click **Transfer Settings**.
3. Modify the protection settings as described in Modifying protection schedules.

Converting a protected cluster node to a protected machine

In AppAssure, you can convert a protected cluster node to a protected machine so that it is still managed by the Core, but it is no longer part of the cluster. This is helpful, for example, if you need to remove the cluster node from the cluster but still keep it protected.

To convert a protected cluster node to a protected machine

1. In the AppAssure Core Console, navigate to the cluster that contains the machine you wish to convert, and then click **Protected Nodes**.
2. On the Protected Nodes page, from the specific node you want to convert, click the **Actions** drop-down menu and select **Convert to Agent**.
3. To add the machine back to the cluster, select the machine, and then on the **Summary** tab, from the Actions menu, select **Convert to Cluster Node**, and then click **Yes** to confirm the action.

Viewing server cluster information

Complete the steps in the following procedures to view summary, event, alert information, and so on for server clusters.

Viewing cluster system information

Complete the steps in this procedure to view detailed system information about a cluster.

To view cluster system information

1. In the AppAssure Core Console, navigate to the cluster that you want to view.
2. Click the Tools tab.

   The system information page displays to show system details about the cluster such as name, included nodes with associated state and Windows versions, network interface information, and volume capacity information.

Viewing cluster tasks, events and alerts

Complete the steps in this procedure to view events and alerts for a cluster.

For information about viewing events and alerts for an individual machine or node in a cluster, see Viewing tasks, alerts, and events.

To view cluster tasks, events and alerts

1. In the AppAssure Core Console, navigate to the cluster that you want to view.
2. Click the Events tab, which opens to show all tasks for this cluster.
3. You can filter the tasks that display for this cluster. For more information, see Viewing tasks.
4. To view only alerts, at the top left-hand side of the page, click **Alerts**.

   The list of events is filtered to display only alerts for the cluster or node you selected.
5. Optionally, if you want to remove all alerts from the page, click **Dismiss All**.
6. To view all events, at the top left-hand side of the page, click **Events**.

   All events display for the cluster or node you selected.

Viewing summary information

Complete the steps in this procedure to view summary information about a cluster including information about the associated quorum for the cluster.

To view summary information

1. In the AppAssure Core Console, navigate to the cluster that you want to view.
2. On the Summary tab, you can view such information as the cluster name, cluster type, quorum type (if applicable), and the quorum path (if applicable). This tab also shows at-a-glance information about the volumes in this cluster, including size and protection schedule. If applicable, you can also view SQL Server or Exchange Server information for a different cluster.
3. To refresh this information to the most current, click the **Actions** drop-down menu, and click **Refresh Metadata**.

For information about viewing summary and status information for an individual machine or node in the cluster, see Viewing machine status and other details.

Working with cluster recovery points

A recovery point, also referred to as a snapshot, is a point-in-time copy of the folders and files for the shared volumes in a cluster, which are stored in the repository. Recovery points are used to recover protected machines or to mount to a local file system. In AppAssure, you can view the lists of recovery points in the repository. Complete the steps in the following procedure to review recovery points.

NOTE: If you are protecting data from a DAG or CCR server cluster, the associated recovery points do not appear at the cluster level. They are only visible at the node or machine level.

For information about viewing recovery points for individual machines in a cluster, see Viewing recovery points.

To work with cluster recovery points

1. In the AppAssure Core Console, navigate to the cluster for which you wish to view recovery points.
2. Click the Recovery Points tab.
3. To view detailed information about a specific recovery point, click the right angle bracket > symbol next to a recovery point in the list to expand the view.
4. For information about the operations you can perform on the recovery points, see Viewing a specific recovery point.
5. Select a recovery point to mount.

   For information about how to mount a recovery point, see Mounting a recovery point.
6. To delete recovery points, see Removing recovery points.

Managing snapshots for a cluster

In AppAssure, you can manage snapshots by forcing a snapshot or by pausing current snapshots. Forcing a snapshot lets you force a data transfer for the currently protected cluster. When you force a snapshot, the transfer starts immediately or will be added to the queue. Only the data that has changed from a previous recovery point transfers. If there is no previous recovery point, all data (the base image) on the protected volumes is transferred. When you pause a snapshot, you temporarily stop all transfers of data from the current machine.

For information about forcing snapshots for the individual machines in a cluster, see Forcing a snapshot. For information about pausing and resuming snapshots for the individual machines in a cluster, see Pausing and resuming protection.

Forcing a snapshot for a cluster

Complete the steps in this procedure to force a snapshot for a cluster.

To force a snapshot for a cluster

1. In the AppAssure Core Console, navigate to the cluster for which you wish to view recovery points.
2. On the Summary tab, click the **Actions** drop-down menu, and then click **Force Snapshot**.

Pausing and resuming cluster snapshots

Complete the steps in this procedure to pause and resume a snapshot for a cluster.

To pause and resume cluster snapshots

1. In the AppAssure Core Console, navigate to the cluster for which you wish to view recovery points.
2. On the Summary tab, click the **Actions** drop-down menu, and then click **Pause Snapshots**.
3. In the Pause Protection dialog box, select one of the options described in the following table.

Table 69. Pause Protection options

Option	Description
Pause until resumed	Pauses the snapshot until you manually resume protection. • To resume protection, click the **Actions** menu and then click **Resume**.
Pause for	Lets you specify an amount of time in days, hours, and minutes to pause snapshots.

Performing a restore for clusters and cluster nodes

A restore is the process of restoring the volumes on a machine from recovery points. For a server cluster, you perform a restore at the node, or machine, level. This section provides guidelines for performing a restore for cluster volumes.

Performing a restore for CCR and DAG (Exchange) clusters

Complete the steps in this procedure to perform a restore for CCR and DAG (Exchange) clusters.

To perform a restore for CCR and DAG (Exchange) clusters

1. Turn off all nodes except one.
2. Perform a restore using the standard AppAssure procedure for the machine as described in Restoring volumes from a recovery point and Restoring volumes for a Linux machine using the command line.
3. When the restore is finished, mount all databases for the cluster volumes.

4 Turn on all other nodes.

5 For Exchange, navigate to the Exchange Management Console, and, for each database, perform the **Update Database Copy** operation.

Performing a restore for SCC (Exchange, SQL) clusters

Complete the steps in this procedure to perform a restore for SCC (Exchange, SQL) clusters.

To perform a restore for SCC (Exchange, SQL) clusters

1 Turn off all nodes except one.

2 Perform a restore using the standard AppAssure procedure for the machine as described in Restoring volumes from a recovery point and Restoring volumes for a Linux machine using the command line.

3 After the restore is finished, mount all databases from the cluster volumes.

4 Turn on all other nodes one-by-one.

> **NOTE:** You do not need to roll back the quorum disk. It can be regenerated automatically or by using cluster service functionality.

Replicating cluster data

When you replicate data for a cluster, you must replicate the entire cluster. For example, if you select a node to replicate, the cluster is automatically selected; likewise, if you select the cluster, all nodes in that cluster are also selected.

For more information and instructions on replicating data, see Configuring replication.

Removing a cluster from protection

Complete the steps in the following procedure to remove a cluster from protection.

To remove a cluster from protection

1 In the AppAssure Core Console, navigate to the cluster you wish to remove.

2 Click the **Actions** drop-down menu, and then click **Remove Cluster**.

3 Select one of the following options.

Table 70. Remove Cluster options

Option	Description
Keep Recovery Points	To keep all currently stored recovery points for this cluster.
Remove Recovery Points	To remove all currently stored recovery points for this cluster from the repository.

Removing cluster nodes from protection

Complete the steps in the following procedures to remove cluster nodes from protection.

If you just want to remove a node from the cluster, see Converting a protected cluster node to a protected machine.

To remove a cluster node from protection

1. In the AppAssure Core Console, navigate to the cluster Protected Nodes tab.
2. Click the Actions drop-down menu and then click **Remove Node**.
3. Select one of the options described in the following table.

Table 71. Remove Node options

Option	Description
Keep Recovery Points	To keep all currently stored recovery points for this machine or node.
Remove Recovery Points	To remove all currently stored recovery points for this machine or node from the repository.

Removing all nodes in a cluster from protection

Complete the steps in this procedure to remove all nodes in a cluster from protection.

△ | **CAUTION: If you remove all cluster nodes, the cluster is also removed.**

To remove all nodes in a cluster from protection

1. In the AppAssure Core Console, navigate to the cluster Protected Nodes tab.
2. On the Protected Nodes tab, select all of the nodes.
3. Click the Actions drop-down menu at the top of the Protected Nodes tab, click **Remove Nodes**, and then select one of the options described in the following table.

Table 72. Remove Nodes options

Option	Description
Keep Recovery Points	To keep all currently stored recovery points for this cluster.
Remove Recovery Points	To remove all currently stored recovery points for this cluster from the repository.

Viewing a cluster or node report

You can create and view compliance and errors reports about AppAssure activities for your cluster and individual nodes. The reports include AppAssure activity information about the cluster, node, and shared volumes.

For more information about AppAssure reporting, see Generating and viewing reports. For more information about the exporting and printing options located in the reports toolbar, see Using the reports toolbar.

To view a cluster or node report

1 In the AppAssure Core Console, navigate to the cluster for which you want to create a report.

2 If you want to create a report for a node under a cluster, select the node.

3 Click the Tools tab and, under the Reports menu, select one of the following options:

- Compliance Report
- Failure Report

4 In the Start Time drop-down calendar, select a start date, and then enter a start time for the report.

> **NOTE:** No data is available before the time the AppAssure Core or Agent was deployed.

5 In the End Time drop-down calendar, select an end date, and then enter an end time for the report.

6 Click **Generate Report**. The report results appear in the page.

If the report spans multiple pages, you can click the page numbers or the arrow buttons at the top of the report results to page through the results.

7 To export the report results to one of the available formats—PDF, XLS, XLSX, RTF, MHT, HTML, TXT, CSV, or image—select the format for export from the drop-down list, and then do one of the following:

- Click the first Save icon to export a report and save it to the disk.
- Click the second Save icon to export a report and show it in a new Web browser window.

8 To print the report results, do one of the following:

- Click the first Printer icon to print the entire report.
- Click the second Printer icon to print the current page of the report.

9

Exporting protected data from Windows machines to virtual machines

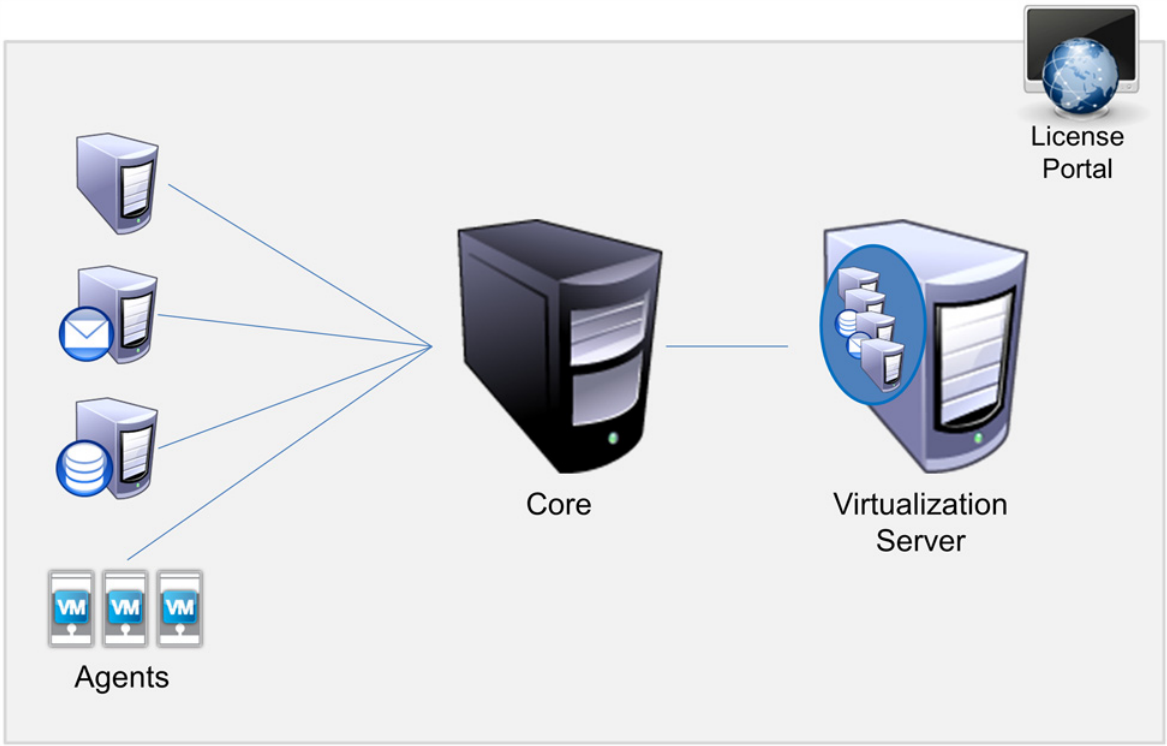
This chapter describes how to export a recovery point to create a virtual machine. It includes the following topics:

- Managing exports
- Exporting data to a Windows-based virtual machine
- Exporting data to an ESXi virtual machine
- Exporting data to a VMware Workstation virtual machine
- Exporting data to a Hyper-V virtual machine
- Exporting data to a VirtualBox virtual machine

AppAssure supports both a one-time export or continual export (to support virtual standby) of Windows backup information to a virtual machine. Exporting your data to a virtual standby machine provides you with a high availability copy of the data. If a protected machine goes down, you can boot up the virtual machine to then perform recovery.

The following diagram shows a typical deployment for exporting data to a virtual machine.

Figure 4. Virtual standby deployment



When you export to a virtual machine, all of the backup data from a recovery point as well as the parameters defined for the protection schedule for your machine will be exported. You can also create a “virtual standby” by having protected data continually exported from your protected machine to a virtual machine.

You can perform virtual export of recovery points for your protected Windows or Linux machines to VMware, ESXi, Hyper-V, and VirtualBox.

NOTE: For ESXi, VMware Workstation, or Hyper-V, the virtual machine version must be a licensed version of these virtual machines and not the trial or free versions.

If you have replication set up between two cores (source and target), you can only export data from the target core after the initial replication is complete.

Managing exports

Virtual Standby is a physical-to-virtual (P2V) process that creates a virtual machine or clone of a protected machine. A virtual standby can be created using a one-time or a continual update export process. A virtual standby created using a continual update is incrementally updated after every snapshot captured from the source machine.

NOTE: AppAssure supports Hyper-V export to Window 8, Window 8.1, Windows Server 2012 and 2012 R2.

On the Virtual Standby tab in the Core Console, you can view the status of exports that you currently have set up, including one-time exports and continual exports for virtual standby. On this tab, you can perform a variety of actions to manage exports, including pausing, stopping, or removing exports. You can also view a queue of upcoming exports.

To manage exports

1 On the Core Console, navigate to the Virtual Standby tab.

On the Virtual Standby tab you can view a table of saved export settings, which includes the information described in the following table.

Table 73. Virtual Standby export settings

Column	Description
Status	The status of the virtual standby configuration, displayed as an icon. The following status icons may appear: • Green icon - The Virtual Standby is successfully configured, is active, and is not paused. The next Virtual Standby export will be performed just after the next snapshot. • Yellow icon - The virtual standby is paused and is still saved by the Core. However, after a new transfer, the export job will not start automatically and there will be no new Virtual Standby exports for this protected machine.
Machine Name	The name of the source machine.
Destination	The virtual machine and path to which data is being exported.
Export Type	The type of virtual machine platform for the export, such as, ESXi, VMware, Hyper-V, or VirtualBox.
Last Export	The date and time of the last export. If an export has just been added but has not completed, a message will display stating the export has not yet been performed. If an export has failed or was cancelled, a corresponding message also will display.

2 To manage saved export settings, select an export, and then click one of the following:

- **Pause** - To pause the export.
- **Resume** - To restart a paused export.
- **Force** - To force a new export. This option could be helpful when virtual standby is paused and then resumed, which means the export job will restart only after a new transfer. If you do not want to wait for the new transfer, you could force an export.

3 To remove an export from the system, click **Remove**. When you remove an export, it is permanently removed from the system and you will not be able to re-start it.

4 To view details about the active exports currently in queue to be completed, click **Show Export Queue**.

The Export Queue table displays under the Virtual Standby table and includes the information described in the following table.

Table 74. Export Queue table information

Column	Description
Machine Name	The name of the source machine.
Destination	The virtual machine and path to which data is being exported.
Export Type	The type of virtual machine platform for the export, such as, ESXi, VMware, Hyper-V, or VirtualBox.
Schedule Type	The type of export as either One-time or Continuous.
Status	The progress of the export, displayed as a percentage in a progress bar.

5 To manage the number of exports that can execute at the same time, do the following:

- In the Export Queue table, click **Max Concurrent Exports**.
- In the Max Concurrent Exports dialog box, enter a number and click **Save**. The default is 5.

6 To cancel an export in the Export Queue, select an export in the Export Queue table, and then click **Cancel**.

7 To add a new virtual standby export, you can click **Add** to launch the Export Wizard. For further information about setting up virtual standby for a specific virtual machine, see one of the following topics:

- Exporting data to an ESXi virtual machine
- Exporting data to a VMware Workstation virtual machine
- Exporting data to a Hyper-V virtual machine
- Exporting data to a VirtualBox virtual machine
- Exporting data to a Linux-based VirtualBox virtual machine

Exporting data to a Windows-based virtual machine

In AppAssure you can export data from your Windows machines to a virtual machine (VMware, ESXi, Hyper-V, and VirtualBox) by exporting all of the backup information from a recovery point as well as the parameters defined for the protection schedule for your machine.

To export Windows backup information to a virtual machine

- In the AppAssure Core Console, navigate to the machine you want to export. On the Summary tab, in the **Actions** drop-down menu, click **Export**, and then select the type of export you want to perform as either One-time or Virtual Standby. The Export Wizard displays.

Proceed to the following topics for more information about exporting Windows data to a specific type of virtual machine.

- Exporting data to an ESXi virtual machine
- Exporting data to a VMware Workstation virtual machine
- Exporting data to a Hyper-V virtual machine
- Exporting data to a VirtualBox virtual machine

Exporting data to an ESXi virtual machine

In AppAssure, you can choose to export data to ESXi by performing a one-time export, or by establishing a continual export (for virtual standby). Complete the steps in the following procedures for the appropriate type of export.

Performing a one-time ESXi export

Complete the steps in this procedure to perform a one-time export to ESXi.

To perform a one-time ESXi export

1 In the AppAssure Core Console, do one of the following:

- From the button bar, click **Export** to launch the Export Wizard, and do the following:
  - a On the Select Export Type page, select **One-time export**, and then click **Next**.
  - b On the Protected Machines page, select the protected machine you want to export to a virtual machine, and then click **Next**.
- Navigate to the machine you want to export, and then in the Summary tab, from the Actions drop-down menu for that machine, select **Export** > **One-time**.

  The Export Wizard appears on the Recovery Points page.

2 On the Recovery Points page, select the recovery point from the AppAssure Core that you want to export, and then click **Next**.

Defining virtual machine information for performing an ESXi export

Complete the steps in this procedure to define the information for the virtual machine.

To define virtual machine information for performing an ESXi export

1 On the Destination page in the Export Wizard, in the Recover to Virtual machine drop-down menu, select **ESX(i)**.

2 Enter the parameters for accessing the virtual machine as described in the following table, and then click **Next**.

Table 75. Virtual machine parameters

Options	Description
Host name	Enter a name for the host machine.
Port	Enter the port for the host machine. The default is 443.
User name	Enter the logon credentials for the host machine.
Password	Enter the logon credentials for the host machine.

3 On the Virtual Machine Options page, enter the information described in the following table.

Table 76. Virtual machine options

Option	Description
Resource Pool	Select a resource pool from the drop-down list.
Data Store	Select a data store from the drop-down list.
Virtual Machine Name	Enter a name for the Virtual Machine.
Memory	Specify the memory usage for the virtual machine by clicking one of the following: • **Use the same amount of RAM as source machine** • **Use a specific amount of RAM, and then specify the amount in MB** The minimum amount is 1024 MB and the maximum allowed by the application is 65536 MB. The maximum amount of memory usage is limited by the amount of RAM available to the host machine.
Disk Provisioning	Select the type of disk provisioning as either Thin or Thick.
Disk Mapping	Specify the type of disk mapping as either Automatic or Manual.
Version	Select the version of the virtual machine.

4 Click **Next**.

5 On the Volumes page, select the volumes you want to export, and then click **Next**.

6 On the Summary page, click **Finish** to complete the wizard and start the export.

NOTE: You can monitor the status and progress of the export by viewing the Virtual Standby or Events tab.

Performing a continual (Virtual Standby) ESXi export

Complete the steps in this procedure to perform a continual export to ESXi.

To perform a continual (virtual standby) ESXi export

1 In the AppAssure Core Console, do one of the following:

- On the Virtual Standby tab, click **Add** to launch the Export Wizard. On the Protected Machines page of the Export Wizard, select the protected machine you want to export, and then click **Next**.
- Navigate to the machine you want to export, and, on the Summary tab in the Actions drop-down menu for that machine, click **Export** > **Virtual Standby**.

2 On the Destination page of the Export Wizard, in the Recover to a Virtual Machine drop-down menu, select **ESXi**.

3 Enter the information for accessing the virtual machine as described in the following table, and then click **Next**.

Table 77. ESXi credentials

Option	Description
Host name	Enter a name for the host machine.
Port	Enter the port for the host machine. The default is 443.
User name	Enter the logon credentials for the host machine.
Password	Enter the logon credentials for the host machine.

4 On the Virtual Machine Options page, enter the information described in the following table.

Table 78. Virtual machine options

Option	Description
Resource Pool	Select a resource pool from the drop-down list.
Data Store	Select a data store from the drop-down list.
Virtual Machine Name	Enter a name for the Virtual Machine.
Memory	Specify the memory usage for the virtual machine by clicking one of the following: • **Use the same amount of RAM as source machine** • **Use a specific amount of RAM, and then specify the amount in MB** The minimum amount is 1024 MB and the maximum allowed by the application is 65536 MB. The maximum amount of memory usage is limited by the amount of RAM available to the host machine.
Disk Provisioning	Select the type of disk provisioning as either Thin or Thick.
Disk Mapping	Specify the type of disk mapping as appropriate (Automatic, Manual, or with VM).
Version	Select the version of the virtual machine.
Perform initial one-time export	Select to perform the virtual export immediately instead of after the next scheduled snapshot (optional)

5 Click **Next**.

6 On the Volumes page, select the volumes you want to export, and then click **Next**.

7 On the Summary page, click **Finish** to complete the wizard and start the export.

NOTE: You can monitor the status and progress of the export by viewing the Virtual Standby or Events tab.

Exporting data to a VMware Workstation virtual machine

In AppAssure, you can choose to export data to VMware Workstation by performing a one-time export or by establishing a continual export (for virtual standby). Complete the steps in the following procedures for the appropriate type of export.

Performing a one-time VMware Workstation export

Complete the steps in this procedure to perform a one-time export to VMware Workstation.

To perform a one-time VMware Workstation export

1 In the AppAssure Core Console, do one of the following:

- From the button bar, click **Export** to launch the Export Wizard, and do the following:
  a On the Select Export Type page, select **One-time export**, and then click **Next**.
  b On the Protected Machines page, select the protected machine you want to export to a virtual machine, and then click **Next**.

- Navigate to the machine you want to export, and then in the Summary tab, from the Actions drop-down menu for that machine, select **Export** > **One-time**.

  The Export Wizard appears on the Recovery Points page.

2 On the Recovery Points page, select the recovery point from the AppAssure Core that you want to export, and then click **Next**.

Defining one-time settings for performing a VMware Workstation export

Complete the steps in this procedure to define the settings for performing a one-time VMware Workstation export.

To define one-time settings for performing a VMware Workstation export

1 On the Destination page in the Export Wizard, in the Recover to Virtual machine drop-down menu, select **VMware Workstation**, and then click **Next**.

2 On the Virtual Machine Options page, enter the parameters for accessing the virtual machine as described in the following table.

Table 79. Virtual machine parameters

Option	Description
Location	Specify the path of the local folder or network share on which to create the virtual machine. **NOTE:** If you specified a network share path, you will need to enter a valid logon credentials for an account that is registered on the target machine. The account must have read and write permissions to the network share.
User name	Enter the logon credentials for the network location for the export. • If you specified a network share path, you need to enter a valid user name for an account that is registered on the target machine. • If you entered a local path, a user name is not required.
Password	Enter the logon credentials for the network location for the export. • If you specified a network share path, you need to enter a valid password for an account that is registered on the target machine. • If you entered a local path, a password is not required.
Virtual Machine Name	Enter a name for the virtual machine being created; for example, VM-0A1B2C3D4. **NOTE:** The default name is the name of the source machine.
Version	Specify the version of VMware Workstation for the virtual machine. You can choose from: • VMware Workstation 7.0 • VMware Workstation 8.0 • VMware Workstation 9.0 • VMware Workstation 10.0
Memory	Specify the memory usage for the virtual machine by clicking one of the following: • **Use the same amount of RAM as source machine** • **Use a specific amount of RAM, and then specify the amount in MB** The minimum amount is 1024 MB and the maximum allowed by the application is 65536 MB. The maximum amount of memory usage is limited by the amount of RAM available to the host machine.

3 Click **Next**.

4 On the Volumes page, select the volumes to export, for example, C:\ and D:\, and then click **Next**.

5 On the Summary page, click **Finish** to complete the wizard and start the export.

NOTE: You can monitor the status and progress of the export by viewing the Virtual Standby or Events tab.

Performing a continual (Virtual Standby) VMware Workstation export

Complete the steps in this procedure to perform a continual export to VMware Workstation.

To perform a continual (virtual standby) VMware Workstation export

1 In the AppAssure Core Console, do one of the following:

- On the Virtual Standby tab, click **Add** to launch the Export Wizard. On the Protected Machines page of the Export Wizard, select the protected machine you want to export, and then click **Next**.
- Navigate to the machine you want to export, and, on the Summary tab in the Actions drop-down menu for that machine, click **Export**> **Virtual Standby**.

2 On the Destination page of the Export Wizard, in the Recover to a Virtual Machine drop-down menu, select **VMware Workstation**, and then click **Next**.

3 On the Virtual Machine Options page, enter the parameters for accessing the virtual machine as described in the following table.

Table 80. Virtual machine parameters

Option	Description
Target Path	Specify the path of the local folder or network share on which to create the virtual machine. NOTE: If you specified a network share path, you will need to enter a valid logon credentials for an account that is registered on the target machine. The account must have read and write permissions to the network share.
User name	Enter the logon credentials for the network location for the export. • If you specified a network share path, you need to enter a valid user name for an account that is registered on the target machine. • If you entered a local path, a user name is not required.
Password	Enter the logon credentials for the network location for the export. • If you specified a network share path, you need to enter a valid password for an account that is registered on the target machine. • If you entered a local path, a password is not required.
Virtual Machine	Enter a name for the virtual machine being created; for example, VM-0A1B2C3D4. NOTE: The default name is the name of the source machine.

Table 80. Virtual machine parameters

Option	Description
Version	Specify the version of VMware Workstation for the virtual machine. You can choose from: • VMware Workstation 7.0 • VMware Workstation 8.0 • VMware Workstation 9.0 • VMware Workstation 10.0
Memory	Specify the memory usage for the virtual machine by clicking one of the following: • **Use the same amount of RAM as source machine** • **Use a specific amount of RAM, and then specify the amount in MB** The minimum amount is 1024 MB and the maximum allowed by the application is 65536 MB. The maximum amount of memory usage is limited by the amount of RAM available to the host machine.

4 Select **Perform initial one-time export** to perform the virtual export immediately instead of after the next scheduled snapshot.

5 Click **Next**.

6 On the Volumes page, select the volumes to export, for example, C:\ and D:\, and then click **Next**.

7 On the Summary page, click **Finish** to complete the wizard and to start the export.

NOTE: You can monitor the status and progress of the export by viewing the Virtual Standby or Events tab.

Exporting data to a Hyper-V virtual machine

In AppAssure, you can choose to export data using Hyper-V Export by performing a one-time export, or by establishing a continual export (for Virtual Standby).

AppAssure supports first-generation Hyper-V export to the following hosts:

- Windows 8
- Windows 8.1
- Windows Server 2008
- Windows Server 2008 R2
- Windows Server 2012
- Windows Server 2012 R2

AppAssure supports second-generation Hyper-V export to the following hosts:

- Windows 8.1
- Windows Server 2012 R2

NOTE: Not all protected machines can be exported to Hyper-V second generation hosts.

Only protected machines with the following Unified Extensible Firmware Interface (UEFI) operating systems support virtual export to Hyper-V second generation hosts:

- Windows 8 (UEFI)
- Windows 8.1 (UEFI)
- Windows Server 2012 (UEFI)
- Windows Server 2012R2 (UEFI)

NOTE: Hyper-V export to second-generation VM can fail if the Hyper-V host does not have enough RAM allocated to perform the export.

Complete the steps in the following procedures for the appropriate type of export.

Performing a one-time Hyper-V export

Complete the steps in this procedure to perform a one-time export to Hyper-V.

To perform a one-time Hyper-V export

1 In the AppAssure Core Console, do one of the following:

- From the button bar, click **Export** to launch the Export Wizard, and do the following:
  a On the Select Export Type page, select **One-time export**, and then click **Next**.
  b On the Protected Machines page, select the protected machine you want to export to a virtual machine, and then click **Next**.
- Navigate to the machine you want to export, and then in the Summary tab, from the Actions drop-down menu for that machine, select **Export** > **One-time**.

  The Export Wizard appears on the Recovery Points page.

2 On the Recovery Points page, select the recovery point from the AppAssure Core that you want to export, and then click **Next**.

Defining one-time settings for performing a Hyper-V export

Complete the steps in this procedure to define the settings for performing a one-time Hyper-V export.

To define one-time settings for performing a Hyper-V export

1 On the Destination page in the Export Wizard, in the Recover to Virtual machine drop-down menu, select **Hyper-V**.

2 To export to a local machine with the Hyper-V role assigned, click **Use local machine**.

3 To indicate that the Hyper-V server is located on a remote machine, click the **Remote host** option, and then enter the information for the remote host as described in the following table.

Table 81. Remote host information

Text Box	Description
Host Name	Enter an IP address or host name for the Hyper-V server. It represents the IP address or host name of the remote Hyper-V server.
Port	Enter a port number for the machine. It represents the port through which the Core communicates with this machine.

Table 81. Remote host information

Text Box	Description
User name	Enter the user name for the user with administrative privileges for the workstation with the Hyper-V server. It is used to specify the logon credentials for the virtual machine.
Password	Enter the password for the user account with administrative privileges on the workstation with Hyper-V server. It is used to specify the logon credentials for the virtual machine.

4 Click **Next**.

5 On the Virtual Machines Options page in the VM Machine Location text box, enter the path for the virtual machine; for example, D:\export. This is used to identify the location of the virtual machine.

> **NOTE:** You need to specify the virtual machine location for both local and remote Hyper-V servers. The path should be a valid local path for the Hyper-V server. Non-existent directories are automatically created. You should not attempt to create them manually. Export to shared folders, for example, \\data\share is not permitted.

6 Enter the name for the virtual machine in the **Virtual Machine Name** text box.

The name you enter appears in the list of virtual machines in the Hyper-V Manager console.

7 To specify memory usage, click one of the following:

- **Use the same amount of RAM as the source machine** - To identify that the RAM use is identical between the virtual and source machines.
- **Use a specific amount of RAM, and then specify the amount in MB**

  The minimum amount is 1024 MB and the maximum allowed by the application is 65536 MB. The maximum amount of memory usage is limited by the amount of RAM available to the host machine.

8 To specify the disk format, next to Disk Format, click one of the following:

- **VHDX**
- **VHD**

> **NOTE:** Hyper-V Export supports VHDX disk formats if the target machine is running Windows 8 (Windows Server 2012) or higher. If the VHDX is not supported for your environment, the option is disabled.
>
> If exporting to Hyper-V generation 2, only VHDX disk format is supported.

9 To specify generation of Hyper-V to use for export, click one of the following:

- Generation 1
- Generation 2

> **NOTE:** Only generation 2 supports the secure boot option.

10 Specify the appropriate network adapter for the exported VM.

11 On the Volumes page, select the volume(s) to export; for example, C:\.

> **NOTE:** If the selected volumes are larger than the appropriate maximum allocations supported by the application as indicated below, or exceed the amount of space available, you will receive an error.

- For VHDX disk format, your selected volumes should be no larger than 64 TB.
- For VHD disk format, your selected volumes should be no larger than 2040 GB.

12 On the Summary page, click **Finish** to complete the wizard and to start the export.

> **NOTE:** You can monitor the status and progress of the export by viewing the Virtual Standby or Events tab.

Performing a continual (Virtual Standby) Hyper-V export

Complete the steps in this procedure to perform a continual export to Hyper-V

To perform a continual (Virtual Standby) Hyper-V export

1 In the AppAssure Core Console, do one of the following:

- On the Virtual Standby tab, click **Add** to launch the Export Wizard. On the Protected Machines page of the Export Wizard, select the protected machine you want to export, and then click **Next**.
- Navigate to the machine you want to export, and, on the Summary tab in the Actions drop-down menu for that machine, click **Export** > **Virtual Standby**.

2 To export to a local machine with the Hyper-V role assigned, click **Use local machine**.

3 To indicate that the Hyper-V server is located on a remote machine, click the **Remote host** option, and then enter the parameters for the remote host as described in the following table.

Table 82. Remote host information

Text Box	Description
Host Name	Enter an IP address or host name for the Hyper-V server. It represents the IP address or host name of the remote Hyper-V server.
Port	Enter a port number for the machine. It represents the port through which the Core communicates with this machine.
User name	Enter the user name for the user with administrative privileges for the workstation with the Hyper-V server. It is used to specify the logon credentials for the virtual machine.
Password	Enter the password for the user account with administrative privileges on the workstation with Hyper-V server. It is used to specify the logon credentials for the virtual machine.

4 Click **Next**.

5 On the Virtual Machines Options page in the VM Machine Location text box, enter the path for the virtual machine; for example, D:\export. This is used to identify the location of the virtual machine.

> **NOTE:** You need to specify the virtual machine location for both local and remote Hyper-V servers. The path should be a valid local path for the Hyper-V server. Non-existent directories are automatically created. You should not attempt to create them manually. Export to shared folders, for example, \\data\share is not permitted.

6 Enter the name for the virtual machine in the **Virtual Machine Name** text box.

The name you enter appears in the list of virtual machines in the Hyper-V Manager console.

7 To specify memory usage, click one of the following:

- **Use the same amount of RAM as the source machine** - To identify that the RAM use is identical between the virtual and source machines.
- **Use a specific amount of RAM, and then specify the amount in MB**

  The minimum amount is 1024 MB and the maximum allowed by the application is 65536 MB. The maximum amount of memory usage is limited by the amount of RAM available to the host machine.

8 To specify the disk format, next to Disk Format, click one of the following:

- **VHDX**
- **VHD**

> **NOTE:** Hyper-V Export supports VHDX disk formats if the target machine is running Windows 8 (Windows Server 2012) or higher. If the VHDX is not supported for your environment, the option is disabled.
>
> If exporting to Hyper-V generation 2, only VHDX disk format is supported.

9 To specify generation of Hyper-V to use for export, click one of the following:

- Generation 1
- Generation 2

> **NOTE:** Only generation 2 supports the secure boot option.

10 Specify the appropriate network adapter for the exported VM.

11 On the Volumes page, select the volume(s) to export; for example, C:\.

> **NOTE:** If the selected volumes are larger than the appropriate maximum allocations supported by the application as indicated below, or exceed the amount of space available, you will receive an error.

- For VHDX disk format, your selected volumes should be no larger than 64 TB.
- For VHD disk format, your selected volumes should be no larger than 2040 GB.

12 Select **Perform initial one-time export** to perform the virtual export immediately instead of after the next scheduled snapshot.

13 On the Summary page, click **Finish** to complete the wizard and to start the export.

> **NOTE:** You can monitor the status and progress of the export by viewing the Virtual Standby or Events tab.

Exporting data to a VirtualBox virtual machine

In AppAssure, you can choose to export data using a VirtualBox Export by performing a one-time export, or by establishing a continual export (for virtual standby).

You can also export to a VirtualBox VM installed on a Linux machine. For more information, see Exporting data to a Linux-based VirtualBox virtual machine.

Complete the steps in the following procedures for the appropriate type of export.

> **NOTE:** To perform this type of export, you should have VirtualBox installed on the Core machine. Virtual Box Version 4.2.18 or higher is supported for Windows hosts.

Performing a one-time VirtualBox export

Complete the steps in this procedure to perform a one-time export to VirtualBox.

To perform a one-time VirtualBox export

1 In the AppAssure Core Console, do one of the following:

- From the button bar, click **Export** to launch the Export Wizard, and do the following:
  - a On the Select Export Type page, select **One-time export**, and then click **Next**.
  - b On the Protected Machines page, select the protected machine you want to export to a virtual machine, and then click **Next**.
- Navigate to the machine you want to export, and then in the Summary tab, from the Actions drop-down menu for that machine, select **Export** > **One-time**.

  The Export Wizard appears on the Recovery Points page.

2 On the Recovery Points page, select the recovery point from the AppAssure Core that you want to export, and then click **Next**.

3 On the Destination page in the Export Wizard, in the Recover to Virtual machine drop-down menu, select **VirtualBox**, and then click **Next**.

4 On the Virtual Machine Options page, select **Use Windows machine**.

5 Enter the parameters for accessing the virtual machine as described in the following table.

Table 83. Virtual machine parameters

Option	Description
Virtual Machine Name	Enter a name for the virtual machine being created. **NOTE:** The default name is the name of the source machine.
Target Path	Specify a local or remote target path to create the virtual machine. **NOTE:** The target path should not be a root directory. If you specify a network share path, you will need to enter valid logon credentials (user name and password) for an account that is registered on the target machine. The account must have read and write permissions to the network share.
Memory	Specify the memory usage for the virtual machine by clicking one of the following: • **Use the same amount of RAM as source machine** • **Use a specific amount of RAM, and then specify the amount in MB** The minimum amount is 1024 MB and the maximum allowed by the application is 65536 MB. The maximum amount of memory usage is limited by the amount of RAM available to the host machine.

6 To specify a user account for the virtual machine, select **Specify the user account for the exported virtual machine**, and then enter the following information. This refers to a specific user account for which the virtual machine will be registered in the event there are multiple user accounts on the virtual machine. When this user account is logged on, only this user will see this Virtual Machine in VirtualBox manager. If an account is not specified, then the Virtual Machine will be registered for all existing users on the Windows machine with VirtualBox.

- **User name** - Enter the user name for which the virtual machine is registered.
- **Password** - Enter the password for this user account.

7 Click **Next**.

8 On the Volumes page, select the volumes to export, for example, C:\ and D:\, and then click **Next**.

9 On the Summary page, click **Finish** to complete the wizard and to start the export.

NOTE: You can monitor the status and progress of the export by viewing the Virtual Standby or Events tab.

Performing a continual (Virtual Standby) VirtualBox export

Complete the steps in this procedure to create a Virtual Standby and perform a continual export to VirtualBox.

To perform a continual (virtual standby) VirtualBox export

1 In the AppAssure Core Console, do one of the following:

- On the Virtual Standby tab, click **Add** to launch the Export Wizard. On the Protected Machines page of the Export Wizard, select the protected machine you want to export, and then click **Next**.
- Navigate to the machine you want to export, and, on the Summary tab in the Actions drop-down menu for that machine, click **Export** > **Virtual Standby**.

2 On the Destination page in the Export Wizard, in the Recover to Virtual machine drop-down menu, select **VirtualBox**, and then click **Next**.

3 On the Virtual Machine Options page, select **Use Windows machine**.

4 Enter the parameters for accessing the virtual machine as described in the following table.

Table 84. Virtual machine parameters

Option	Description
Virtual Machine Name	Enter a name for the virtual machine being created. **NOTE:** The default name is the name of the source machine.
Target Path	Specify a local or remote target path to create the virtual machine. **NOTE:** The target path should not be a root directory. If you specify a network share path, you will need to enter valid logon credentials (user name and password) for an account that is registered on the target machine. The account must have read and write permissions to the network share.
Memory	Specify the memory usage for the virtual machine by clicking one of the following: • **Use the same amount of RAM as source machine** • **Use a specific amount of RAM, and then specify the amount in MB** The minimum amount is 1024 MB and the maximum allowed by the application is 65536 MB. The maximum amount of memory usage is limited by the amount of RAM available to the host machine.

5 To specify a user account for the virtual machine, select **Specify the user account for the exported virtual machine**, and then enter the following information. This refers to a specific user account for which the virtual machine will be registered in the event there are multiple user accounts on the virtual machine. When this user account is logged on, only this user will see this Virtual Machine in VirtualBox manager. If an account is not specified, then the Virtual Machine will be registered for all existing users on the Windows machine with VirtualBox.

- **User name** - Enter the user name for which the virtual machine is registered.
- **Password** - Enter the password for this user account.

6 Select **Perform initial one-time export** to perform the virtual export immediately instead of after the next scheduled snapshot.

7 Click **Next**.

8 On the Volumes page, select the volumes to export, for example, C:\ and D:\, and then click **Next**.

9 On the Summary page, click **Finish** to complete the wizard and to start the export.

> **NOTE:** You can monitor the status and progress of the export by viewing the Virtual Standby or Events tab.

10

Managing protected machines

This chapter describes how to view, configure and manage the protected machines in your AppAssure environment. It includes the following sections:

- Configuring machine settings
- Accessing protected machine diagnostics
- Managing machines
- Managing snapshots and recovery points

From the Home tab on the AppAssure Core Console, you can view summary information for any machines protected by the Core.

NOTE: A software agent acts on behalf of the user to take specific actions. Protected machines are sometimes referred to as *agents*, since they run the AppAssure Agent software to facilitate data backup and replication on the AppAssure Core.

You can view the status, the display name for each machine, which repository it uses, the date and time of the last snapshot, how many recovery points exist in the repository for the machine, and the total amount of storage space the snapshots use in the repository.

To manage aspects of any protected machine, start by navigating to the machine you want to view, configure, or manage. From the Home tab, there are three ways to navigate to a protected machine:

- You can click on the display name of any protected machine. This takes you to the Summary tab for any protected machine. For a detailed description of this page, see Viewing the Machines tab for a protected machine.
- In the left navigation area, you can click **Protected Machines**. The Machines tab appears. On the Protected Machines pane, you can see summary information about each machine. For a detailed description of this page, see Viewing the Summary tab.
- In the left navigation area, under Protected Machines, you can click any protected machine display name. This takes you to the Summary tab for any protected machine. For a detailed description of this page, see Viewing the Summary tab

The tasks you can accomplish to manage protected machines are broken down into a few categories.

- You can configure machine settings, access system information, or configure notifications for events regarding a particular machine. For more information, see Configuring machine settings.
- You can for access diagnostics for a protected machine. For more information, see Accessing protected machine diagnostics.
- You can remove a machine from protection, cancel current operations, or view license information for a protected machine. For more information, see Managing machines.
- You can view and manage data saved in the Core. For more information, see Managing snapshots and recovery points.

Configuring machine settings

Once you have added protection for machines in AppAssure, you can easily modify basic machine configuration settings (display name, host name, port, encryption key, and repository), protection settings (changing the protection schedule for volumes on the machine, adding or removing volumes, or pausing protection), and more. This section describes the various ways you can view and modify machine settings in AppAssure.

Viewing and modifying configuration settings

Complete the steps in this procedure to view and modify configuration settings.

This task is also a step in the Modifying cluster node settings.

To view and modify configuration settings

1. In the AppAssure Core Console, navigate to the machine for which you want to view and modify configuration settings.
2. Click the Configuration tab.

   The Settings page displays.
3. Click **Change** to modify the machine settings as described in the following table.

Table 85. Machine settings

Text Box	Description
Display Name	Enter a display name for the machine. This is the name that displays for a protected machine in the AppAssure Core Console. You can enter up to 64 characters. By default, this is the host name of the machine. You can change this to something more user-friendly if needed. Do not use prohibited characters or prohibited phrases.
Host Name	Enter a host name for the machine.
Port	Enter a port number for the machine. The port is used by the AppAssure Core service to communicate with this machine. The default port is *8006*.
Encryption Key	If you want encryption defined for this AppAssure Core to be applied to the data for every volume on this protected machine, you can specify the encryption key here. If no encryption keys exist, you can add an encryption key. For more information on managing encryption keys, see If the volumes on this protected machine are encrypted, you can change to a different encryption key. Alternatively, you can disassociate an encryption key by selecting (none) from the Encryption key drop-down menu. **NOTE:** After you apply an encryption key, change an encryption key, or you disassociate an encryption key for a protected machine, AppAssure takes a new base image upon the next scheduled or forced snapshot.
Repository	Select a repository for the recovery points. Displays the repository on the AppAssure Core in which to store the data from this machine. **NOTE:** This setting can only be changed if there are no recovery points or if the previous repository is missing.

Viewing system information for a machine

The AppAssure Core Console provides you with easy access to all of the machines that are being protected.

Complete the steps in this procedure to view detailed system information for a protected machine.

To view system information for a machine

1. In the left navigation area of the Core Console, under Protected Machines, select the machine for which you want to view detailed system information.
2. Click the Tools tab for that machine, which opens with the System Info page displayed.

   The System Information page displays detailed information about the machine, including the following:

   - Host Name
   - OS Version
   - OS Architecture
   - Memory (Physical)
   - Display Name
   - Fully Qualified Domain Name
   - Virtual Machine Type (if applicable)

   Detailed information about the volumes contained on this machine also displays and includes:

   - Name
   - Device ID
   - File System
   - Capacity (including Raw, Formatted, and Used)

   Other machine information displayed, includes:

   - Processors
   - Network Adapters
   - IP Addresses associated with this machine

Configuring notification groups for system events

In AppAssure, you can configure how system events are reported for an individual machine by creating notification groups. These events could be system alerts, errors, and so on.

To configure notification groups for system events

1. In the AppAssure Core Console, navigate to the machine you want to modify.

   The Summary Tab appears.
2. Click the Configuration tab, and then click **Events**.

   The Notification Groups page displays.
3. Click **Use custom alert settings**.

   The Custom Notification Groups screen appears.
4. Click **Add Group** to add new notification groups for sending a list of system events.

   The Add Notification Group dialog box displays.

   NOTE: To use the default alert settings, select the **Use Core alert settings** option.

5 Add the notification options as described in the following table.

Table 86. Notification options

Text Box	Description
Name	Enter a name for the notification group.
Description	Enter a description for the notification group.
Enable Alerts	Select which events to share with this notification group. You can select **All** or select a subset of events to include: • Exchange • Auto Update • Dedupe Cache • Recovery Point Check • Remote Mount • Boot CD • Security • Database Retention • Local Mount • Metadata • Clusters • Notification • Power Shell Scripting • Push Install • Attachability • Jobs • Licensing • Log Truncation • Archive • Core Service • Export • Protection • Replication • Repository • Rollback (Restore) • Rollup **NOTE:** When you choose to select by type, by default, the appropriate events are automatically enabled. For example, if you choose *Warning*, the Attachability, Jobs, Licensing, Archive, CoreService, Export, Protection, Replication, and Rollback (Restore) events are enabled.

Table 86. Notification options

Text Box	Description

Notification Options	Select the method to specify how to handle notifications.
	You can choose from the following options:
	• **Notify by Email**. You would need to specify to which email addresses to send the events in the To, CC and, optionally, BCC text boxes.
	NOTE: To receive mail, SMTP must be previously configured.
	• **Notify by Windows Event log**. The Windows Event log controls the notification.
	• **Notify by syslogd**. You would need to specify to which host name and port to send the events.
	  • **Host.** Enter the host name for the server.
	  • **Port.** Enter a port number for communicating with the server.
	• **Notify by Toast alerts.** Select this option if you want the alert to appear as a pop-up in the lower-right corner of your screen.

6 Click **OK** to save your changes.

7 To edit an existing notification group, click **Edit** next to the notification group that **you want to edit**.

The Edit Notification Group dialog box displays for you to edit the settings.

Editing notification groups for system events

Complete the steps in the following procedure to edit notification groups for system events.

To edit notification groups for system events

1 In the AppAssure Core Console, navigate to the machine you want to modify.

The Summary Tab appears.

2 Click the Configuration tab, and then click **Events**.

3 Click **Use custom alert settings**.

The Custom Notification Groups screen appears.

4 Click the Edit icon under the Action column.

The Edit Notification Group dialog box displays.

5 Edit the notification options as described in the following table.

Table 87. Notification options

Text Box	Description
Name	Represents the name of the notification group. NOTE: You cannot edit the name of the notification group.
Description	Enter a description for the notification group.

Table 87. Notification options

Text Box	Description
Enable Alerts	Select which events to share with the notification group. You can select **All** or select a subset of events to include: • Exchange • Auto Update • Dedupe Cache • Recovery Point Check • Remote Mount • Boot CD • Security • Database Retention • Local Mount • Metadata • Clusters • Notification • Power Shell Scripting • Push Install • Attachability • Jobs • Licensing • Log Truncation • Archive • Core Service • Export • Protection • Replication • Repository • Rollback (Restore) • Rollup You can also choose to select by type, which are: Info, Warning, and Error. **NOTE:** When you choose to select by type, by default, the appropriate events are automatically enabled. For example, if you choose *Warning*, the Attachability, Jobs, Licensing, Archive, CoreService, Export, Protection, Replication, and Rollback (Restore) events are enabled.

Notification Options	Select the method to specify how to handle notifications. The options are:
	• **Notify by Email**. You would need to specify the email addresses to which to send the events in the To, CC and optionally, BCC text boxes.
	NOTE: To receive email, SMTP must be previously configured.
	• **Notify by Windows Event log**. The Windows Event log controls the notification.
	• **Notify by syslogd**. You would need to specify the host name and port to which to send the events.
	  • **Host.** Enter the host name for the server.
	  • **Port.** Enter a port number for communicating with the server.
	• **Notify by Toast alerts.** Select this option if you want the alert to appear as a pop-up in the lower-right corner of your screen.

6 Click **OK**.

Modifying transfer settings

In AppAssure, you can modify the settings to manage the data transfer processes for a protected machine. The transfer settings described in this section are set at the protected machine level. To affect transfer at the core level, see Modifying transfer queue settings.

AppAssure supports Windows 8 and Windows Server 2012 for normal transfers, both base and incremental, as well as with restore, bare metal restore, and virtual machine export.

There are three types of transfers in AppAssure:

- **Snapshot.** Backs up the data on your protected machine. Two types of snapshots are possible: a base image of all protected data, and an incremental snapshot for data updated since the last snapshot. This type of transfer creates recovery points, which are stored on the repository associated with the Core. For more information, see Managing snapshots and recovery points.
- **Virtual Machine Export.** Creates a virtual machine (VM) from a recovery point, containing all of the data from the backup of the protected machine, as well the operating system and drivers and associated data to ensure the VM is bootable. For more information, see Exporting protected data from Windows machines to virtual machines.
- **Restore**. Restores backup information to a protected machine. For more information, see Restoring volumes from a recovery point.

  NOTE: The entire volume is always rewritten during restore of Windows systems using EFI system partitions.

Data transfer in AppAssure involves the transmission of a volume of data along a network from AppAssure protected machines to the Core. In the case of replication, transfer also occurs from the originating or source Core to the target Core.

Data transfer can be optimized for your system through certain performance option settings. These settings control data bandwidth usage during the process of backing up protected machines, performing VM export, or performing a restore. These are some factors that affect data transfer performance:

- Number of concurrent agent data transfers
- Number of concurrent data streams
- Amount of data change on disk
- Available network bandwidth
- Repository disk subsystem performance
- Amount of memory available for data buffering

You can adjust the performance options to best support your business needs and fine-tune the performance based on your environment.

To modify transfer settings

1. In the AppAssure Core Console, navigate to the machine you want to modify.
2. Click the Configuration tab, and then click **Transfer Settings**.

   The current transfer settings are displayed.
3. On the Transfer Settings page, click **Change**.

   The Transfer Settings dialog box displays.
4. Enter the Transfer Settings options for the machine as described in the following table.

Table 88. Transfer Settings options

Text Box	Description
Priority	Sets the transfer priority between protected machines. Enables you to assign priority by comparison with other protected machines. Select a number from 1 to 10, with 1 being the highest priority. The default setting establishes a priority of 5. **NOTE:** Priority is applied to transfers that are in the queue.
Maximum Concurrent Streams	Sets the maximum number of TCP links that are sent to the Core to be processed in parallel per protected machine. **NOTE:** Dell recommends setting this value to 8. If you experience dropped packets, try increasing this setting.
Maximum Concurrent Writes	Sets the maximum number of simultaneous disk write actions per protected machine connection. **NOTE:** Dell recommends setting this to the same value you select for Maximum Concurrent Streams. If you experience packet loss, set slightly lower—for example, if Maximum Current Streams is 8, set this to 7.
Use Core Default Maximum Retries	Select this option to use default retries number for each protected machine, if some of the operations fail to complete.
Maximum Segment Size	Specifies the largest amount of data, in bytes, that a computer can receive in a single TCP segment. The default setting is 4194304. **CAUTION: Do not change this setting from the default.**
Maximum Transfer Queue Depth	Specifies the amount of commands that can be sent concurrently. You can adjust this to a higher number if your system has a high number of concurrent input/output operations.
Outstanding Reads per Stream	Specifies how many queued read operations will be stored on the back end. This setting helps to control the queuing of protected machines. **NOTE:** Dell recommends setting this value to 24.
Excluded Writers	Select a writer if you want to exclude it. Since the writers that appear in the list are specific to the machine you are configuring, you will not see all writers in your list. For example, some writers you may see include: • ASR Writer • COM+ REGDB Writer • Performance Counters Writer • Registry Writer • Shadow Copy Optimization Writer • SQLServerWriter • System Writer • Task Scheduler Writer • VSS Metadata Store Writer • WMI Writer
Transfer Data Server Port	Sets the port for transfers. The default setting is 8009.
Transfer Timeout	Specifies in minutes and seconds the amount of time to allow a packet to be static without transfer.
Snapshot Timeout	Specifies in minutes and seconds the maximum time to wait to take a snapshot.

Table 88. Transfer Settings options

Text Box	Description
Network Read Timeout	Specifies in minutes and seconds the maximum time to wait for a read connection. If the network read cannot be performed in that time, the operation is retried.
Network Write Timeout	Specifies the maximum time in seconds to wait for a write connection. If the network write cannot be performed in that time, the operation is retried.

5 Click **OK**.

Customizing nightly jobs for a protected machine

Nightly jobs can be configured at the Core level and the machine level on the appropriate Configuration tab. When nightly job settings are changed at the Core level, the changes are applied to all relevant machines protected by that Core. Changes made to the nightly jobs at the machine level supersede the changes made at the Core level.

For a list of all nightly jobs, including descriptions and the scope available for each, see the topic Understanding nightly jobs.

Complete the steps in the following procedure to make changes to the nightly jobs for a single machine.

To customize nightly jobs for a protected machine

1 In the left navigation area of the AppAssure Core, select the machine for which you want to customize nightly jobs.

The Summary tab for the selected machine displays.

2 Click the Configuration tab for the machine, and then click **Settings**.

3 Next to Nightly Jobs, click **change**.

The Nightly Jobs dialog box appears.

4 Select the jobs you want to include in the nightly jobs or clear the options you want to omit for this machine.

NOTE: Options may vary by machine. For example, a protected machine using Exchange Server may include Checksum Check Job and Truncate Exchange logs.

5 Click **OK**.

NOTE: The results of this procedure apply only to the select protected machine. To apply elsewhere, repeat the procedure for each machine you want to customize. To change the nightly job settings for all machines protected by a Core, see Configuring nightly jobs for the Core, Configuring nightly SQL attachability checks and log truncation for all protected machines, or Configuring nightly Exchange database checksum checks and log truncation.

Accessing protected machine diagnostics

In AppAssure, you can download and view diagnostic information for individual protected machines. Additionally, AppAssure lets you download, view, and upload log data for the Core. For more information about Core logs, see Accessing diagnostics for the Core. To access machine logs, see the following procedures:

- Downloading machine logs
- Viewing machine logs

Downloading machine logs

If you encounter any errors or issues with a protected machine, you can download the machine logs to view them or upload them to AppAssure Support.

To download machine logs

1. Navigate to the AppAssure Core Console, and then select the protected machine for which you want to download logs.
2. Click the Tools drop-down menu or tab, lick **Diagnostics**, and then click **View Log**.
3. On the Download Agent Log page of the Tools tab, click **Click here to begin the download**.
4. In the Opening AgentAppRecovery.log dialog box, select **Save File**.
5. Click **OK**.

   The file saves to your Downloads folder.

Viewing machine logs

If you encounter any errors or issues with the machine, it may be useful to view the logs.

To view machine logs

1. Navigate to the AppAssure Core Console, and then select the machine for which you want to view log data.
2. Click the Tools drop-down menu or tab, lick **Diagnostics**, and then click **View Log**.
3. On the Download Agent Log page of the Tools tab, click **Click here to begin the download**.
4. In the Opening AgentAppRecovery.log dialog box, select **Open with**, and then use the drop-down list to select a program with which you want to open the file; for example, Notepad.
5. Click **OK**.
6. The file opens in the program you selected.

Viewing machine status and other details

Complete the step in this procedure to view the status as well as other details for a machine.

To view machine status and other details

- In the AppAssure Core Console, navigate to the protected machine you want to view.

  The information about the machine displays on the Summary tab. The details that display include:

  - Host name
  - Last Snapshot taken
  - Next Snapshot scheduled
  - Encryption status
  - Version number

If Exchange Server is installed on the machine, detailed information about the server also displays and includes:

- Last successful Mountability check performed
- Last successful Checksum check performed
- Last Log Truncation performed

Details information about the volumes contained on this machine also displays and includes:

- Name
- File System type
- Space Usage
- Schedule
- Current Schedule
- Next Snapshot
- Total size

If SQL Server is installed on the machine, detailed information about the server also displays and includes:

- Online Status
- Name
- Install Path
- Version

If Exchange Server is installed on the machine, detailed information about the server and mail stores also displays and includes:

- Version
- Install Path
- Data Path
- Database Name
- Exchange Databases Path
- Log File Path
- Log Prefix
- System Path
- MailStore Type

Managing machines

This section describes a variety of tasks you can perform in managing your machines. Topics include:

- Removing a machine
- Canceling operations on a machine
- Viewing license information on a machine

Removing a machine

When you remove a machine from protection on the AppAssure Core, you are presented with two options: you can keep the recovery points saved thus far to the AppAssure Core, or you can remove the recovery points. If you keep the recovery points, you have what is known as a “recovery points only” machine. Using those recovery points for the machine that has been removed from current protection, you can continue to restore the machine in the future, but only up to the state captured in a saved recovery point.

If you remove the recovery points, this action deletes any snapshot data for that formerly protected machine from the AppAssure Core.

△ | **CAUTION: If you delete recovery points, you will no longer be able to restore data for that machine.**

Complete the steps in the following procedure to remove a machine from protection in your AppAssure environment.

To remove a machine

1. In the AppAssure Core Console, navigate to the machine you want to remove.
2. In the **Actions** drop-down menu, click **Remove Machine**, and then select one of the option described in the following table.

Table 89. Remove Machine options

Option	Description
Keep Recovery Points	Keeps all currently stored recovery points for this machine.
Remove Recovery Points	Removes all currently stored recovery points for this machine from the repository.

Canceling operations on a machine

You can cancel currently executing operations for a machine. You can specify to cancel just a current snapshot or to cancel all current operations, which would include exports, replications, and so on.

To cancel operations on a machine

1. In the AppAssure Core Console, navigate to the machine for which you want to cancel operations.
2. Click the Events tab.
3. Expand the event details for the event or operation you want to cancel.
4. Click **Cancel**.

Viewing license information on a machine

You can view current license status information for the AppAssure Agent software installed on a protected machine.

To view license information

1. In the AppAssure Core Console, navigate to the machine you want to view.
2. Click the Configuration tab, and then click **Licensing**.

   The Status screen appears and presents the following details about the product licensing:

   - Expiration Date
   - License Status
   - License Type
   - Agent Type

Managing snapshots and recovery points

A recovery point is a collection of snapshots taken of individual disk volumes and stored in the repository. Snapshots capture and store the state of a disk volume at a given point in time while the applications that generate the data are still in use. In AppAssure, you can force a snapshot, temporarily pause snapshots, and view lists of current recovery points in the repository as well as delete them if needed. Recovery points are used to restore protected machines or to mount to a local file system.

The snapshots that are captured by AppAssure are done so at the block level and are application aware. This means that all open transactions and rolling transaction logs are completed and caches are flushed to disk before creating the snapshot.

AppAssure uses a low-level volume filter driver, which attaches to the mounted volumes and then tracks all block-level changes for the next impending snapshot. Microsoft Volume Shadow Services (VSS) is used to facilitate application crash consistent snapshots.

Viewing recovery points

Complete the steps in the following procedure to view recovery points.

To view recovery points

1. In the AppAssure Core Console, navigate to the protected machine for which you want to view recovery points.
2. Click the Recovery Points tab.

   You can view information about the recovery points for the machine as described in the following table.

   Table 90. Recovery point information

Info	Description
Status	Indicates current status of the recovery point.
Encrypted	Indicates if the recovery point is encrypted.
Contents	Lists the volumes included in the recovery point.
Type	Defines a recovery point as either a base image or an incremental (differential) snapshot.

Table 90. Recovery point information

Info	Description
Creation Date	Displays the date when the recovery point was created.
Size	Displays the amount of space that the recovery point consumes in the repository.

Viewing a specific recovery point

Complete the steps in the following procedure to view details about a specific recovery point.

To view a specific recovery point

1. In the AppAssure Core Console, navigate to the protected machine for which you want to view recovery points.
2. Click the Recovery Points tab.
3. Click the right angle bracket > symbol next to a recovery point in the list to expand the view.

   You can view more detailed information about the contents of the recovery point for the selected machine, as well as access a variety of operations that can be performed on the recovery point, as described in the following table.

Table 91. Recovery point details

Info	Description
Actions	The Actions menu includes the following operations you can perform on the selected recovery point: **Mount**. Select this option to mount the selected recovery point. For more information, see Mounting a recovery point. **Export**. By using the Export option, you can export the selected recovery point to ESXi, VMware workstation, HyperV, or Virtual Box. For more information, see Exporting data to a Windows-based virtual machine. **Restore**. Select this option to perform a restore from the selected recovery point to a volume you specify. For more information, see Selecting a recovery point and initiating BMR. **Check.** If the protected machine has Exchange Server installed, select this option to force a checksum check or a mountability check. For more information, see Managing Exchange database mountability checks and log truncation. If the protected machine has SQL Server installed, select this option to force an attachability check. For more information, see Forcing a SQL Server attachability check.
Contents	The Contents area includes a row for each volume in the expanded recovery point, listing the following information for each volume: **Status**. Indicates current status of the recovery point. **Title**. Lists the specific volume in the recovery point. **Type**. Indicates if the specific recovery point is a base image or an incremental (differential) snapshot. **Size**. Displays the amount of space that the recovery point consumes in the repository.

4 Click the right angle bracket > symbol next to a volume in the selected recovery point to expand the view.

You can view information about the selected volume in the expanded recovery point as described in the following table.

Table 92. Volume information

Text Box	Description
Title	Indicates the specific volume in the recovery point.
File System	Indicates the file system type for the selected volume.
Raw Capacity	Indicates the amount of raw storage space on the entire volume.
Formatted Capacity	Indicates the amount of storage space on the volume that is available for data after the volume is formatted.
Used Capacity	Indicates the amount of storage space currently used on the volume.

Mounting a recovery point

In AppAssure, you can mount a recovery point for a Windows machine to access stored data through a local file system.

NOTE: To mount a Linux recovery point with the aamount utility, see Mounting a recovery point volume on a Linux machine.

NOTE: When mounting recovery points from data restored from a machine that has data deduplication enabled, you must also enable deduplication on the Core server.

To mount a recovery point

1 In the AppAssure Core Console, navigate to the machine that you want to mount to a local file system.

The Summary tab for the selected machine displays.

2 Click the Recovery Points tab.

3 In the list of recovery points, click the right angle bracket > symbol to expand the recovery point that you want to mount.

4 In the expanded details for that recovery point, click **Mount**.

The Mount Recovery Points dialog box appears.

5 In the Mount Recovery Point dialog box, edit the settings for mounting a recovery point as described in the following table.

Table 93. Mount Recovery Point settings

Option	Description
Mount Location: Local folder	Specify the path used to access the mounted recovery point.
Mount Options: Mount type	Specify the way to access data for the mounted recovery point: • Mount Read-only • Mount Read-only with previous writes • Mount Writable
Volume Images	Specify the volume images that you want to mount
Create a Windows share for this Mount	Optionally, select the check box to specify whether the mounted recovery point can be shared and then set access rights to it including the Share name and allowed groups.

6 Click **Mount** to mount the recovery point.

> **NOTE:** If you want to copy directories or files from a mounted recovery point to another Windows machine, you can use Windows Explorer to copy them with default permissions or original file access permissions. For details, see Restoring a directory or file using Windows Explorer to Restoring a directory or file and preserving permissions using Windows Explorer.

7 Optionally, while the task is in process, you can view its progress from the Running Tasks drop-down menu on the Core Console, or you can view detailed information in the Events tab. For more information about monitoring AppAssure events, see Viewing tasks, alerts, and events.

Dismounting recovery points

Complete the steps in this procedure to dismount recovery points that are mounted on the Core.

> **NOTE:** When dismounting a recovery point mounted remotely, this is referred to as *disconnecting*.

To dismount recovery points

1 In the AppAssure Core Console, click the Tools drop-down menu, and then click **Mounts**.

The Mounts page appears. There is a pane for Local Mounts (recovery points mounted from the Core) and another for Remote Mounts (recovery points mounted using the Local Mount Utility). In each pane, the respective mounted recovery points appears in the list.

2 For the recovery points you want to dismount, do the following:

- In the **Local Mounts** pane, for each locally mounted recovery point in the list that you want to dismount, click **dismount**. If you want to dismount all recovery points mounted locally, click **Dismount All**.
- In the **Remote Mounts** pane, for each remotely mounted recovery point in the list that you want to dismount, click **disconnect**. If you want to dismount all recovery points mounted remotely, click **Disconnect All**.

3 In the Dismounting the Recovery Point dialog box, click **Yes** to confirm.

4 Confirm that the previously mounted recovery points no longer appear in the Local Mounts or Remote Mounts list, as appropriate.

Forcing a snapshot

Forcing a snapshot lets you force a data transfer for the current protected machine. When you force a snapshot, the transfer starts immediately or is added to the queue. Only the data that has changed from a previous recovery point is transferred. If there is no previous recovery point, all data on the protected volumes is transferred, which is referred to as a base image.

> **NOTE:** AppAssure supports Window 8, Windows 8.1, Windows Server 2012, and Windows Server 2012 R2 for both base and incremental transfers.

To force a snapshot

1 In the AppAssure Core Console, navigate to the machine or cluster with the recovery point for which you want to force a snapshot.

2 On the Summary tab in the Volumes section specify the volumes for which the snapshot should be taken, and then click **Force Snapshot** or **Force Base Image** button.

Table 94. Force Snapshot options

Option	Description
Force Snapshot	Takes an incremental snapshot of data updated since the last snapshot was taken.
Force Base Image	Takes a complete snapshot of all data on the volumes of the machine.

Removing recovery points

You can easily remove recovery points for a particular machine from the repository. When you delete recovery points in AppAssure, you can specify one of the following options.

- **Delete All Recovery Points.** Removes all recovery points for the selected protected machine from the Repository.
- **Delete a Range of Recovery Points**. Removes all recovery points in a specified range before the current, up to and including the base image, which is all data on the machine as well as all recovery points after the current until the next base image.

NOTE: You cannot recover the recovery points you have deleted.

To remove recovery points

1 In the AppAssure Core Console, navigate to the machine for which you want to view recovery points.

2 Click the Recovery Points tab.

3 Click the **Actions** menu.

4 Select one of the following options:

- To delete all currently stored recovery points, click **Delete All**.
- To delete a set of recovery points in a specific data range, click **Delete Range**.

  The Delete dialog box displays.

  - In the Delete Range dialog box, specify the range of recovery points you want to delete using a start date and time and an end date and time, and then click **Delete**.

Deleting an orphaned recovery point chain

An orphaned recovery point is an incremental snapshot that is not associated with a base image. Subsequent snapshots continue to build onto this recovery point; however, without the base image, the resulting recovery points are incomplete and are unlikely to contain the data necessary to complete a recovery. These recovery points are considered to be part of the orphaned recovery point chain. If this situation occurs, the best solution is to delete the chain and create a new base image.

For more information about forcing a base image, see Forcing a snapshot.

To delete an orphaned recovery point chain

1 In the AppAssure Core Console, navigate to the protected machine for which you want to delete the orphaned recovery point chain.

2 Click the Recovery Points tab.

3 Under Recovery Points, expand the orphaned recovery point.

This recovery point is labeled in the Type column as “Incremental, Orphaned.”

4 Next to Actions, click **Delete**.

The Delete Recovery Points windows appears.

5 In the Delete Recovery Points window, click **Yes**.

CAUTION: Deleting this recovery point deletes the entire chain of recovery points, including any incremental recovery points that occur before or after it, until the next base image. This operation cannot be undone.

The orphaned recovery point chain is deleted.

Migrating recovery points to a different repository

If you want to remove the recovery points of a protected machine from a repository without deleting them, you can migrate them to a different repository using this procedure. For example, if you want to remove the recovery points for a protected machine because the repository is full or because you want to protect a machine using a different core and repository, you can use the following procedure to safely migrate your data.

To migrate recovery points to a different repository

1 In the AppAssure Core Console, pause protection for the protected machine or machines whose recovery points you want to migrate. For more information, see Pausing and resuming protection.

2 Cancel all current operations for the protected machine or machines whose recovery points you want to migrate. For more information, see Canceling operations on a machine.

3 Archive the recovery points for the machine or machines you paused. For more information, see Creating an archive.

4 Create a new repository for the migrated recovery points. For more information, see Creating a repository.

- If you want to use an existing repository, continue to Step 5.

5 After you finish creating the archive, delete all of the recovery points for the paused machines from the original repository using the **Delete All** option. For more information, see Removing recovery points.

6 Change the repository for each machine that you paused by completing the following steps:

a On the Core Console, click the protected machine in the navigation tree.

b Click the Configuration tab.

c Next to Settings, click **Change**.

d From the Repository drop-down list, select in the name of the repository you created in Step 4.

- If you want to use an existing repository, select the name of an existing repository.

NOTE: When migrating recovery points to an existing repository, ensure that the existing repository has enough free space to contain the migrated recovery points.

e Click **OK**.

7 Resume protection for the machine or machines that you paused. For more information, see Pausing and resuming protection.

8 Take a new base image for each of the protected machines you moved. For more information, see Forcing a snapshot and use the **Force Base Image** option.

9 Import the archived data for the machines you want to migrate. For more information, see Importing an archive.

11

Understanding replication

This chapter describes how to configure and manage the replication of protected data from an AppAssure source core to an AppAssure target core for disaster recovery. It includes the following sections:

- Understanding seed drives
- Understanding failover and failback in AppAssure
- Performance considerations for replicated data transfer
- Configuring replication
- Replicating to a self-managed target core
- Replicating to a third-party target core
- Adding a machine to existing replication
- Consuming the seed drive on a target core
- Managing replication settings
- Removing replication
- Recovering replicated data
- Understanding failover and failback

This section provides conceptual and procedural information to help you understand and configure replication in AppAssure.

Replication is the process of copying recovery points from an AppAssure core and transmitting them to another AppAssure core in a separate location for the purpose of disaster recovery. The process requires a paired source-target relationship between two or more cores.

The source core copies the recovery points of selected protected machines, and then asynchronously and continually transmits the incremental snapshot data to the target core at a remote disaster recovery site. You can configure outbound replication to a company-owned data center or remote disaster recovery site (that is, a "self-managed" target core). Or, you can configure outbound replication to a third-party managed service provider (MSP) or cloud provider that hosts off-site backup and disaster recovery services. When replicating to a third-party target core, you can use built-in work flows that let you request connections and receive automatic feedback notifications.

Replication is managed on a per-protected-machine basis. Any machine (or all machines) protected or replicated on a source core can be configured to replicate to a target core.

Possible scenarios for replication include:

- **Replication to a Local Location.** The target core is located in a local data center or on-site location, and replication is maintained at all times. In this configuration, the loss of the Core would not prevent a recovery.
- **Replication to an Off-site Location.** The target core is located at an off-site disaster recovery facility for recovery in the event of a loss.
- **Mutual Replication.** Two data centers in two different locations each contain a core and are protecting machines and serving as the off-site disaster recovery backup for each other. In this scenario, each core replicates the protected machines to the Core that is located in the other data center.

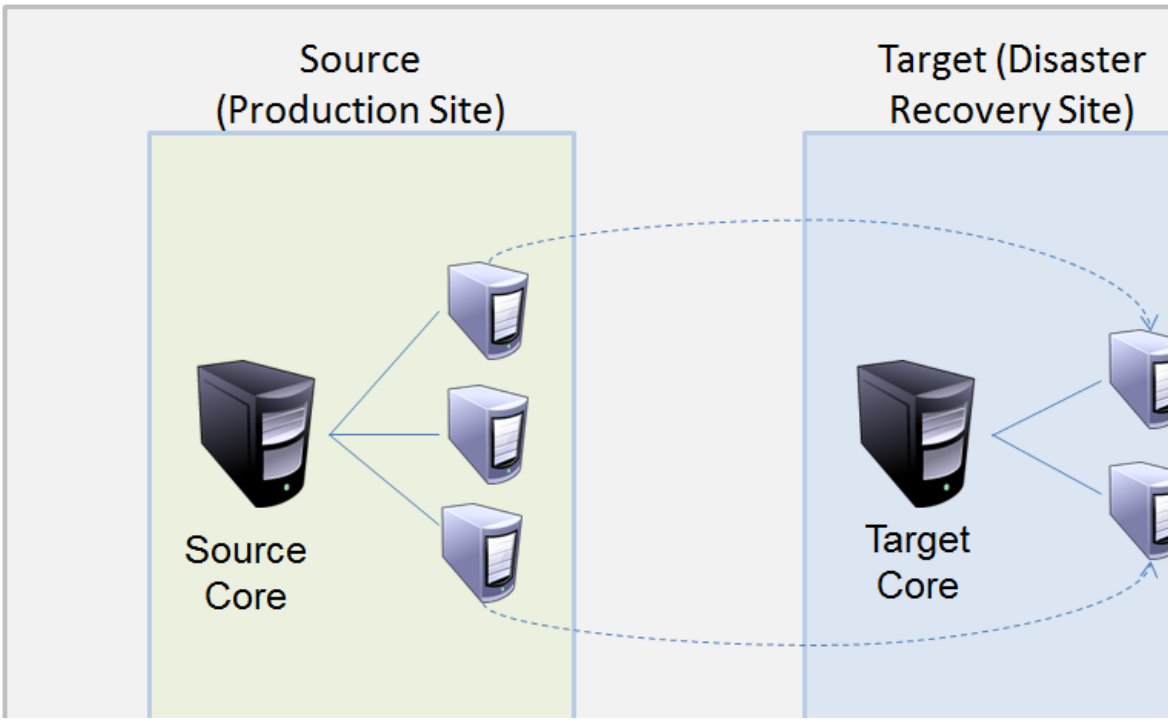
- **Hosted and Cloud Replication.** AppAssure MSP partners maintain multiple target cores in a data center or a public cloud. On each of these cores, the MSP partner lets one or more of their customers replicate recovery points from a source core on the customer's site to the MSP's target core for a fee.

  NOTE: In this scenario, customers would only have access to their own data.

Possible replication configurations include:

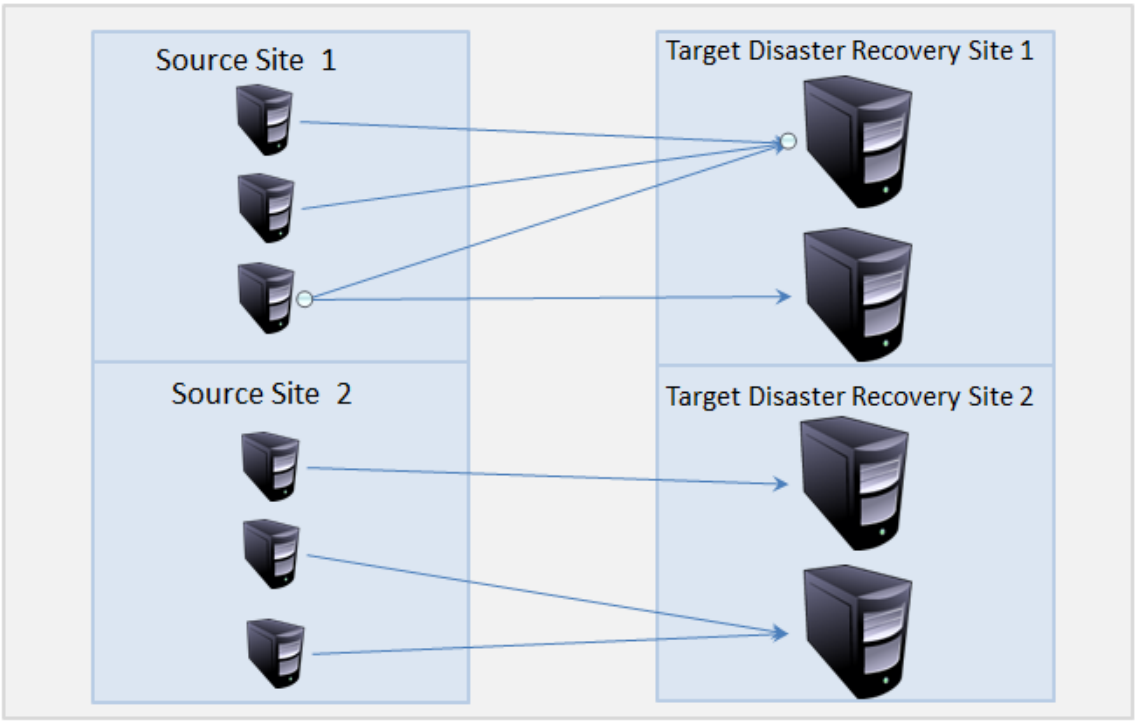
- **Point to Point.** Replicates a single protected machine from a single source core to a single target core.

  Figure 5. Point to point configuration



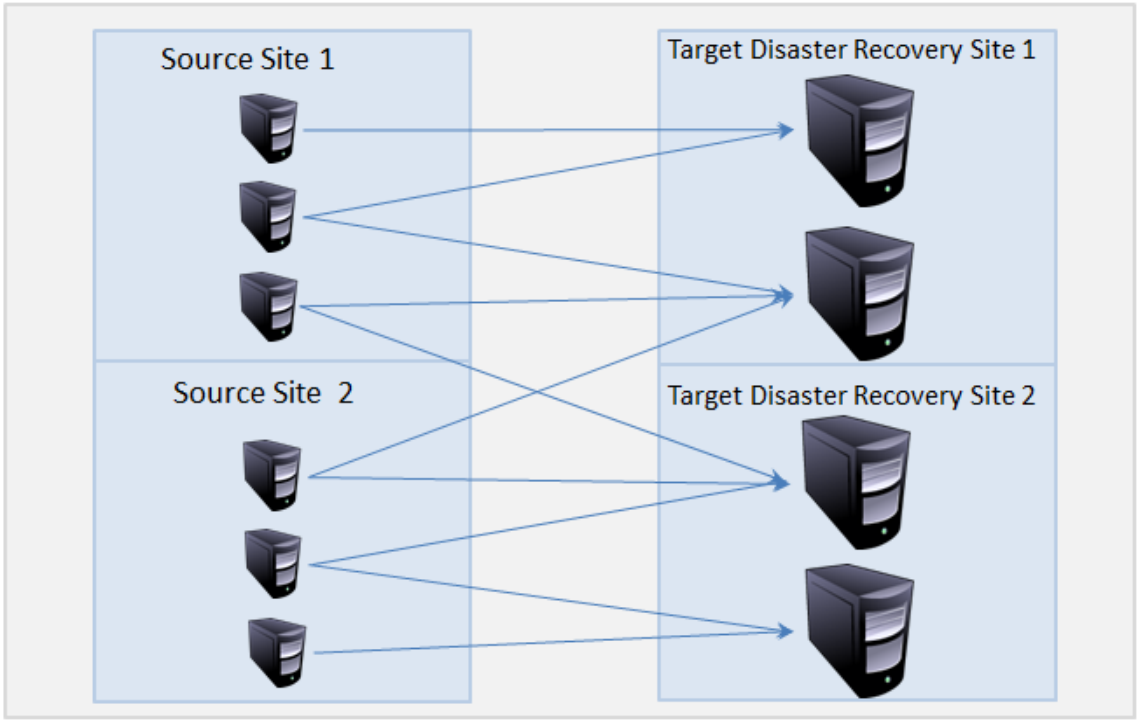
- **Multi-Point to Point.** Replicates protected machines from multiple source cores to a single target core.

Figure 6. Multi-point to point configuration



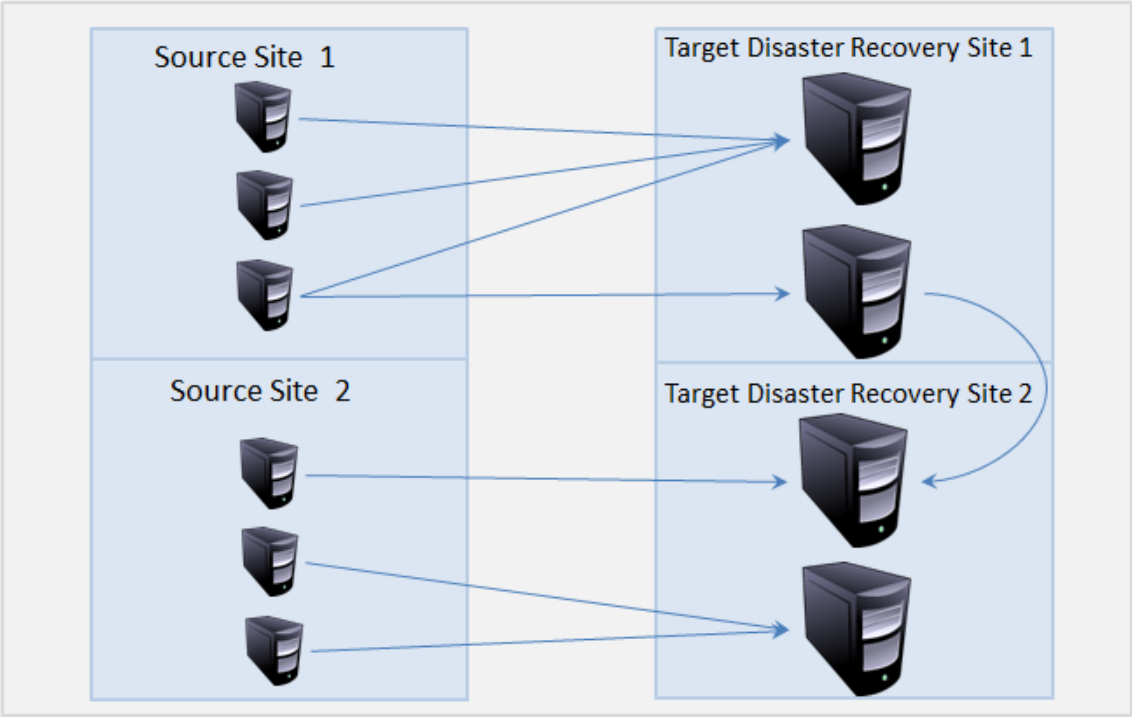
- **Point to Multi-point.** Replicates a protected machine from a single source core to more than one target core.

Figure 7. Point to Multi-point configuration



- **Multi-hop.** Replicates a replicated protected machine from one target core to another target core, producing an additional failover or recovery option.

Figure 8. Multi-hop configuration



Understanding seed drives

Replication begins with seeding: the initial transfer of deduplicated base images and incremental snapshots of the protected machines, which can add up to hundreds or thousands of gigabytes of data. Initial replication can be seeded to the target core over a network connection or by creating and saving a seed drive on external media and then transferring the initial data to the target core. This seed drive is typically useful for large sets of data or sites with slow links.

NOTE: While it is possible to seed the base data over a network connection, it is not recommended. Initial seeding involves potentially very large amounts of data, which could overwhelm a typical WAN connection. For example, if the seed data measures 10 GB and the WAN link transfers 24 Mbps, the transfer could take approximately one hour to complete.

The data in the seeding archive is compressed, optionally, encrypted, and deduplicated. If the total size of the archive is larger than the space available on the removable media, the archive can span across multiple devices based on the available space on the media After replication is established, all recovery points created after the establishment of the seed drive replicate to the target site in advance of the transfer of archived data. When the seed drive is consumed, the archived data synchronizes with the replicated recovery points present on the target core repository.

Seeding is a two-part process (also known as copy-consume):

- The first part involves copying, which is the writing of the initial replicated data to a removable media source. There are two copying options: duplicating not-yet replicated recovery points from the source core to a local removable storage device, such as a USB drive, or duplicating all existing recovery points [Build RP chains (fix orphans) option]. After copying is complete, you must then transport the drive from the source core location to the remote target core location.

- The second part is consuming, which occurs when a target core receives the transported drive and copies the replicated data to the repository. The target core then consumes the recovery points and uses them to form replicated agents (protected machines).

  NOTE: While replication of incremental snapshots can occur between the source and target cores before seeding is complete, the replicated snapshots transmitted from the source to the target will remain "orphaned" until the initial data is consumed, and they are combined with the replicated base images. For more information about orphaned recovery points, see Deleting an orphaned recovery point chain.

Because large amounts of data need to be copied to the portable storage device, an eSATA, USB 3.0, or other high-speed connection to the portable storage device is recommended.

Seeding is an option available within the procedure for replicating to a self-managed target core and the procedure for replicating to a core managed by a third party (such as a managed service provider or MSP). For more information, see Replicating to a self-managed target core and Replicating to a third-party target core, respectively.

Understanding failover and failback in AppAssure

In the case of a severe outage in which your source core and protected machines fail, AppAssure supports failover and failback in replicated environments. Failover refers to switching to a redundant or standby target AppAssure Core upon system failure or abnormal termination of a source core and associated protected machines. The main goal of failover is to launch a new protected machine identical to the failed machine that was protected by the failed source core. The secondary goal is to switch the target core into a new mode so that the target core protects the failover protected machine in the same way as the source core protected the initial machine before the failure. The target core can recover instances from replicated protected machines and immediately commence protection on the failed-over machines.

Failback is the process of restoring a protected machine and core back to their original states (before failure). The primary goal of failback is to restore the protected machine (in most cases, this is a new machine replacing a failed machine) to a state identical to the latest state of the new, temporary protected machine. When restored, it is protected by a restored source core. Replication is also restored, and the target core acts as a replication target again. For more information, see Understanding failover and failback.

Performance considerations for replicated data transfer

If the bandwidth between the source core and the target core cannot accommodate the transfer of stored recovery points, replication begins with seeding the target core with base images and recovery points from the selected servers protected on the source core. The seeding process can be performed at any time, as part of the initial transfer of data to serve as the foundation for regularly scheduled replication, or in the case of re-instating replication for a previously replicated machine whose replication had been paused or deleted. In this case, the Build RP Chain option would let you copy not-yet replicated recovery points to a see drive.

When preparing for replication, you should consider the following factors:

- **Change rate.** The change rate is the rate at which the amount of protected data is accumulated. The rate depends on the amount of data that changes on protected volumes and the protection interval of the volumes. If a set of blocks change on the volume, reducing the protection interval reduces the change rate.
- **Bandwidth.** The bandwidth is the available transfer speed between the source core and the target core. It is crucial that the bandwidth be greater than the change rate for replication to keep up with the recovery points created by the snapshots. Due to the amount of data transmitted from core to core,

multiple parallel streams may be required to perform at wire speeds up to the speed of a 1GB Ethernet connection.

> **NOTE:** Bandwidth specified by the ISP is the total available bandwidth. The outgoing bandwidth is shared by all devices on the network. Make sure that there is enough free bandwidth for replication to accommodate the change rate.

- **Number of protected machines.** It is important to consider the number of machines protected per source core and how many you plan to replicate to the target. AppAssure lets you perform replication on a per-protected server basis, so you can choose to replicate certain servers. If all protected servers must be replicated, this drastically affects the change rate, particularly if the bandwidth between the source and target cores is insufficient for the amount and size of the recovery points being replicated.

Depending on your network configuration, replication can be a time-consuming process.

The Maximum Change Rate for WAN Connection Types is shown in the table below with examples of the necessary bandwidth per gigabyte for a reasonable change rate.

Table 95. Examples of bandwidth per gigabyte

Broadband	Bandwidth	Max Change Rate
DSL	768 Kbps and up	330 MB per hour
Cable	1 Mbps and up	429 MB per hour
T1	1.5 Mbps and up	644 MB per hour
Fiber	20 Mbps and up	8.38 GB per hour

> **NOTE:** For optimum results, you should adhere to the recommendations listed in the table above.

If a link fails during data transfer, replication resumes from the previous failure point of the transfer once link functionality is restored.

About replication and encrypted recovery points

While the seed drive does not contain backups of the source core registry and certificates, the seed drive does contain encryption keys from the source core if the recovery points being replicated from source to target are encrypted. The replicated recovery points remain encrypted after they are transmitted to the target core. The owners or administrators of the target core need the passphrase to recover the encrypted data.

About retention policies for replication

Retention policies on the source and target cores are not synchronized. Rollup and ad-hoc deletion perform independently on each core on initial action as well as during the nightly jobs.

For more information on retention policies, see Managing retention policies.

Configuring replication

To replicate data using AppAssure, you must configure the source and target cores for replication. After you configure replication, you can then replicate protected machine data, monitor and manage replication, and perform recovery.

Performing replication in AppAssure involves performing the following operations:

- **Set up a repository on the target core.** For more information on adding a repository to the target core, see Creating a repository.
- **Configure self-managed replication.** For more information on replicating to a self-managed target core, see Replicating to a self-managed target core.
- **Configure third-party replication.** For more information on replicating to a third-party target core, see Replicating to a third-party target core.
- **Replicate an existing protected machine.** For more information on replicating a machine that is already protected by the source core, see Adding a machine to existing replication.
- **Consume the seed drive.** For more information on consuming seed drive data on the target core, see Consuming the seed drive on a target core.
- **Set the replication priority for a protected machine.** For more information on prioritizing the replication of protected machines, see Setting replication priority for a protected machine.
- **Set a replication schedule for a protected machine.** For more information on setting a replication schedule, see Scheduling replication.
- **Monitor replication as needed.** For more information on monitoring replication, see Monitoring replication.
- **Manage replication settings as needed.** For more information on managing replication settings, see Managing replication settings.
- **Recover replicated data in the event of disaster or data loss.** For more information on recovering replicated data, see Recovering replicated data.

Replicating to a self-managed target core

A self-managed core is a core to which you have access, often because it is managed by your company at an off-site location. Replication can be completed entirely on the source core, unless you choose to seed your data. Seeding requires that you consume the seed drive on the target core after you configure replication on the source core.

NOTE: This configuration applies to Replication to an Off-site Location and to Mutual Replication. The AppAssure Core must be installed on all source and target machines. If you are configuring AppAssure for Multi-Point to Point replication, you must perform this task on all source cores and the one target core.

Complete the steps in the following procedure to configure your source core to replicate to a self-managed target core.

To replicate a self-managed target core

1. Navigate to the AppAssure Core, and then click the Replication tab or icon.
2. On the Replication tab, click **Add Target Core**.

   The Replication Wizard appears.
3. On the Target Core page of the Replication Wizard, select **I have my own Target Core**, and then enter the information as described in the following table.

   NOTE: With AppAssure, you can replicate to the Dell DL-4000. For information about this feature, see the *Dell AppAssure Release Notes* or the AppAssure Knowledge Base.

Table 96. Target core information

Text Box	Description
Host Name	Enter the host name or IP address of the Core machine to which you are replicating.
Port	Enter the port number on which the AppAssure Core will communicate with the machine. The default port number is *8006*.
User Name	Enter the user name for accessing the machine.
Password	Enter the password for accessing the machine.

- If the Core you want to add has been paired with this source core previously, you can do the following:
  a Select **Use an existing target core**.
  b Select the target core from the drop-down list.
  c Click **Next**.
  d Skip to Step 7.

4 Click **Next**.

NOTE: If no repository exists on the target core, a warning appears notifying you that you can pair the source core with the target core, but that you are unable to replicate agents (protected machines) to this location until a repository is established. For information about how to set up a repository to a core, see Creating a repository.

5 On the Details page, enter a name for this replication configuration; for example, SourceCore1.

- If you are re-initiating or repairing a previous replication configuration, select **My Core has been migrated and I would like to repair replication**.

6 Click **Next**.

7 On the Machines page, select the protected machines you want to replicate, and then use the drop-down lists in the Repository column to select a repository for each protected machine.

8 If you want to perform the seeding process for the transfer of the base data, complete the following steps:

NOTE: Because large amounts of data need to be copied to the portable storage device, an eSATA, USB 3.0, or other high-speed connection to the portable storage device is recommended.

a On the Machines page, select *Use a seed drive to perform initial transfer.*
   - If you currently have one or more protected machines replicating to a target core, you can include these protected machines on the seed drive by selecting **With already replicated**.

b Click **Next**.

c On the Seed Drive Location page, use the **Location type** drop-down list to select from the following destination types:
   - Local
   - Network
   - Cloud

d Enter the details for the archive as described in the following table based on the location type you selected in Step c.

Table 97. Archive details

Option	Text Box	Description
Local	Output location	Enter the location for the output. It is used to define the location path where you want the archive to reside; for example, d:\work\archive.
Network	Output location	Enter the location for the output. It is used to define the location path where you want the archive to reside; for example, \\servername\sharename.
	User Name	Enter a user name. It is used to establish logon credentials for the network share.
	Password	Enter a password for the network path. It is used to establish logon credentials for the network share.
Cloud	Account	Select an account from the drop-down list. **NOTE:** To select a cloud account, you must first have added it in the Core Console. For more information, see Adding a cloud account.
	Container	Select a container associated with your account from the drop-down menu.
	Folder Name	Enter a name for the folder in which the archived data is to be saved. The default name is AppAssure-5-Archive-[DATE CREATED]-[TIME CREATED]

e Click **Next**.

f On the Seed Drive Options page, enter the information as described in the following table.

Table 98. Seed drive options

Item	Description
Maximum Size	Large archives of data can be divided into multiple segments. Select the maximum size of the segment you want to reserve for creating the seed drive by doing one of the following: • Select **Entire Target** to reserve all available space in the path provided on the Seed Drive Location page for future use (for example, if the location is D:\work\archive, all of the available space on the D: drive is reserved if required for copying the seed drive, but is not reserved immediately after starting the copying process). • Select the blank text box, enter an amount, and then select a unit of measurement from the drop-down list to customize the maximum space you want to reserve.
Customer ID (optional)	Optionally, enter the customer ID that was assigned to you by the service provider.
Recycle action	In the event the path already contains a seed drive, select one of the following options: • **Do not reuse.** Does not overwrite or clear any existing seed data from the location. If the location is not empty, the seed drive write will fail. • **Replace this Core.** Overwrites any pre-existing seed data pertaining to this core but leaves the data for other cores intact. • **Erase completely.** Clears all seed data from the directory before writing the seed drive.
Comment	Enter a comment that describes the seed drive.
Add all Agents to Seed Drive	Select this option to replicate all protected machines on the source core using the seed drive. This option is selected by default.

Table 98. Seed drive options

Item	Description
Build RP chains (fix orphans)	Select this option to replicate the entire recovery point chain to the seed drive. This option is selected by default. **NOTE:** Typical seeding in AppAssure 5.4 replicates only the latest recovery point to the seed drive, which reduces the amount of time and space required for creating the seed drive. Opting to build recovery point (RP) chains to the seed drive requires enough space on the seed drive to store the latest recovery points from the specified protected machine or machines, and may take additional time to complete the task.
Use compatible format	Select this option to create the seed drive in a format that is compatible with both new and older versions of the AppAssure Core. **NOTE:** The current seed drive format is not compatible with 5.3 versions of the Core. Archives created in AppAssure 5.3.x are compatible with AppAssure 5.4.x and later.

g Do one of the following:

- If you cleared the **Add all Agents to Seed Drive** check box, click **Next**.
- If you selected **Add all Agents to Seed Drive**, go to Step 9.

h On the Machines page, select the protected machines you want to replicate to the target core using the seed drive.

9 Click **Finish**.

10 If you created a seed drive, send it to your target core.

The pairing of the source core to the target core is complete. Replication begins, but produces orphaned recovery points on the target core until the seed drive is consumed and provides the necessary base images.

Replicating to a third-party target core

A third-party core is a target core that it managed and maintained by an MSP. Replicating to a core managed by a third party does not require the customer to have access to the target core.

The process of replicating to a third-party core involves tasks that must be completed by the customer as well as the third party. After a customer submits a request for replication on the source core or cores, the MSP must complete the configuration on the target core by reviewing the request.

NOTE: This configuration applies to Hosted and Cloud Replication. The AppAssure Core must be installed on all source core machines. If you are configuring AppAssure for Multi-Point to Point replication, you must perform this task on all source cores.

To replicate to a target core managed by a third party, complete the following tasks:

1 Submitting a replication request to a third-party service provider

2 Reviewing a replication request from a customer or Ignoring a replication request from a customer

Submitting a replication request to a third-party service provider

If you are an end user who subscribes to a core managed by a third party, such as an MSP, complete the steps in this procedure to submit a replication request to your third-party service provider.

To submit a replication request to a third-party service provider

1 Navigate to the AppAssure Core, and then click the Replication tab or the replication symbol in the left column.

2 On the Replication tab, click **Add Target Core**.

The Replication Wizard appears.

3 On the Target Core page of the Replication Wizard, select **I have a subscription to a third-party providing off-site backup and disaster recovery services**, and then enter the information as described in the following table.

Table 99. Third-party target core information

Text Box	Description
Host Name	Enter the host name, IP address, or FQDN for the third-party core machine.
Port	Enter the port number that was given to you by your third-party service provider. The default port number is *8006*.

- If the Core you want to add has been paired with this source core previously, you can do the following:
  a Select **Use an existing target core**.
  b Select the target core from the drop-down list.
  c Click **Next**.
  d Skip to Step 7.

4 Click **Next**.

5 On the Details page, enter the information as described in the following table.

Table 100. Third-party target core details

Text Box	Description
Email Address	Enter the email address associated with your third-party service subscription.
Customer ID (optional)	Optionally, enter the customer ID that was assigned to you by the service provider.

6 Click **Next**.

7 On the Machines page, select the protected machines you want to replicate to the third-party core.

8 If you want to perform the seeding process for the transfer of base data, complete the following steps.

NOTE: Because large amounts of data need to be copied to the portable storage device, an eSATA, USB 3.0, or other high-speed connection to the portable storage device is recommended.

a On the Machines page, select *Use a seed drive to perform initial transfer.*

- If you currently have one or more protected machines replicating to a target core, you can include these machines on the seed drive by selecting **With already replicated**.

b Click **Next**.

c On the Seed Drive Location page, use the **Location type** drop-down list to select from the following destination types:

- Local
- Network
- Cloud

d Enter the details for the archive as described in the following table based on the location type you selected in Step c.

Table 101. Archive details

Option	Text Box	Description
Local	Output location	Enter the location for the output. It is used to define the location path where you want the archive to reside; for example, d:\work\archive.
Network	Output location	Enter the location for the output. It is used to define the location path where you want the archive to reside; for example, \\servername\sharename.
	User Name	Enter a user name. It is used to establish logon credentials for the network share.
	Password	Enter a password for the network path. It is used to establish logon credentials for the network share.
Cloud	Account	Select an account from the drop-down list. **NOTE:** To select a cloud account, you must first have added it in the Core Console. For more information, see Adding a cloud account.
	Container	Select a container associated with your account from the drop-down menu.
	Folder Name	Enter a name for the folder in which the archived data is to be saved. The default name is AppAssure-5-Archive-[DATE CREATED]-[TIME CREATED]

e Click **Next**.

f On the Seed Drive Options page, enter the information as described in the following table.

Table 102. Seed drive options

Item	Description
Maximum Size	Large archives of data can be divided into multiple segments. Select the maximum amount of space you want to reserve for creating the seed drive by doing one of the following: • Select **Entire Target** to reserve all available space in the path provided on the Seed Drive Location page (for example, if the location is D:\work\archive, all of the available space on the D: drive is reserved). • Select the blank text box, enter an amount, and then select a unit of measurement from the drop-down list to customize the maximum space you want to reserve.
Customer ID (optional)	Optionally, enter the customer ID that was assigned to you by the service provider.

Table 102. Seed drive options

Item	Description
Recycle action	In the event the path already contains a seed drive, select one of the following options: • **Do not reuse.** Does not overwrite or clear any existing seed data from the location. If the location is not empty, the seed drive write will fail. • **Replace this Core.** Overwrites any pre-existing seed data pertaining to this core but leaves the data for other cores intact. • **Erase completely.** Clears all seed data from the directory before writing the seed drive.
Comment	Enter a comment that describes the seed drive.
Add all Agents to Seed Drive	Select this option to replicate all protected machines on the source core using the seed drive. This option is selected by default.
Build RP chains (fix orphans)	Select this option to replicate the entire recovery point chain to the seed drive. This option is selected by default. **NOTE:** Typical seeding in AppAssure 5.4 replicates only the latest recovery point to the seed drive, which reduces the amount of time and space required for creating the seed drive. Opting to build recovery point (RP) chains to the seed drive requires enough space on the seed drive to store the latest recovery points from the specified protected machine or machines, and may take additional time to complete the task.
Use compatible format	Select this option to create the seed drive in a format that is compatible with both new and older versions of the AppAssure Core. **NOTE:** The current seed drive format is not compatible with 5.3 versions of the Core. Archives created in AppAssure 5.3.x are compatible with AppAssure 5.4.x and later.

g Do one of the following:

- If you cleared the **Add all Agents to Seed Drive** check box, click **Next**.
- If you selected **Add all Agents to Seed Drive**, go to Step 9.

h On the Machines page, select the protected machines you want to replicate to the target core using the seed drive.

9 Click **Finish**.

10 If you created a seed drive, send it as directed by your third-party service provider.

Reviewing a replication request from a customer

After an end user completes the procedure Submitting a replication request to a third-party service provider, a replication request is sent from the source core to the third-party target core. As the third party, you can review the request, and then approve it to begin replication for your customer, or you can deny it to prevent replication from occurring.

Choose from the following options:

- Approving a replication request
- Denying a replication request

Approving a replication request

Complete the following procedure to approve a replication request on a third-party target core.

To approve a replication request

1. On the target core, open the AppAssure Core Console, and then click the Replication tab or icon.
2. On the Replication tab, click **Pending Requests (#)**.

   The Pending Replication Requests section appears.
3. Under Pending Replication Requests, click the drop-down menu next to the request you want to review, and then click **Review**.

   The Review Replication Request window appears.

   NOTE: The information that appears in the Source Core Identity section of this window is determined by the request completed by the customer.
4. Under Source Core Identity, do one of the following:
   - Select Replace an existing replicated Core, and then select a core from the drop-down list.
   - Select Create a new source Core, and then confirm that the Core Name, customer Email Address, and Customer ID, provided are correct. Edit the information as necessary.
5. Under Agents, select the machines to which the approval applies, and then use the drop-down lists in the Repository column to select the appropriate repository for each machine.
6. Optionally, in the Comment text box, enter a description or message to include in the response to the customer.
7. Click **Send Response**.

   Replication is accepted.

Denying a replication request

Complete the steps in the following procedure to deny a replication request sent to a third-party core from a customer.

To deny a request without reviewing it, see Ignoring a replication request from a customer.

To deny a replication request

1. On the target core, open the AppAssure Core Console, and then click the Replication tab or icon.
2. On the Replication tab, click **Pending Requests (#)**.

   The Pending Replication Requests section appears.
3. Under Pending Replication Requests, click the drop-down menu next to the request you want to review, and then click **Review**.

   The Review Replication Request window appears.
4. Click **Deny**.

   Replication is denied. Notification of denial appears under Alerts on the Events tab of the source core.

Ignoring a replication request from a customer

As a third-party service provider of a target core, you have the option to ignore a request for replication sent from a customer. This option could be used if a request was sent by mistake or if you want to deny a request without reviewing it.

For more information about replication requests, see Reviewing a replication request from a customer.

Complete the following procedure to ignore a replication request from a customer.

To ignore a replication request from a customer

1. On the target core, open the AppAssure Core Console, and then click the Replication tab or icon.
2. On the Replication tab, click **Pending Requests (#)**.

   The Pending Replication Requests section appears.
3. Under Pending Replication Requests, click the drop-down menu next to the request you want to ignore, and then click **Ignore**.
4. On the Ignoring request dialog box, click **Yes** to confirm the command.

   Notification that the request has been ignored is sent to the source core and the request is removed from the target core Replication tab.

Adding a machine to existing replication

After replication is established between a source and target core, it is possible to add new protected machines to replicate to the target. Complete the steps in the following procedure to add a new protected machine to a paired target core for replication.

For more information about replication, see Understanding replication and Replicating to a self-managed target core.

To add a machine to existing replication

1. On the AppAssure Core Console, click the Replication tab or icon.
2. Click the drop-down menu next to the target core to which you want to replicate a new machine, and then click **Add Machines**.

   The Replication Wizard opens to the Machines page.
3. On the Machines page, select the protected machines you want to replicate, and then use the drop-down lists in the Repository column to select a repository for each protected machine.
4. If you want to perform the seeding process for the transfer of the base data, complete the following steps:

   NOTE: Because large amounts of data need to be copied to the portable storage device, an eSATA, USB 3.0, or other high-speed connection to the portable storage device is recommended.

   a. On the Machines page, select *Use a seed drive to perform initial transfer.*
      - If you currently have one or more protected machines replicating to a target core, you can include these machines on the seed drive by selecting **With already replicated**.

   b. Click **Next**.

c On the Seed Drive Location page, use the **Location type** drop-down list to select from the following destination types:

- Local
- Network
- Cloud

d Enter the details for the archive as described in the following table based on the location type you selected in Step c.

Table 103. Archive details

Option	Text Box	Description
Local	Output location	Enter the location for the output. It is used to define the location path where you want the archive to reside; for example, d:\work\archive.
Network	Output location	Enter the location for the output. It is used to define the location path where you want the archive to reside; for example, \\servername\sharename.
	User Name	Enter a user name. It is used to establish logon credentials for the network share.
	Password	Enter a password for the network path. It is used to establish logon credentials for the network share.
Cloud	Account	Select an account from the drop-down list. NOTE: To select a cloud account, you must first have added it in the Core Console. For more information, see Adding a cloud account.
	Container	Select a container associated with your account from the drop-down menu.
	Folder Name	Enter a name for the folder in which the archived data is to be saved. The default name is AppAssure-5-Archive-[DATE CREATED]-[TIME CREATED]

e Click **Next**.

f On the Seed Drive Options page, enter the information as described in the following table.

Table 104. Seed drive options

Item	Description
Maximum Size	Large archives of data can be divided into multiple segments. Select the maximum amount of space you want to reserve for creating the seed drive by doing one of the following: • Select **Entire Target** to reserve all available space in the path provided on the Seed Drive Location page (for example, if the location is D:\work\archive, all of the available space on the D: drive is reserved). • Select the blank text box, enter an amount, and then select a unit of measurement from the drop-down list to customize the maximum space you want to reserve.
Customer ID (optional)	Optionally, enter the customer ID that was assigned to you by the service provider.

Table 104. Seed drive options

Item	Description
Recycle action	In the event the path already contains a seed drive, select one of the following options: • **Do not reuse.** Does not overwrite or clear any existing seed data from the location. If the location is not empty, the seed drive write will fail. • **Replace this Core.** Overwrites any pre-existing seed data pertaining to this core but leaves the data for other cores intact. • **Erase completely.** Clears all seed data from the directory before writing the seed drive.
Comment	Enter a comment that describes the seed drive.
Add all Agents to Seed Drive	Select this option to replicate all protected machines on the source core using the seed drive. This option is selected by default.
Build RP chains (fix orphans)	Select this option to replicate the entire recovery point chain to the seed drive. This option is selected by default. **NOTE:** Typical seeding in AppAssure 5.4 replicates only the latest recovery point to the seed drive, which reduces the amount of time and space required for creating the seed drive. Opting to build recovery point (RP) chains to the seed drive requires enough space on the seed drive to store the latest recovery points from the specified protected machine or machines, and may take additional time to complete the task.
Use compatible format	Select this option to create the seed drive in a format that is compatible with both new and older versions of the AppAssure Core. **NOTE:** The current seed drive format is not compatible with 5.3 versions of the Core. Archives created in AppAssure 5.3.x are compatible with AppAssure 5.4.x and later.

g Do one of the following:

- If you cleared the **Add all Agents to Seed Drive** check box, click **Next**.
- If you selected **Add all Agents to Seed Drive**, go to Step 5.

h On the Machines page, select the protected machines you want to replicate to the target core using the seed drive.

5 Click **Finish**.

Consuming the seed drive on a target core

Complete the follow procedure to consume the data from the seed drive on the target core.

NOTE: This procedure is only necessary if a seed drive was created as part of Replicating to a self-managed target core or Replicating to a third-party target core.

To consume the seed drive on a target core

1. If the seed drive was saved to a portable storage device, such as a USB drive, connect the drive to the target core.
2. On the target core, open the AppAssure Core Console, and then click the Replication tab or icon.
3. On the Replication tab, under Incoming Replication, click the drop-down menu for the correct source core, and then click **Consume**.

   The Consume window appears.
4. For **Location type**, select one of the following options from the drop-down list:
   - Local
   - Network
   - Cloud
5. Enter the details for the archive as described in the following table based on the location type you selected in Step 4.

   Table 105. Archive details

Option	Text Box	Description
Local	Location	Enter the path for the archive.
Network	Location	Enter the path for the archive.
	User Name	Enter the user name. It is used to establish logon credentials for the network share.
	Password	Enter the password for the network path. It is used to establish logon credentials for the network share.
Cloud	Account	Select an account from the drop-down list. **NOTE:** To select a cloud account, you must first have added it in the Core Console. For more information, see Adding a cloud account.
	Container	Select a container associated with your account from the drop-down menu.
	Folder Name	Enter the name of the folder in which the archived data is saved; for example, AppAssure-5-Archive-[DATE CREATED]-[TIME CREATED]

6. Click **Check File**.

   The Core searches for the file.

   After finding the file, the following text boxes appear in the Consume window pre-populated with the data gathered from Step 4, Step 5, and the file. The Date Range displays the dates of the oldest and newest recovery points contained in the seed drive. Any comments entered when the seed drive was created are automatically imported.
7. On the Consume window, under Agents, select the machines for which you want to consume data.
8. Click **Consume**.
9. To monitor the progress of consuming data, click the Events tab.

Abandoning a seed drive

If you create a seed drive with the intent to consume it on the target core but choose not to send it to the remote location, a link for the Outstanding Seed Drive remains on the source core replication tab. In this case, you may want to abandon the outstanding seed drive in favor of different or more current data.

Complete the steps in the following procedure to abandon an outstanding seed drive.

NOTE: This procedure removes the link to the outstanding drive from the AppAssure Core Console on the source core. It does not remove the drive from the storage location on which it is saved. Abandoning the drive automatically replicates all recovery points from the abandoned seed drive during the next replication job.

To abandon a seed drive

1. On the source core, open the AppAssure Core Console, and then click the Replication tab or icon.
2. On the replication tab, click **Outstanding Seed Drives (#)**.

   The Outstanding seed drives section appears. It includes the name of the remote target core, the date and time on which the seed drive was created, and the data range of the recovery points included on the seed drive.
3. Under Outstanding seed drives, click the drop-down menu for the drive you want to abandon, and then click **Abandon**.

   The Outstanding Seed Drives window appears.
4. On the Outstanding Seed Drives window, click **Yes** to confirm.

   The seed drive is removed.

   If no more seed drives exist on the source core, the Outstanding Seed Drives (#) link and Outstanding seed drives section are removed from the Replication tab.

Managing replication settings

AppAssure lets you monitor, schedule, and adjust replication at the overall, core, and protected machine levels.

You can edit the following replication settings:

- To schedule replication jobs, see Scheduling replication.
- To create a seed drive of a protected machine that is already paired for replication, see Using the Copy function to create a seed drive
- To monitor the progress of a replication job, see Monitoring replication.
- To pause or resume a paused replication job, see Pausing and resuming replication.
- To force replication of an incoming or outgoing protected machine, see Forcing replication.
- To manage settings for all target cores and replication procedures, see Managing settings for outgoing replication.
- To manage settings for an individual target core, see Changing target core settings.
- To manage priority settings for an individual protected machine being replicated to a target core, see Setting replication priority for a protected machine.

Scheduling replication

You can use the replication scheduler to set a time, such as on a specific day or during off-peak hours, for transferring replicated data from the source core to the target.

To schedule replication

1. On the AppAssure Core Console, click the Replication tab or icon.
2. On the Replication tab, click the drop-down menu next to the core for which you want schedule replication, and then click **Schedule**.

   The Replication Schedule for [CoreName] opens.
3. Select from one of the following three options:
   - **At All Times.** Replicates after every new snapshot, checksum check, and attachability check, and after the nightly jobs complete.
   - **Daily (Start Replication only during the specified time period).** Begins replicating only within the time range provided.
     a. In the **From** text box, enter the earliest time at which replication should begin.
     b. In the **To** text box, enter the latest time at which replication should begin.

   NOTE: If replication is in progress when the scheduled time ends, the replication job completes after the allotted time period.

   - **Custom.** Replicates only within the time range provided at the time of week specified.
     a. Next to Weekdays, in the **From** text box, enter the earliest time at which replication should occur on a weekday; and then in the **To** text box, enter the latest time at which replication should occur on a weekday.
     b. Next to Weekends, in the **From** text box, enter the earliest time at which replication should occur on weekends; and then in the **To** text box, enter the latest time at which replication should occur on weekends.
4. Click **Save**.

   The schedule is applied to all replication to the selected target core.

Using the Copy function to create a seed drive

If you chose not to create a seed drive when you configured replication, you can create a seed drive using the Copy function in the protected machine drop-down menu.

To use the Copy function to create a seed drive

1. In the AppAssure Core Console of the source core, click the Replication tab.
2. Under Outgoing Replication, expand the Core that protects the machine for which you want to create a seed drive, and then select the machine.
3. In the table heading, click the drop-down menu, and then click **Copy**.
4. On the Seed Drive Location page, use the **Location type** drop-down list to select from the following destination types:
   - Local
   - Network
   - Cloud
5. Enter the details for the archive as described in the following table based on the location type you selected in Step c.

Table 106. Archive details

Option	Text Box	Description
Local	Output location	Enter the location for the output. It is used to define the location path where you want the archive to reside; for example, d:\work\archive.
Network	Output location	Enter the location for the output. It is used to define the location path where you want the archive to reside; for example, \\servername\sharename.
	User Name	Enter a user name. It is used to establish logon credentials for the network share.
	Password	Enter a password for the network path. It is used to establish logon credentials for the network share.
Cloud	Account	Select an account from the drop-down list. **NOTE:** To select a cloud account, you must first have added it in the Core Console. For more information, see Adding a cloud account.
	Container	Select a container associated with your account from the drop-down menu.
	Folder Name	Enter a name for the folder in which the archived data is to be saved. The default name is AppAssure-5-Archive-[DATE CREATED]-[TIME CREATED]

6 Click **Next**.

7 On the Seed Drive Options page, enter the information as described in the following table.

Table 107. Seed drive options

Item	Description
Maximum Size	Large archives of data can be divided into multiple segments. Select the maximum amount of space you want to reserve for creating the seed drive by doing one of the following: • Select **Entire Target** to reserve all available space in the path provided on the Seed Drive Location page (for example, if the location is D:\work\archive, all of the available space on the D: drive is reserved). • Select the blank text box, enter an amount, and then select a unit of measurement from the drop-down list to customize the maximum space you want to reserve.
Customer ID (optional)	Optionally, enter the customer ID that was assigned to you by the service provider.
Recycle action	In the event the path already contains a seed drive, select one of the following options: • **Do not reuse.** Does not overwrite or clear any existing seed data from the location. If the location is not empty, the seed drive write will fail. • **Replace this Core.** Overwrites any pre-existing seed data pertaining to this core but leaves the data for other cores intact. • **Erase completely.** Clears all seed data from the directory before writing the seed drive.
Comment	Enter a comment that describes the seed drive.
Add all Agents to Seed Drive	Select this option to replicate all protected machines on the source core using the seed drive. This option is selected by default.

Table 107. Seed drive options

Item	Description
Build RP chains (fix orphans)	Select this option to replicate the entire recovery point chain to the seed drive. This option is selected by default. **NOTE:** Typical seeding in AppAssure 5.4.x replicates only the latest recovery point to the seed drive, which reduces the amount of time and space required for creating the seed drive. Opting to build recovery point (RP) chains to the seed drive requires enough space on the seed drive to store the latest recovery points from the specified protected machine or machines, and may take additional time to complete the task.
Use compatible format	Select this option to create the seed drive in a format that is compatible with both new and older versions of the AppAssure Core. **NOTE:** The current seed drive format is not compatible with 5.3.x versions of the Core. Archives created in AppAssure 5.3.x are compatible with AppAssure 5.4.x and later.

8 Do one of the following:

- If you cleared the **Add all Agents to Seed Drive** check box, click **Next**.
- If you selected **Add all Agents to Seed Drive**, go to Step 10.

9 On the Machines page, select the protected machines you want to create a seed drive.

10 Click **Finish**.

Monitoring replication

When replication is set up, you can monitor the status of replication tasks for the source and target cores. You can refresh status information, view replication details, and more.

To monitor replication

1 In the Core Console, click the Replication tab or icon.

2 On this tab, you can view information about and monitor the status of replication tasks as described in the following table.

Table 108. Replication tasks

Section	Description	Available Actions
Pending Replication Requests	Lists your customer ID, email address, and host name when a replication request is submitted to a third-party service provider. It is listed here until the request is accepted by the MSP.	In the drop-down menu, click **Ignore** to ignore or reject the request.
Outstanding Seed Drives	Lists seed drives that have been written but not yet consumed by the target core. It includes the remote core name, date on which it was created, and the date range.	In the drop-down menu, click **Abandon** to abandon or cancel the seed process.

Table 108. Replication tasks

Section	Description	Available Actions
Outgoing Replication	Lists all target cores to which the source core is replicating. It includes the target core name, the state of existence, the number of machines being replicated, and the progress of a replication transmission.	On a source core, in the drop-down menu, you can select the following options: • **Details.** Lists the ID, URL, display name, state, customer ID, email address, and comments for the replicated core. • **Change Settings.** Lists the display name and lets you edit the host and port for the target core. • **Delete.** Lets you delete the target core from the source core. Doing so ceases all replication to this core. • **Schedule.** Lets you set a customized schedule for replication to this target core. • **Add Machines.** Lets you select a host from a drop-down list, select protected machines for replication, and create a seed drive for the new protected machine's initial transfer.
Incoming Replication	Lists all source machines from which the target receives replicated data. It includes the remote core name, state, machines, and progress.	On a target core, in the drop-down menu, you can select the following options: • **Details.** Lists the ID, host name, customer ID, email address, and comments for the replicated core. • **Consume.** Consumes the initial data from the seed drive and saves it to the local repository. • **Delete.** Lets you delete the source core from the target core. Doing so ceases all replication from this core.

Pausing and resuming replication

You can pause replication temporarily for the source (outgoing) or target (incoming) cores.

To pause and resume replication

1. In the Core Console, click the Replication tab.
2. Use the bracket (>) to expand the appropriate source or target core.
3. Select the protected machine or machines for which you want to pause replication, click the drop-down menu next to Agent Name, and then click **Pause** to pause replication temporarily.

   The status of the machine reads as “Established (paused)” in the State column.
4. To resume replication, select the paused agent, click the Agent Name drop-down menu, and then click **Resume**.

Forcing replication

Complete the steps in the following procedure to force replication from either the source or the target core.

To force replication

1. On the AppAssure Core Console, click the Replication tab or icon.
2. Use the bracket (>) to expand the appropriate source or target core.
3. Do one of the following:
   - Click the drop-down menu next to the protected machine for which you want to force replication, and then click **Force**.
   - Select the protected machine or machines for which you want to force replication, click the drop-down menu next to the machine name, and then click **Force**.
4. In the dialog box, you can optionally select **restore orphaned Recovery Point chains** to repair any orphaned chains of recovery points from this machine on the target core.
5. Click **Yes**.

Managing settings for outgoing replication

The changes made to these settings affect the data transfer to all target cores associated with this source core.

To manage settings for outgoing replication

1 In the Core Console, click the Replication tab or icon.

2 Click **Settings**.

3 In the Replication Settings window, edit the replication settings as described in the following table.

Table 109. Replication settings

Option	Description
Cache lifetime (seconds)	Specify the amount of time between each target-core status request performed by the source core.
Volume image session timeout (minutes)	Specify the amount of time the source core spends attempting to transfer a volume image to the target core.
Maximum parallel streams	Specify the number of network connections permitted to be used by a single protected machine to replicate that machine's data at one time.
Maximum transfer speed (MB/s)	Specify the speed limit for transferring the replicated data.

4 Click **Save**.

Changing target core settings

AppAssure lets you change the host and port settings for individual target cores.

To change target core settings

1 On the AppAssure Core Console, click the Replication tab or icon.

2 Next to the appropriate target core, click the drop down menu, and then click **Change Settings**.

The Settings window appears.

3 Edit any of the options described in the following table.

Table 110. Target core settings

Option	Description
Host	Enter the host for the target core.
Port	Enter a port for the target core to use for communication with the source core. NOTE: The default port is 8006.

4 Click **Save**.

Setting replication priority for a protected machine

Complete the steps below to edit the settings that prioritize when a protected machine replicates.

To set replication priority for a protected machine

1. From the AppAssure Core Console, click the Replication tab or icon.
2. Next to the target core, click the bracket (>) to expand the replicating protected machines.
3. Click the drop-down menu for the protected machine you want to prioritize, and then click Settings.
4. Use the Priority drop-down list to select one of the following options. You can choose from 1 (Highest) to 10 (Lowest). The default setting is 5.
5. Click **Save**.

Removing replication

You can discontinue replication and remove protected machines from replication in several ways. The options include:

- Removing a protected machine from replication on the source core
- Removing a protected machine on the target core
- Removing a target core from replication
- Removing a source core from replication

> **NOTE:** Removing a source core results in the removal of all replicated protected machines protected by that core.

Removing a protected machine from replication on the source core

Complete the steps in this procedure to remove a protected machine from replication on the source core.

To remove a protected machine from replication on the source core

1. From the source core, open the AppAssure Core Console, and click the Replication tab or icon.
2. Expand the Outgoing Replication section.
3. In the drop-down menu for the protected machine machine that you want to remove from replication, click **Delete**.
4. In the Outgoing Replication dialog box, click **Yes** to confirm deletion.

Removing a protected machine on the target core

Complete the steps in this procedure to remove a protected machine on the target core.

To remove a protected machine on the target core

1. On the target core, open the AppAssure Core Console, and click the Replication tab or icon.
2. Expand the Incoming Replication section.
3. In the drop-down menu for the protected machine that you want to remove from replication, click **Delete**.
4. If you want to delete all replicated recovery points received from that machine as well as remove the protected machine, select **With Recovery Points**.
5. In the Incoming Replication dialog box, click **Yes** to confirm deletion.

Removing a target core from replication

Complete the steps in this procedure to remove a target core from replication.

To remove a target core from replication

1. On the source core, open the AppAssure Core Console, and click to the Replication tab or icon.
2. Under Outgoing Replication, click the drop-down menu next to the remote core that you want to delete, and click **Delete**.
3. In the Outgoing Replication dialog box, click **Yes** to confirm deletion.

Removing a source core from replication

Complete the steps in this procedure to remove a source core from replication.

NOTE: Removing a source core results in the removal of all replicated protected machines protected by that core.

To remove a source core from replication

1. On the target core, open the AppAssure Core Console, and click the Replication tab or icon.
2. Under Incoming Replication, in the drop-down menu, click **Delete**.
3. If you want to delete all replicated recovery points received from that machine as well as remove the source core, select **With Recovery Points**.
4. In the Incoming Replication dialog box, click **Yes** to confirm deletion.

Recovering replicated data

“Day-to-day” replication functionality is maintained on the source core, while only the target core is capable of completing the functions necessary for disaster recovery.

For disaster recovery, the target core can use the replicated recovery points to recover the protected machines and core. You can perform the following recovery options from the target core:

- Mount recovery points. For more information, see Mounting a recovery point.
- Roll back to recovery points. For more information, see Restoring volumes from a recovery point or Restoring volumes for a Linux machine using the command line.
- Perform a virtual machine (VM) export. For more information, see Exporting data to a Windows-based virtual machine.
- Perform a bare metal restore (BMR). For more information, see Performing a bare metal restore for Windows machines.
- Perform Failback (in the event you have a Failover/Failback replication environment set up). For more information, see Performing failback.

Understanding failover and failback

When you encounter a disaster situation in which your source core and associated protected machine have failed, you can enable failover in AppAssure to switch protection to your identical failover (target) core and launch a new (replicated) protected machine identical to the failed machine. Once your source core and protected machines have been repaired, you can then perform failback to restore the data from the failed-over core and protected machine back to the source core and protected machine. In AppAssure, failover and failback involve the following procedures.

- **Setting up your environment for failover.** See the section, Setting up an environment for failover.
- **Perform failover for the target core and associated protected machine.** See the section, Performing failover on the target core.
- **Restore a source core by performing failback.** See the section, Performing failback.

Setting up an environment for failover

Setting up your environment for failover requires that you have a source and target AppAssure Core and associated protected machine set up for replication. Complete the steps in this procedure to set up replication for failover.

To set up an environment for failover

1. Install an AppAssure Core for the source and install an AppAssure Core for the target. For more information, see the *Dell AppAssure Installation and Upgrade Guide*.
2. Install the AppAssure Agent software for any machine to be protected by the source core. For more information, see the *Dell AppAssure Installation and Upgrade Guide*.
3. Create one repository on the source core and one repository on the target core. For more information, see Creating a repository.
4. Add the machine for protection under the source core. For more information, see Protecting a machine.
5. Set up replication from the source to target core and replicate the protected machine with all recovery points. Follow the instructions in the section Replicating to a self-managed target core to add the target core to which to replicate.

Performing failover on the target core

When you encounter a disaster situation in which your source core and associated protected machines have failed, you can enable failover in AppAssure to switch protection to your identical failover (target) core. The target core becomes the only core protecting the data in your environment, and you then launch a new protected machine to temporarily replace the failed machine.

Before you perform this procedure, you must complete the steps in Setting up an environment for failover.

To perform failover on the target core

1. Navigate to the AppAssure Core Console on the target core, and click the Replication tab or icon.
2. Under Incoming Replication, expand the details of the selected source core.
3. Click the drop-down menu for the preferred protected machine, and then click **Fail Over**.

   The Fail Over dialog box appears and lists the next steps required for completing a failover.

   - To discontinue any in-progress replication tasks for this protected machine, select **Cancel replication job if running**.
4. Click **Continue**.
5. In the left navigation area, under Protected Machines, select the machine that has the associated AppAssure protected machine with recovery points.
6. Export the backup recovery point information on that protected machine to a virtual machine. For more information, see Exporting data to a Windows-based virtual machine.
7. Start the virtual machine that now includes the exported backup information. You need to wait for the device driver software to be installed.
8. Reboot the virtual machine and wait for the AppAssure Agent service to start.
9. Go back to the AppAssure Core Console for the target core and verify that the new protected machine appears in the left navigation area under Protected Machines and on the Replication tab under Incoming Replication.
10. Force multiple snapshots, and verify they complete correctly. For more information, see Forcing a snapshot.
11. You can now proceed with performing failback. See the following section, Performing failback.

Performing failback

After you repair or replace the failed original source core and protected machines, you need to move the data from your failed-over machines to restore the source machines.

To perform failback

1. Navigate to the AppAssure Core Console on the target core, and click the Replication tab or icon.
2. Under Incoming Replication, select the failover protected machine and expand the details.
3. On the Actions menu, click **Fail Back**.

   The Fail Back dialog box opens to describe the steps you need to follow before you click the Continue button to complete failback.
4. Click **Cancel**.
5. If the failed-over machine is running Microsoft SQL Server or Microsoft Exchange Server, stop those services.
6. Force a snapshot of the machine. For more information, see Forcing a snapshot.
7. Shut down the failed-over machine.

8 Create an archive of the failed-over protected machine and output it to disk or a network share location. For more information, see the section, Creating an archive.

9 After you create the archive, navigate to the AppAssure Core Console on the newly repaired source core, and then click the Tools tab.

10 Import the archive you just created in Step 8. For more information, see the section, Importing an archive.

11 Go back to the Core Console on the target core, and click the Replication tab.

12 Under Incoming Replication, select the failover protected machine and expand the details.

13 In the drop-down menu for the protected machine, click **Fail Back**.

14 In the Fail Back dialog box, click **Continue**.

15 Shut down the machine that contains the exported protected machine that was created during failover.

16 Perform a bare metal restore (BMR) for the source core and protected machine. For more information, see Performing a bare metal restore for Windows machines.

> **NOTE:** When you launch the restore as described in, Selecting a recovery point and initiating BMR, you will need to use the recovery points that were imported from the target core to the protected machine on the virtual machine.

17 Wait for the BMR reboot and for the AppAssure Agent service to restart, and then view and record the network connection details of the machine.

18 Navigate to the Core Console on the source core, navigate to the machine, and then modify the machine protection settings to add the new network connection details. For more information, see Configuring machine settings.

19 Navigate to the Core Console on the target core, and delete the protected machine from the Replication tab. For more information, see Removing replication.

20 In the Core Console of the source core, set up replication again between the source and target by clicking the Replication tab, and then adding the target core for replication. For more information, see the section, Replicating to a self-managed target core.

12

Managing events

This chapter describes how to manage events on the AppAssure Core that monitor the state of the AppAssure Core and its protected machines.It includes the following topics:

- Viewing tasks, alerts, and events
- Understanding email notifications
- Configuring notification groups
- Configuring repetition reduction
- Configuring event retention

The Core includes predefined sets of events, which can be used to notify administrators of critical issues on the Core or with backup jobs on protected machines.

- You define the event types that trigger alerts, and which approaches to use for these notifications, by configuring notification groups. For more information, see Configuring notification groups.
- If you want administrators to receive notifications using email, refer to the section Configuring repetition reduction. In addition to configuring a notification group, receiving alerts by email requires Configuring an email server and Configuring an email notification template.
- You can reduce the number of events of the same type and scope that appear in the Events tab, using the repetition reduction feature, which is enabled by default. You can disable this feature, or you can control the span of time for which events are combined into a single occurrence in the event log. For more information, see Configuring repetition reduction.
- You can control how long events and job history information is retained in the Events tab in the Core console. For more information, see Configuring event retention.

Viewing tasks, alerts, and events

The Events tab on the Core Console displays a log of all system events related to the AppAssure Core. When you view the Events tab for a selected machine, you see a log of all system events related to that specific machine.

The contents of the Events tab is divided into three sections: Tasks, Alerts, and Events, for you to view details about each event as appropriate.

You can define how you are notified of various events by configuring notification groups. For more information, see Configuring notification groups.

Complete the steps in the procedures below to view tasks, alerts, or all events, respectively.

Viewing tasks

A task is a job that the AppAssure Core must perform, such as transferring data in a regularly scheduled backup, or performing a restore from a recovery point.

As a task is running, it is listed in the Running tasks drop-down menu at the top of the Core Console.

You can also view all tasks for the AppAssure Core, or all tasks associated with a specific machine.

To view tasks

1 To view all tasks for the AppAssure Core, navigate to the AppAssure Core Home tab and then click the **Events** tab.

If you want to view tasks for a specific protected machine, then navigate to the Events tab for that machine.

2 To view only tasks, at the top left-hand side of the page, click **Tasks**.

The list of events is filtered to display only tasks for the Core or for the machine you selected.

3 Optionally, to filter the list of tasks by keyword, start date, end date, or any combination, do the following:

a To filter by keyword, enter the keyword in the Search keyword text box.

b To filter by start date and time, enter the starting date and time using one of the following options:

- In the From text box, type the date and time in format MM/DD/YYYY HH:MM AM/PM. For example, to search from the first day of January in 2014 at 8:00 AM, enter 1/1/2013 8:00 AM.
- To select the current data and time, click the Calendar widget in the From text box and then click **Now**.
- Click the Calendar widget, select the date and time using the calendar and slider controls, and then click **Done**.

c To further refine the list of tasks that appears, you can also define an end date and time in the same format.

The list of tasks is immediately filtered based on the criteria you selected.

4 Optionally, you can filter the tasks appearing in the list as follows:

- To see only active tasks, click the **Active Tasks** icon.
- To see only tasks that are in the queue to be performed, click the **Queued** icon.
- To see only tasks that are waiting to be performed, click the **Waiting Tasks** icon.
- To see only tasks that have been completed, click the **Completed Tasks** icon.
- To see only tasks that have failed, click the **Failed Tasks** icon.

5 To export the list of tasks, select a format from the list and then click **Export the report**.

You can export using the following formats:

Table 111. Export formats

Format	Description
PDF	Portable Document Format
XLS	Excel 1997 - 2003 Workbook
XLSX	Excel Workbook
RTF	Rich Text Format
CSV	Comma-separated values

6 Click the **Job Details** icon for any task to launch a new window with task details, which include:

- Start Time
- End Time
- Status
- Elapsed Time
- Progress

- Rate
- Total Work (size or percentage completed)
- Child Tasks (if applicable)
- Failure Reason (if applicable)

Viewing alerts

An alert is a notification related to a task or event. Alerts types include errors, warnings, or information.

You can view all alerts for the AppAssure Core, or all alerts associated with a specific machine.

To view alerts

1. To view all alerts for the AppAssure Core, navigate to the AppAssure Core Home tab and then click the **Events** tab.

   If you want to view alerts for a specific protected machine, then navigate to the Events tab for that machine.
2. To view only alerts, at the top left-hand side of the page, click **Alerts**.

   The list of events is filtered to display only alerts for the Core or for the machine you selected.
3. Optionally, if you want to remove all alerts, click **Dismiss All**.

Viewing all events

You can view all events for the AppAssure Core, or all events associated with a specific machine.

To view events

1. To view all events for the AppAssure Core, navigate to the AppAssure Core **Home** tab and then click the **Events** tab.

   If you want to view events for a specific protected machine, then navigate to the Events tab for that machine.
2. To view all events, at the top left-hand side of the page, click **Events**.

   All events display for the Core or for the machine you selected.

Understanding email notifications

The events which trigger alerts are defined in the notification group.

NOTE: Notification groups must be established regardless of whether you use email as a notification method. For more information, see Configuring notification groups.

If you choose email as one of the notification options, you must also configure an email SMTP server. The AppAssure Core uses the server you define to send alerts based on the parameters in the notification group.

Additionally, you must also define an email notification template. The Core uses this to define the email subject line for each alert, as well as the content in the email message body. The template has default settings; you can use the default as-is or you can test and make modifications to serve your needs.

This section includes the following topics:

- Configuring an email server
- Configuring an email notification template

Configuring an email server

Complete the steps in this procedure to configure an email server.

NOTE: You must also configure notification group settings, including enabling the Notify by email option, before email alert messages will be sent. For more information on specifying events to receive email alerts, see Configuring notification groups.

To configure an email server

1 Navigate to the AppAssure Core, click the Configuration tab, and then click **Events**.

2 In the Email Settings pane, click **SMTP server**.

The SMTP Server Settings dialog box appears.

3 Enter details for the email server as described in the following table.

Table 112. SMTP server settings

Text Box	Description
SMTP Server	Enter the name of the email server to be used by the email notification template. The naming convention includes the host name, domain, and suffix; for example, smtp.gmail.com.
From	Enter a return email address. It is used to specify the return email address for the email notification template; for example, noreply@localhost.com.
User name	Enter a user name for the email server.
Password	Enter the password associated with the user name required to access the email server.
Port	Enter a port number. It is used to identify the port for the email server; for example, the port 587 for Gmail. The default is *25*.
Timeout (seconds)	Enter an integer value to specify how long to try to connect to the email server. It is used to establish the time in seconds before a timeout occurs. The default is 60 *seconds*.
TLS	Select this option if the mail server uses a secure connection such as Transport Layer Security (TLS) or Secure Sockets Layer (SSL).

4 Click **Send Test Email** and then do the following:

a In the Send Test Email dialog box, enter a destination email address for the test message and then click **Send**.

b If the test message fails, exit the error dialog box and the Send Test Email dialog box, and revise your email server configuration settings. Then repeat Step 4.

c Once the test message is successful, click **OK** to confirm the successful operation.

d Check the email account to which you sent the test email message.

e Once you are satisfied with the results of your tests, return to the SMTP Server Settings dialog box, and click **Save** to close the dialog box and save your settings.

Configuring an email notification template

Complete the steps in this procedure to configure an email notification template. This template is used by your SMTP email server to send notifications regarding AppAssure events by email.

NOTE: You must also configure an email server and notification group settings, including enabling the **Notify by email** option, before email alert messages will be sent. For more information about configuring an email server for sending alerts, see Configuring an email server. For more information on specifying events to receive email alerts, see Configuring notification groups.

To configure an email notification template

1. Navigate to the AppAssure Core, click the Configuration tab, and then click **Events**.
2. In the Email Settings pane, click **change**.

   The Edit Email Notification Configuration dialog box appears.
3. Select **Enable Email Notifications**.
4. In the Email Subject text box, enter a subject for the email template.

   The Email Subject is used to define the subject of the email notification template, for example, <hostname> - <level> <name>.
5. In the Email text box, enter the information for the body of the template which describes the event, when it occurred, and the severity.
6. Click **Send Test Email** and then do the following:
   a. In the Send Test Email dialog box, enter a destination email address for the test message and then click **Send**.
   b. If the test message fails, exit the error dialog box and the Send Test Email dialog box, click **OK** to save the current email template settings, and modify your email server settings as described in the procedure Configuring an email server, ensuring you reenter the password for that email account. Save those settings and then return to this procedure.
   c. Once the test message is successful, click **OK** to confirm the successful operation.
   d. Check the email account to which you sent the test email message.

Once you are satisfied with the results of your tests, return to the Edit Email Notification Configuration dialog box, and click **OK** to close the dialog box and save your settings.

Configuring notification groups

Notification groups let you define sets of specific events for which users are alerted, and the manner in which that notification takes place. You can configure alerts to be sent by the following methods:

- By email
- In the Windows event log
- Using syslogd,
- Using Toast alerts
- Using alerts
- Using SNMP trap

NOTE: You must also configure Simple Mail Transfer Protocol (SMTP) server settings if you want to send alerts as email messages, as described in this procedure. For more information on setting email server configuration settings, see Configuring an email server.

Complete the steps in this procedure to configure notification groups for alerts.

To configure notification groups

1 Navigate to the AppAssure Core, click the Configuration tab, and then click **Events**.

2 Click **Add Group**.

The Add Notification Group dialog box displays.

The Add Notification Group dialog box contains a general description area and two tabs:

- Enable Alerts
- Notification Options

3 Enter the basic information for the notification group, as described in the following table.

Table 113. Notification group information

Text Box	Description
Name	Enter a name for the event notification group. This information is required.
Description	Enter a description that clarifies the purpose for the event notification group. This information is optional.

4 On the Enable Alerts tab, define the set of system events that you want to log, create reports for, and for which you want to be alerted, as follows:

- If you want to create alerts for all events, select **All Alerts**.
- If you want to create alerts specific to errors, warnings, informational messages, or a combination of these, then next to Select Types, click the appropriate option:
  - Error (red triangle icon)
  - Warning (yellow triangle icon)
  - Info (blue circle)
  - Restore default (curved arrow)
- If you want to create alerts for specific events, then do the following:
  a Click the right angle bracket > symbol next to All Alerts to expand to view to show Groups. Then click the right angle bracket > symbol next to Groups to display groups of related events for which you can set alerts. The event group categories include:
    - Clouds
    - Server Logs
    - Exchange
    - Scheduled Archives
    - Auto Update
    - Dedupe Cache
    - Recovery Point Check
    - Remote Mount
    - Boot CD
    - Security
    - Database Retention
    - Local Mount
    - Metadata
    - Clusters
    - Notification

- PowerShell Scripting
- Push Install
- Attachability
- Jobs
- Licensing
- Log Truncation
- Archive
- Core Service
- Export
- Protection
- Replication
- Repository
- Rollback (Restore)
- Rollup

b To view the individual events in any group, click the right angle bracket > symbol next to the relevant group, and then select those specific events for which you want to log, report, and set alerts.

c To define alerts for all events within any group, select the checkbox next to that group.

5 Click the Notification Options tab.

6 On the Notification Options tab, specify how to handle the notification process.

The following table describes the notification options.

Table 114. Notification options

Text Box	Description
Notify by email	Designate the recipients of the email notification. You can choose to specify separate multiple email addresses as well as blind and carbon copies. You can choose: • To: • CC: • BCC:
Notify by Windows Event Log	Select this option if you want alerts to be reported through the Windows Event Log.
Notify by sys logd	Select this option if you want alerts to be reported through syslogd. Specify the details for the syslogd in the following text boxes: • Host: • Port:
Notify by Toast alerts	Select this option if you want the alert to appear as a pop-up in the lower-right corner of your screen.

Table 114. Notification options

Text Box	Description
Notify by Alerts	
Notify by SNMP trap	The AppAssure Core serves as an SNMP agent, sending traps (notifications about specific events) to an SNMP manager. The result is the reporting of Core information such as alerts, repository status, and protected machines. Select this option if you want to notify Core events by SNMP trap. You must also specify a trap number. For example, the default trap number used by the AppAssure Core is 162.

7 Click **OK**.

You will see a message indicating that the notification group name you defined cannot be changed after creating the group. Other properties within the notification group can be changed at any time.

- If you are satisfied with the group name, confirm this message and save your work.
- If you want to change the group name, click **No** to return to the Create Notification Group window, update the group name and any other notification group settings, and save your work.

Configuring repetition reduction

The ability for administrators to receive alerts upon certain events is critical. Nonetheless, in certain circumstances, receiving repeated notification of events that you are aware of can also be frustrating or inconvenient. Even if an alert is generated due to an environmental failure that you wish to know about immediately, it is possible for the same error condition to generate hundreds or thousands of events in the event log.To reduce repetition in the event log, and reduce the inconvenience of receiving repeated e-mail notifications for the same event in the Core Console, AppAssure includes a repetition reduction setting, which is enabled by default and set at 5 minutes. This setting can be set as low as 1 minute and as high as 60 minutes. It can also be disabled entirely.

When repetition reduction is disabled, then every time an event of the same type and scope occurs, it is logged in the database. Regardless of how much time passed since that event previously occurred, each new occurrence is shown in the Alerts portion of the Events tab.

When repetition reduction is configured (for example, with the default time of 5 minutes), then the first time that specific event occurs, it is logged in the database and shown in the alerts log. If subsequently an event of the same type and scope is again logged within the threshold of time established, then the count for the event in the database increases by 1 for each repeat occurrence within that threshold. The log shown in the Alerts portion of the Events tab, however, shows the event only once, with the date and time of the most recent occurrence. is not updated with the same event until the threshold of time from the first occurrence expires.

NOTE: Repetition reduction settings apply only to event logging, and do not have any effect on email notifications.

Complete the steps in this procedure to configure repetition reduction for events.

To configure repetition reduction

1 On the AppAssure Core home page, click the Configuration drop-down menu, and then click **Events**.

2 From the Repetition Reduction area, click **Change**.

The Repetition Reduction dialog box displays.

3 Select **Enable Repetition Reduction**.

4 In the **Store Events for** text box, use the up and down arrows to enter the number of minutes to store the events for repetition reduction.

5 Click **OK**.

Configuring event retention

Complete the steps in this procedure to configure retention for events.

To configure event retention

1. On the AppAssure Core home page, click the Configuration drop-down menu, and then click **Settings**.
2. Under Database Connection Settings, click **change**.

   The Database Connection Settings dialog box displays.
3. In the **Retain event and job history for** text box, enter the number of days that you want to retain information about events; for example, *30 days* (default).
4. Click **Save**.

13

Generating and viewing reports

This chapter provides an overview of reporting available in Dell AppAssure. It consists of the following topics:

- Understanding Compliance reports
- Understanding Failure reports
- Understanding the Summary report
- Generating a report for a core or agent
- Understanding Central Management Console core reports
- Generating a report from the Central Management Console

AppAssure lets you generate and view compliance, error, and summary information for multiple core and agent machines.

You can choose to view reports online, print reports, or export and save them in one of several supported formats. The formats from which you can choose are:

- PDF
- XLS
- XLSX
- RTF
- MHT
- HTML
- TXT
- CSV
- Image

With this release of AppAssure, reports now include units of measure which make it easier to determine if a column is represented in GB, TB, or in seconds.

Using the reports toolbar

The toolbar available for all reports lets you print and save in two different ways. The following table describes the print and save options.

Table 115. Reports toolbar icons

Icon	Description
	Print the report
	Print the current page
	Export a report and save it to the disk

For information about generating a report, see Generating a report for a core or agent. For information about the generating a report for multiple cores in the Central Management Console, see Generating a report from the Central Management Console. For information about generating cluster reports, see Viewing a cluster or node report.

Understanding Compliance reports

Compliance reports are available for the AppAssure Core and AppAssure Agent. They provide you with a way to view the status of jobs performed by a selected core or agent. Failed jobs appear in red text. Information in the Core Compliance Report that is not associated with an agent appears blank.

Details about the jobs are presented in a column view that includes the following categories:

- Core
- Protected Agent
- Type
- Summary
- Status
- Error
- Start Time
- End Time
- Time
- Total Work

For information about how to generate a report, see Generating a report for a core or agent.

Understanding Failure reports

Failure reports are subsets of the compliance reports and are available for AppAssure Cores and AppAssure Agents. Failure reports include only the failed jobs listed in Compliance Reports and compile them into a single report that can be printed and exported.

Details about the errors are presented in a column view with the following categories:

- Core
- Agent
- Type
- Summary
- Error
- Start Time
- End Time
- Elapsed Time
- Total Work

For information about how to generate a report, see Generating a report for a core or agent.

Understanding the Summary report

The Summary report includes information about the repositories on the selected AppAssure Core and about the agents protected by that core. The information appears as two summaries within one report. For information on how to generate a summary report, see Generating a report for a core or agent.

Repositories summary

The Repositories portion of the Summary Report includes data for the repositories located on the selected AppAssure Core. Details about the repositories are presented in a column view with the following categories:

- Name
- Data Path
- Metadata Path
- Allocated Space
- Used Space
- Free Space
- Compression/Dedupe Ratio

NOTE: Allocated, used, and free space are represented in units of measurement. The measurements are GB, TB, or Seconds.

Agents summary

The Agents portion of the Summary Report includes data for all machines protected by the selected AppAssure Core.

Details about each protected machine is presented in a column view with the following categories:

- Name
- Protected Volumes
- Total protected space
- Current protected space
- Change rate per day (Average | Median)
- Jobs Statistic (Passed | Failed | Canceled)

Generating a report for a core or agent

Complete the steps in the following procedure to generate a report for an AppAssure Core or for a protected machine.

To generate a report for a core or protected machine

1 Navigate to the AppAssure Core Console and select the Core or the protected machine for which you want to run the report.

2 Click the Tools tab.

3 From the Tools tab, expand **Reports** in the left navigation area.

4 In the left navigation area, select the report you want to run. The reports available depend on the selection you made in Step 1 and are described in the following table.

Table 116. Available reports

Machine	Available Reports
Core	Compliance Report
	Summary Report
	Failure Report
Agent	Compliance Report
	Failure Report

5 In the Start Time drop-down calendar, select a start date, and then enter a start time for the report.

> **NOTE:** No data is available before the time the Core or the AppAssure Agent software on the protected machine was deployed.

6 In the End Time drop-down calendar, select an end date, and then enter an end time for the report.

7 For a Summary Report, select the **All Time** check box if you want the Start Time and the End Time to span the lifetime of the Core.

8 For a Compliance Report or a Failure Report, use the Target Cores drop-down list to select the Core for which you want to view data.

9 Click **Generate Report**.

After the report generates, you can use the toolbar to print or export the report. For more information about the toolbar, see Using the reports toolbar.

Understanding Central Management Console core reports

AppAssure lets you generate and view compliance, error, and summary information for multiple AppAssure Cores. Details about the Cores are presented in column views with the same categories described in the sections Understanding Compliance reports, Understanding Failure reports, and Understanding the Summary report.

For information on how to generate a report for multiple cores, see Generating a report from the Central Management Console.

Generating a report from the Central Management Console

Complete the following procedure to generate a report for multiple AppAssure Cores from the Central Management Console.

To generate a report from the Central Management Console

1. From the Central Management Console Welcome screen, click the drop-down menu in the upper-right corner.
2. From the drop-down menu, click **Reports** and then select one of the following options:
   - Compliance Report
   - Summary Report
   - Failure Report
3. From the left navigation area, select the AppAssure Core or Cores for which you want to run the report.
4. In the Start Time drop-down calendar, select a start date, and then enter a start time for the report.

   NOTE: No data is available before the time the Cores were deployed.

5. In the End Time drop-down calendar, select an end date, and then enter an end time for the report.
6. Click **Generate Report**.

   After the report generates, you can use the toolbar to print or export the report. For more information about the toolbar, see Using the reports toolbar.

14

Restoring data

This chapter describes how to restore backed up data. It includes the following sections:

The AppAssure Core can instantly restore data or recover machines to physical or virtual machines from the recovery points. The recovery points contain agent volume snapshots captured at the block level. These snapshots are application aware, meaning that all open transactions and rolling transaction logs are completed and caches are flushed to disk before creating the snapshot. Using application-aware snapshots in tandem with Verified Recovery enables the Core to perform several types of recoveries, including:

- Recovery of files and folders
- Recovery of data volumes, using Live Recovery
- Recovery of data volumes for Microsoft Exchange Server and Microsoft SQL Server, using Live Recovery
- Bare metal restore, using Universal Recovery
- Bare metal restore to dissimilar hardware, using Universal Recovery
- Ad-hoc and continual export to virtual machines

Restoring data from recovery points

AppAssure protects your data on Windows and Linux machines. Backups of protected agent machines are saved to the AppAssure Core as recovery points, Using these recovery points, you can restore your data using one of three methods.

From the AppAssure Core Console, you can restore entire volumes from a recovery point of a non-system volume, replacing the volumes on the destination machine. You can do this for Windows or Linux machines. For more information, see Restoring volumes from a recovery point.

You can also restore entire volumes on Linux machines from recovery points using the command line from the Linux agent. For more information on using the command line aamount utility, see Restoring volumes for a Linux machine using the command line.

You cannot restore a volume that contains the operating system directly from a recovery point, because the machine to which you are restoring is using the operating system and drivers that are included in the restore process. If you want to restore from a recovery point to a system volume (for example, the C drive of the agent machine), you must perform a Bare Metal Restore (BMR). This involves creating a bootable image from the recovery point, which includes operating system and configuration files as well as data, and starting the target machine from that bootable image to complete the restore. The boot image differs if the machine you want to restore uses a Windows operating system or a Linux operating system.

If you want to restore from a recovery point to a system volume on a Windows machine, see Performing a bare metal restore for Windows machines.

If you want to restore from a recovery point to a system volume on a Linux machine, see Performing a bare metal restore for Windows machines.

Finally, in contrast to restoring entire volumes, you can mount a recovery point from a Windows machine, and browse through individual folders and files to recover only a specific set of files. For more information, see Restoring a directory or file using Windows Explorer. If you need to perform this while preserving original file permissions (for example, when restoring a user's folder on a file server), see Restoring a directory or file and preserving permissions using Windows Explorer.

The topics in this section describe information about restoring data on physical machines. For more information on exporting protected data from Windows Machines to virtual machines, see Exporting protected data from Windows machines to virtual machines.

NOTE: When recovering data on Windows machines, if the volume that you are restoring has Windows data deduplication enabled, you will need to make sure that deduplication is also enabled on the Core server.

AppAssure supports Windows 8, Windows 8.1, Windows Server 2012, and Windows Server 2012 R2 for normal transfers (both base and incremental) as well as with restoring data, bare metal restore, and virtual exports.

For more information on the types of volumes supported and not supported for backup and recovery, see Dynamic and basic volumes support limitations.

Restoring volumes from a recovery point

You can restore the volumes on a protected machine from the recovery points stored in the AppAssure Core. In AppAssure 5.4 and later, this process uses the Restore Machine Wizard.

NOTE: In previous releases, this process was referred to as performing a rollback.

AppAssure supports the protection and recovery of machines configured with EISA partitions. Support is also extended to Window 8, Windows 8.1, Windows Server 2012, and Windows Server 2012 R2 machines that use Windows Recovery Environment (Windows RE).

You can begin a restore from any location on the AppAssure Core Console by clicking the **Restore** icon in the AppAssure button bar. When you start a restore in this manner, you must specify which of the machines protected on the Core you want to restore, and then drill down to the volume you want to restore.

Or, you can drill down in the Core Console UI to a specific machine, expand the recovery points for volumes on that machine, and from the appropriate recovery point, select **Restore**. If you begin a restore in this manner, then follow this procedure starting with Step 5.

If you want to restore a recovery point on a Linux machine, you should first dismount the disk on which you will be restoring data.

If you want to restore from a recovery point to a system volume, or restore from a recovery point using a boot CD, you must perform a Bare Metal Restore (BMR). For information about BMR, see Understanding bare metal restore for Windows machines, and for prerequisite information for Windows or Linux operating systems, see Prerequisites for performing a bare metal restore for a Windows machine and Prerequisites for performing a bare metal restore for a Linux machine, respectively. You can access BMR functions from the Core Console as described in the roadmap for each operating system. You can also perform a BMR from the Restore Machines Wizard. This procedure will direct you at the appropriate point in the wizard to the procedure Managing a Windows boot image and launching a BMR from the Restore Machine Wizard.

Follow the procedure below to restore volumes from a recovery point.

To restore volumes from a recovery point

1 To restore a volume on a protected machine from the Restore icon, navigate to the Core Console and click **Restore** from the AppAssure button bar.

The Restore Machine Wizard appears.

2 From the Protected Machines page, select the protected machine for which you want to restore data, and then click **Next**.

> **NOTE:** The protected machine must have the Agent software installed and must have recovery points from which you will perform the restore operation.

The Recovery Points page appears.

3 From the list of recovery points, search for the snapshot you want to restore to the agent machine.

- If necessary, use the navigation buttons at the bottom of the page to display additional recovery points.
- Optionally, If you want to limit the amount of recovery points showing in the Recovery Points page of the wizard, you can filter by volumes (if defined) or by creation date of the recovery point.

4 Click any recovery point to select it, and then click **Next**.

The Destination page appears.

5 On the Destination page, choose the machine to which you want to restore data as follows:

- If you want to restore data from the selected recovery point to the same agent machine (for example, Machine1), and if the volumes you want to restore do not include the system volume, then select **Recover to a protected machine (only non-system volumes)**, verify that the destination machine (Machine1) is selected, and then click **Next**.

  The Volume Mapping page appears. Proceed to Step 9.

- If you want to restore data from the selected recovery point to a different protected machine (for example, to replace the contents of Machine2 with data from Machine1), then select **Recover to a protected machine (only non-system volumes)**, select the destination machine (for example, Machine2) from the list, and then click **Next**.

  The Volume Mapping page appears. Proceed to Step 9.

- If you want to restore from the selected recovery point to the same machine or a different machine using a boot CD, this is considered a bare metal restore (BMR). For information about BMR, see Understanding bare metal restore for Windows machines.

  > **NOTE:** Performing a BMR has specific requirements, based on the operating system of the agent machine you want to restore. To understand these prerequisites, see Prerequisites for performing a bare metal restore for a Windows machine and Prerequisites for performing a bare metal restore for a Linux machine, respectively.

  If the volumes you want to restore do not include the system volume, then select **Recover to any target machine using a boot CD**. This option will prompt you to create a boot CD.

  - To continue and create the boot CD with information from the selected recovery point using the Restore Machine Wizard, click **Next** and proceed to Managing a Windows boot image and launching a BMR from the Restore Machine Wizard.
  - If you have already created the boot CD and the target machine has been started using the boot CD, then proceed to Step 8 of the topic Managing a Windows boot image and launching a BMR from the Restore Machine Wizard.

- If you want to restore from a recovery point to a system volume (for example, the C drive of the agent machine named Machine1), this is also considered a BMR. Select **Recover to any target machine using a boot CD**. This option will prompt you to create a boot CD.

  - To continue and create the boot CD with information from the selected recovery point using the Restore Machine Wizard, click Next and proceed to Managing a Windows boot image and launching a BMR from the Restore Machine Wizard.
  - If you have already created the boot CD, then proceed to Step 6.

6 Start the machine you want to restore to using the boot CD. For more information, for BMR on a Windows machine, see Loading the boot CD and starting the target machine and for BMR on a Linux machine, see Loading the Live DVD and starting the target machine.

7 Back on the Core server, in the Destination page of the Restore Machine Wizard, select **I already have a boot CD running on the target machine** and enter the information about the machine to which you want to connect as described in the following table.

Table 117. Machine information

Text Box	Description
IP Address	The IP address of the machine to which you want to restore. This is identical to the IP address displayed in the URC.
Authentication Key	The specific password to connect to the selected server. This is identical to the Authentication Key displayed in the URC.

8 Click **Next**.

If the connection information you entered matches the URC, and if the Core and the target server can identify each other properly on the network, then the volumes for the selected recovery point are loaded, and the Disk Mapping page appears.

To complete your BMR from the Restore Machine Wizard, proceed to Step 8 of the topic Managing a Windows boot image and launching a BMR from the Restore Machine Wizard.

NOTE: While AppAssure supports FAT32 and ReFS partitions, at present, only full restore and BMR are supported as a driver limitation exists with ReFS, so restore is implemented in user mode, VM export, and so on. If a Core is protecting at least one agent volume that contains the ReFS file system, it should be installed on Windows 8/2012 which provides native support of ReFS, otherwise functionality will be limited and operations that involve such things as mounting a volume image will not work. The AppAssure Core Console will present applicable error messages in these occurrences.

Bare metal restore of Storage Spaces disks configuration (a feature of Windows 8.1) is also not supported in this release. For details, see the *Dell AppAssure Installation and Upgrade Guide*.

9 On the Volume Mapping page, for each volume in the recovery point that you want to restore, select the appropriate destination volume. If you do not want to restore a volume, in the Destination Volumes column, select **Do not restore**.

10 Select **Show advanced options** and then do the following:

- For restoring to Windows machines, if you want to use Live Recovery, select **Live Recovery**.

  Using the Live Recovery instant recovery technology in AppAssure, you can instantly recover or restore data to your physical machines or to virtual machines from stored recovery points of Windows machines, which includes Microsoft Windows Storage Spaces. Live Recovery is not available for Linux machines.

- If you want to force dismount, select **Force Dismount**.

  If you do not force a dismount before restoring data, the restore may fail with a volume in use error.

11 If the volumes you want to restore contain SQL or Microsoft Exchange databases, then on the Dismount Databases page, you are prompted to dismount them. Optionally, if you want to remount these databases after the restore is complete, select **Automatically remount all databases after the recovery point is restored**. Then click **Finish**.

12 Click **OK** to confirm the status message that the restore process has started.

13 Optionally, to monitor the progress of your restore action, on the Core Console, click **Events**. For more information, see Viewing tasks, alerts, and events.

Restoring a directory or file using Windows Explorer

You can use Windows Explorer to copy and paste directories and files from a mounted recovery point to any Windows machine. This can be helpful when you want to distribute only a portion of a recovery point to your users.

When you copy directories and files, the access permissions of the user who is performing the copy operation are used and applied to the pasted directories and files. If you want to restore directories and files to your users while preserving original file permissions (for example, when restoring a user's folder on a file server), see Restoring a directory or file and preserving permissions using Windows Explorer.

To restore a directory or file using Windows Explorer

1. Mount the recovery point that contains the data you want to restore. For details, see Mounting a recovery point.
2. In Windows Explorer, navigate to the mounted recovery point and select the directories and files that you want to restore. Right-click and select **Copy**.
3. In Windows Explorer, navigate to the machine location to where you want to restore the data. Right-click and select **Paste**.

Restoring a directory or file and preserving permissions using Windows Explorer

You can use Windows Explorer to copy and paste directories and files from a mounted recovery point to any Windows machine while preserving file access permissions.

For example, if you need to restore a folder accessed only by specific users on a file server, you can use the **Copy** and **Paste with Permissions** commands to ensure that the restored files retain the permissions that restrict access. In this way, you can avoid having to manually apply permissions to the restored directories and files.

NOTE: The Paste with Permissions command is installed with AppAssure Core and Agent software. It is not available in the Local Mount Utility.

To restore a directory or file and preserve permissions using Windows Explorer

1. Mount the recovery point that contains the data you want to restore. For details, see Mounting a recovery point.
2. In Windows Explorer, navigate to the mounted recovery point and select the directories and files that you want to restore. Right-click and select **Copy**.
3. In Windows Explorer, navigate to the machine location to where you want to restore the data. Right-click and select **Paste with Permissions**.

   NOTE: In this step, if the **Paste with Permissions** command is disabled on the right-click menu, then Windows Explorer is not aware of the files that you want to copy. Repeat Step 2 to enable the **Paste with Permissions** command on the right-click menu.

15

Understanding bare metal restore for Windows machines

This chapter describes how to restore a protected Windows machine from bare-metal similar or dissimilar hardware. It includes the following topics:

- Performing a bare metal restore for Windows machines
- Managing a Windows boot image
- Launching a bare metal restore for Windows
- Verifying a bare metal restore

Servers, when operating as expected, perform the tasks they are configured to do. It is only when they fail that things change. When a catastrophic event occurs, rendering a server inoperable, immediate steps are needed to restore the full functionality of that machine.

AppAssure provides the ability to perform a bare metal restore (BMR) for your Windows or Linux machines. BMR is a process that restores the full software configuration for a specific system. It uses the term “bare metal” because the restore operation recovers not only the data from the server, but also reformats the hard drive and reinstalls the operating system and all software applications. To perform a BMR, you specify a recovery point from a protected machine, and roll back (perform a restore) to the designated physical or virtual machine. If you are performing a restore to a system volume, this is considered a BMR. If you are performing a restore and require a boot CD, this is also considered a BMR. Other circumstances in which you may choose to perform a bare metal restore include hardware upgrade or server replacement In both of these cases, you perform a restore from a recovery point to the upgraded or replaced hardware.

AppAssure supports Windows 8, 8.1 and Windows Server 2012, 2012 R2 operating systems that are booted from FAT32 EFI partitions are available for protection or recovery, as well as Resilient File System (ReFS) volumes.

NOTE: Bare metal restore of Storage Spaces disks configuration (a feature of Windows 8.1) is also not supported in this release.

At present, only full restore and BMR are supported as a driver limitation exists with ReFS, so restore is implemented in user mode, VM export, and so on. If a Core is protecting at least one agent volume that contains the ReFS file system, it should be installed on a Window 8, Windows 8.1, Windows Server 2012, or Windows Server 2012 R2 machine, since these operating systems provides native support of ReFS. Otherwise, functionality will be limited and operations that involve such things as mounting a volume image will not work. The AppAssure Core Console will present applicable error messages in these occurrences.

Only supported Linux operating systems are available for protection or recovery. This includes Ubuntu, Red Hat Enterprise Linux, CentOS, and SUSE Linux Enterprise Server (SLES). For details, see the *Dell AppAssure Installation and Upgrade Guide*.

Performing a BMR is possible for physical or virtual machines. As an added benefit, AppAssure allows you to perform a BMR whether the hardware is *similar* or *dissimilar*. Performing a BMR on AppAssure separates the operating system from a specific platform, providing portability.

Examples of performing a BMR for similar hardware include replacing the hard drive of the existing system, or swapping out the failed server with an identical machine.

Examples of performing a BMR for dissimilar hardware include restoring a failed system with a server produced by a different manufacturer or with a different configuration. This process encompasses creating a boot CD image, burning the image to disk, starting up the target server from the boot image, connecting to the recovery

console instance, mapping volumes, initiating the recovery, and then monitoring the process. Once the bare metal restore is complete, you can continue with the task of loading the operating system and the software applications on the restored server, followed by establishing unique settings required for your configuration.

Bare metal restore is used not only in disaster recovery scenarios, but also to migrate data when upgrading or replacing servers.

While BMR is supported for virtual machines, it is also worth noting that it is easier to perform a Virtual Export for a VM than it is to perform a BMR on a physical machine. For more information on performing a VM export for virtual machines, see the appropriate procedure for the supported VM.

- For more information on performing a VM export using ESXi, see Exporting data to an ESXi virtual machine.
- For more information on performing a VM export using VMware Workstation, see Exporting data to a VMware Workstation virtual machine.
- For more information on performing a VM export using Hyper-V, see Exporting data to a Hyper-V virtual machine.
- For more information on performing a VM export using VirtualBox, see Exporting data to a VirtualBox virtual machine.
- For more information on performing a VM export of a protected Linux machine, see Exporting data to a Linux-based VirtualBox virtual machine.

To perform a BMR on a Windows machine, refer to the roadmap specific to Windows, including the prerequisites. For more information, see Performing a bare metal restore for Windows machines.

You can also perform a BMR from the Restore Machine Wizard. To do this, start with the procedure Restoring volumes from a recovery point and, when directed in that procedure, proceed to Managing a Windows boot image and launching a BMR from the Restore Machine Wizard.

To perform a BMR on a Linux machine, refer to the roadmap specific to Linux, including prerequisites. In addition to performing a BMR using the command line aamount utility, you can now perform a BMR from within the Core Console UI. The roadmap takes both approaches into account.

Performing a bare metal restore for Windows machines

To perform a bare metal restore for Windows machines, perform the following tasks.

- **Manage a Windows boot image**. This boot CD ISO image will be used to start up the destination drive, from which you can access the Universal Recovery Console to communicate with backups on the Core. See Managing a Windows boot image.
    - If you require physical media to start up the destination machine, you will need to **transfer the boot CD ISO image to media**. See Transferring the boot CD ISO image to media.
    - In all cases, you will need to **load the boot image into the destination server and start the server** from the boot image. See Loading the boot CD and starting the target machine.
- **Launch a Bare Metal Restore for Windows**. Once the destination machine is started from the boot CD, you can launch the BMR. See Launching a bare metal restore for Windows.
    - You will need to **initiate a restore from a recovery point on the Core**. See Selecting a recovery point and initiating BMR.
    - You will need to **map the volumes**. See Mapping volumes for a bare metal restore.
    - If restoring to dissimilar hardware, and the necessary storage and network drivers are not present on the boot CD, you may need to load the drivers from a portable media device. For more information, see Loading drivers using the Universal Recovery Console.

    - If restoring to dissimilar hardware, and all necessary drivers are present on the boot CD, you will need to **inject drivers for hardware devices** that were not in the previous configuration but are included in the system replacing the server. For more information, see Injecting drivers to your target server.

- **Performing a BMR from the Restore Machine Wizard**. Optionally, the processes for managing a Windows boot image and for launching the BMR, including all sub-tasks, can be performed from the Restore Machine Wizard. For information on launching the wizard, see steps 1 through 5 of Restoring volumes from a recovery point, and then refer to Managing a Windows boot image and launching a BMR from the Restore Machine Wizard.
- **Verifying a Bare Metal Restore**. After starting the bare metal restore, you can verify and monitor your progress. See Verifying a bare metal restore.
    - You can **monitor the progress of your restore**. See Viewing the recovery progress.
    - Once completed, you can **start the restored server**. See Starting a restored target server
    - **Troubleshoot the BMR process**. See Troubleshooting connections to the Universal Recovery Console and Repairing startup problems.

Prerequisites for performing a bare metal restore for a Windows machine

Before you can begin the process of performing a bare metal restore for a Windows machine, you must ensure that the following conditions and criteria exist:

- **Backups of the machine you want to restore.** You must have a functioning AppAssure Core containing recovery points of the protected server you want to restore
- **Hardware to restore (new or old, similar or dissimilar).** The target machine must meet the installation requirements for an agent; for details, see the *Dell AppAssure Installation and Upgrade Guide*.
- **Image media and software.** You must have a blank CD or DVD and disk burning software, or software to create an ISO image. If managing machines remotely using virtual network computing software such as UltraVNC, then you must have VNC Viewer.
- **Compatible storage drivers and network adapter drivers.** If restoring to dissimilar hardware, then you must have Windows 7 PE (32-bit) compatible storage drivers and network adapter drivers for the target machine, including RAID, AHCI, and chipset drivers for the target operating system, as appropriate.
- **Storage space and partitions, as appropriate.** Ensure that there is enough space on the hard drive to create destination partitions on the target machine to contain the source volumes. Any destination partition should be at least as large as the original source partition.
- **Compatible partitions.** Windows 8, Windows 8.1, Windows Server 2012, and Windows Server 2012 R2 operating systems that are booted from FAT32 EFI partitions are available for protection or recovery, as well as are Resilient File System (ReFS) volumes. UEFI partitions are treated as simple FAT32 volumes. Incremental transfers are fully supported and protected. AppAssure provides support of UEFI systems for BMR including automatic partitioning GPT disks.

Managing a Windows boot image

A bare metal restore for Windows requires a boot image referred to as the boot CD, which you create by defining parameters in the AppAssure Core Console. This image is tailored to your specific needs. You will use the image to start the destination Windows machine. Based on the specifics of your environment you may need to transfer this image to physical media such as a CD or DVD. You must then virtually or physically load the boot image, and start the Windows server from the boot image.

This process is a step in Performing a bare metal restore for Windows machines.

To manage a Windows boot image, you can perform the following tasks:

- Creating a boot CD ISO image for Windows
- Defining boot CD ISO image parameters
- Transferring the boot CD ISO image to media
- Loading the boot CD and starting the target machine

NOTE: This process describes how to manage a boot CD image from the Create Boot CD dialog box. You can also perform these steps from the Restore Machine Wizard, starting from the Boot CD page of the wizard. You access this when you specify **Recover to any target machine using a boot CD** from the Destination page of the wizard.

For step-by-step instructions for managing a Windows boot image from the Restore Machine Wizard as part of a bare metal restore, see Managing a Windows boot image and launching a BMR from the Restore Machine Wizard.

Creating a boot CD ISO image for Windows

The first step when performing a bare metal restore (BMR) for a Windows machine is to create the boot CD file in the AppAssure Core Console. This is a bootable ISO image which contains the AppAssure Universal Recovery Console (URC) interface, an environment that is used to restore the system drive or the entire server directly from the AppAssure Core.

The boot CD ISO image that you create is tailored to the machine being restored; therefore, it must contain the correct network and mass storage drivers. If you anticipate that you will be restoring to different hardware from the machine on which the recovery point originated, then you must include storage controller and other drivers in the boot CD. For information about injecting those drivers in the boot CD, see Injecting drivers in a boot CD.

NOTE: The International Organization for Standardization (ISO) is an international body of representatives from various national organizations that sets file system standards. The ISO 9660 is a file system standard that is used for optical disk media for the exchange of data and supports various operating systems, for example, Windows. An ISO image is the archive file or disk image, which contains data for every sector of the disk as well as the disk file system.

This task is a step in Performing a bare metal restore for Windows machines. It is part of the process for Managing a Windows boot image.

To create a boot CD ISO image

1. From the AppAssure Core Console where the server you need to restore is protected, select the Core and then click the Tools tab.
2. Click **Boot CDs**.
3. Select **Actions**, and then click **Create Boot CD**.

   The Create Boot CD dialog box displays. Use the following procedures to complete the dialog box.

Defining boot CD ISO image parameters

Once you open the Create Boot CD dialog box, there are several parameters that may be required. Based on the specifics of your situation, perform the following tasks as required to define properties for a boot CD ISO image to use for a bare metal restore.

This task is a step in Performing a bare metal restore for Windows machines. It is part of the process for Creating a boot CD ISO image for Windows.

Naming the boot CD file and setting the path

Complete the following step to name the boot CD file and set the path where the ISO image is stored.

This task is a step in the process of Defining boot CD ISO image parameters. It is part of the process for Creating a boot CD ISO image for Windows.

To name the boot CD file and set the path

- In the Create Boot CD dialog box, in Output Options, in the Output path text box, enter the path where you want to store the boot CD ISO image on the Core server.

  If the shared drive on which you want to store the image is low on disk space, you can set the path as needed; for example, D:\filename.iso.

  > **NOTE:** The file extension must be .iso. When specifying the path, use only alphanumeric characters, the hyphen, the backslash (only as a path delimiter), and the period (only to separate host names and domains). The letters a to z are case-insensitive. Do not use spaces. No other symbols or punctuation characters are permitted.

Creating connections

Complete the following steps to create the connections.

This task is a step in the process of Defining boot CD ISO image parameters. It is part of the process for Creating a boot CD ISO image for Windows.

To create connections

1. In Connection Options, do one of the following:
   - To obtain the IP address dynamically using Dynamic Host Configuration Protocol (DHCP), select **Obtain IP address automatically**.
   - Optionally, to specify a static IP address for the recovery console, select **Use the following IP address** and enter the IP address, subnet mask, default gateway, and DNS server in the appropriate fields. You must specify all four of these fields.
2. If required, in the UltraVNC Options, select **Add UltraVNC** and then enter UltraVNC Password and UltraVNC Port.

   The UltraVNC settings enable you to manage the recovery console remotely while it is in use.

   > **NOTE:** This step is optional. If you require remote access to the recovery console, you must configure and use the UltraVNC. You cannot log on using Microsoft Terminal Services while using the boot CD.

Specifying a recovery environment

The boot CD image must be created so that the boot CD (physical or virtual) is mountable on the hardware you are restoring to. You must specify the architecture best suited for that machine.

This task is a step in the process of Defining boot CD ISO image parameters. It is part of the process for Creating a boot CD ISO image for Windows.

Complete the following step to specify a recovery environment

- Under Recovery Environment, select from the environment options as described in the following table.

Table 118. Recovery Environment options

Option	Description
64-bit Windows OS	To restore on any Windows machine with a 64-bit architecture, including machines with a UEFI BIOS
32-bit Windows OS	To restore on any machine with a 32-bit (x86) architecture

Injecting drivers in a boot CD

The boot CD image requires storage drivers to recognize the drives of the server, and network adapter drivers in order to communicate with the AppAssure Core over the network.

A generic set of Windows 7 PE 32-bit storage controller and network adapter drivers are included automatically when you generate a boot CD for Windows. This will satisfy the requirements of newer Dell systems. Systems from other manufacturers or older Dell systems may require you to inject storage controller or network adapter drivers when creating the boot CD. If you discover the boot CD you created does not contain the drivers necessary to complete the restore, you can also load drivers on to the target machine using the URC. Fore more information, see Loading drivers using the Universal Recovery Console.

When creating the boot CD, driver injection is used to facilitate the operability between the recovery console, network adapter, and storage on the target server.

Data restored from the recovery point includes drivers for the hardware previously in place. If performing a bare metal restore to dissimilar hardware, then you must also inject storage controller drivers into the operating system being restored using the URC after the data has been restored to the drive, This allows the restored operating system to boot using the new set of hardware. After the OS is booted after the restore, you can then download and install any additional drivers needed by the OS to interact with its new hardware.

For more information, see Injecting drivers to your target server.

This task is a step in the process of Defining boot CD ISO image parameters. It is part of the process for Creating a boot CD ISO image for Windows.

Complete the following steps to inject storage controller and network adapter drivers in a boot CD.

To inject drivers in a boot CD

1. Download the drivers from the manufacturer's Web site for the server and unpack them.
2. Compress each driver into a .zip file using an appropriate compression utility (for example, WinZip).
3. In the Create Boot CD dialog box, in the Drivers pane, click **Add a Driver**.
4. Navigate through the filing system to locate the compressed driver file, select the file, and then click **Open**.

   The injected drivers appear highlighted in the Drivers pane.
5. Repeat Step 3 and Step 4, as appropriate, until all drivers have been injected.

Creating the boot CD ISO image

This task is a step in the process of Defining boot CD ISO image parameters. It is part of the process for Creating a boot CD ISO image for Windows.

Complete the following step to create the boot CD ISO image.

To create a boot CD ISO image

- After you have named the boot CD file and specified the path, created a connection, and optionally injected the drivers, from the Create Boot CD screen, click **Create Boot CD**.

  The ISO image is then created and saved with the filename you provided.

Viewing the ISO image creation progress

This task is a step in the process of Defining boot CD ISO image parameters. It is part of the process for Creating a boot CD ISO image for Windows.

Complete the following step to view the progress of the creation of the ISO image.

To view the ISO image creation progress

- Select the Events tab, and then under Tasks, you can monitor the progress for building the ISO image. For more information about monitoring AppAssure events, see Viewing tasks, alerts, and events.

  When the creation of the ISO image is complete, it will appear on the Boot CDs page, accessible from the Tools menu.

Accessing the ISO image

This task is a step in Performing a bare metal restore for Windows machines. It is part of the process for Managing a Windows boot image.

Complete the following step to access the ISO image.

To access the ISO image

- To access the ISO image, navigate to the output path you specified, or you can click the link to download the image to a location from which you can then load it on the new system; for example, network drive.

Transferring the boot CD ISO image to media

When you create the boot CD file, it is stored as an ISO image in the path you specified. You must be able to mount this image as a drive on the server on which you are performing a bare metal restore.

You can burn the boot CD ISO image onto compact disc (CD) or digital video disk (DVD) media accessible at system startup.

When you start the machine from the boot CD, the Universal Recovery Console launches automatically.

If performing a BMR on a virtual machine, this step is not required. Simply load the ISO image in a drive and edit settings for that VM to start from that drive.

This task is a step in Performing a bare metal restore for Windows machines. It is part of the process for Managing a Windows boot image.

Loading the boot CD and starting the target machine

After you create the boot CD image, you need to boot the target server with the newly created boot CD.

NOTE: If you created the boot CD using DHCP, you must capture the IP address and password.

This task is a step in Performing a bare metal restore for Windows machines. It is part of the process for Managing a Windows boot image.

To load a boot CD and start the target machine

1. Navigate to the new server and load the boot CD image from the appropriate location. Specify that the server will start from the boot CD image.

2 Start the machine, which loads the following:

- Windows 7 PE
- AppAssure Agent software

The AppAssure Universal Recovery Console starts and displays the IP address and authentication password for the machine.

> **NOTE:** A new temporary password is generated each time the machine is started with the boot CD. Write down the IP address displayed in the Network Adapters Settings pane and the authentication password displayed in the Authentication pane. You will need this information later during the data recovery process to log back on to the console.

3 If you want to change the IP address, select it and click **Change**.

> **NOTE:** If you specified an IP address in the Create Boot CD dialog box, the Universal Recovery Console will use it and display it in the Network Adapter settings screen.

Once started with the boot CD, this machine is ready for the user to connect to it from the Core to begin the bare metal restore process.

Managing a Windows boot image and launching a BMR from the Restore Machine Wizard

Restoring volumes from a recovery point using a boot CD or restoring a system volume is considered performing a bare metal restore. Before performing a BMR, see Prerequisites for performing a bare metal restore for a Windows machine or Prerequisites for performing a bare metal restore for a Linux machine, as appropriate. If starting your BMR for a Windows machine from the Core Console, see Performing a bare metal restore for Windows machines.

If starting your BMR from the Restore Machine Wizard, then in the Destination page of that wizard, select the option **Recover to any target machine using a boot CD**, and then follow this procedure. Managing a Windows boot image through the wizard includes initiating creation of the boot cd; defining the path for the image on the Core machine; selecting the recovery environment appropriate to the hardware you want to restore on; optionally defining connection parameters for the restored agent for using the network or UltraVNC; optionally injecting drivers for hardware you want to restore on; and optionally transferring the boot image to physical media. This process also includes booting the machine to which you want to restore data from the CD; connecting to the Universal Recovery Console; mapping volumes; and initiating the bare metal restore from the selected recovery point on the core.

> **NOTE:** This process describes how to manage a boot CD image from the Restore Machine Wizard, as part of the process for performing a BMR using that wizard. You can also manage a boot image from the Create Boot CD dialog box. For information on managing a boot CD image outside of the Restore Machine Wizard, see Managing a Windows boot image.

To manage a Windows boot image and launch a BMR from the Restore Machines Wizard

1 On the Boot CD page, do the following:

a In the **Output path** text field, type the path where the boot CD ISO image should be stored.

If the shared drive on which you want to store the image is low on disk space, you can set the path as needed; for example, D:\filename.iso.

> **NOTE:** The file extension must be .iso. When specifying the path, use only alphanumeric characters, the hyphen, the backslash (only as a path delimiter), and the period (only to separate host names and domains). The letters a to z are case-insensitive. Do not use spaces. No other symbols or punctuation characters are permitted.

b Under **Environment,** select the architecture best suited for the hardware you are restoring:

- To restore on any Windows machine with a 64-bit architecture, select **Windows 8 64-bit (necessary for machines that are configured with a UEFI bios)**.
- To restore on any machine with a 32-bit (x86) architecture, select **Windows 7 32-bit**.

2 Optionally, to set up network parameters for the restored agent, or to use UltraVNC, select **Show advanced options** and do one of the following:

- To establish a network connection for the restored machine, select **Use the following IP address** as described in the following table.

Table 119. Network connection options

Option	Description
IP Address	Specify an IP address or host name for the restored machine.
Subnet Mask	Specify the subnet mask for the restored machine.
Default Gateway	Specify the default gateway for the restored machine.
DNS Server	Specify the domain name server for the restored machine.

- To define UltraVNC information, select **Add UltraVNC** as described in the following table.

  Use this option if you require remote access to the recovery console. You cannot log on using Microsoft Terminal Services while using the boot CD

Table 120. UltraVNC connection

Option	Description
Password	Specify a password for this UltraVNC connection.
Port	Specify a port for this UltraVNC connection. The default port is 5900.

3 When you are satisfied with your selections on the Boot CD page, click **Next**.

4 Optionally, on the Driver Injection page, to inject a driver, do the following:

a Select **Add an archive of drivers**

b Navigate to a ZIP file containing the archive, select the ZIP file, and click **Open**.

The archive uploads and appears in the Driver Injection page.

c Then click **Next**.

5 On the ISO Image page, you can see the status as the boot CD ISO image is created. When the boot CD is successful, click **Next**.

The Connection page appears.

6 At this point, you want to start the agent machine for which you want to restore data from the boot CD.

- If you can boot the agent machine from the boot CD ISO image, do so now.
- If not, copy the ISO image to physical media (a CD or DVD), load the disc in the agent machine, configure the machine to load from the boot CD, and restart from the boot CD.

  NOTE: You may need to change the BIOS settings of the agent machine to ensure the volume that loads first is the boot CD.

  The agent machine, when started from the boot CD, displays the Universal Recovery Console (URC) interface. This environment is used to restore the system drive or selected volumes directly from the AppAssure Core. Note the IP address and authentication key credentials in the URC, which refresh each time you start from the boot CD.

7 Back on the Core Console in the Connection page, enter authentication information from the URC instance of the machine you want to restore as follows:

   a In the IP Address text box, enter the IP address of the machine to which you are restoring from a recovery point.

   b In the Authentication Key text box, enter the information from the URC.

   c Then click **Next**.

   The Disk Mapping page appears.

8 If you want to map volumes automatically, do the following. If you want to map volumes manually, proceed to Step 9.

   a Select **Automatic volume mapping**.

   b In the Automatic volume mapping area, on the left side, select the volumes you want to restore. Optionally, if you do not wish to restore a listed volume, clear the option.

   NOTE: At least one volume must be selected to perform the restore.

   c On the right side, select the destination disk for the restore.

   d Click **Next**, and then proceed to Step 10.

9 If you want to map volumes manually, do the following:

   a Select **Manual volume mapping**.

   b In the Manual volume mapping area, from the Destination Volumes drop-down list for each volume, select the volume you want to restore. Optionally, if you do not wish to restore a listed volume, clear the option.

   NOTE: At least one volume must be selected to perform the restore.

   c When satisfied, click **Finish**.

CAUTION: If you select Finish, all existing partitions and data on the target drive will be removed permanently, and replaced with the contents of the selected recovery point, including the operating system and all data.

The Restore Machine Wizard closes, and the data is restored from the selected volumes of the recovery point to the target machine.

Proceed to Step 13.

10 In the Disk Mapping Preview page, review the parameters of the restore actions you selected. To perform the restore, click **Finish**.

CAUTION: If you select Finish, all existing partitions and data on the target drive will be removed permanently, and replaced with the contents of the selected recovery point, including the operating system and all data.

11 If the volumes you want to restore contain SQL or Microsoft Exchange databases, and if you are performing a Live Restore, then on the Dismount Databases page, you are prompted to dismount them. Optionally, if you want to remount these databases after the restore is complete, select **Automatically remount all databases after the recovery point is restored**. Then click **Finish**.

12 Click **OK** to confirm the status message that the restore process has started.

13 Optionally, to monitor the progress of your restore action, on the Core Console, click **Events**. For more information, see Viewing tasks, alerts, and events.

Launching a bare metal restore for Windows

Before launching a bare metal restore (BMR) for a Windows machine, certain conditions are required.

To restore a recovery point saved on the Core, you must have the appropriate hardware in place. For more information, see Prerequisites for performing a bare metal restore for a Windows machine.

The BMR destination Windows machine must be started using the boot CD image. For more information, see Managing a Windows boot image.

The first step is to select the appropriate recovery point, then initiate the restore to the hardware by specifying the IP address and temporary password you obtained from the Universal Recovery Console.

You must then map the drives and start the restore.

The recovery point includes drivers from the previous hardware. If restoring to dissimilar hardware, then you must inject storage controller drivers into the operating system being restored using the URC after the data has been restored to the drive, This allows the restored operating system to boot using the new set of hardware. Once the OS is booted after the restore, you can then download and install any additional drivers needed by the OS to interact with its new hardware.

To launch a BMR from the AppAssure Core Console, perform the following tasks.

- Selecting a recovery point and initiating BMR
- Mapping volumes for a bare metal restore
- Loading drivers using the Universal Recovery Console
- Injecting drivers to your target server

This process is a step in Performing a bare metal restore for Windows machines.

Selecting a recovery point and initiating BMR

Once the Universal Recovery Console is accessible on the machine on which you want to perform a BMR, you must select the recovery point that you want to restore. Navigate to the Core Console to select which recovery point you want to load, and designate the recovery console as the destination for the restored data.

NOTE: This step is required to perform BMR on all Windows machines and optional to perform BMR on Linux machines.

This task is a step in Performing a bare metal restore for Windows machines. It is part of the process for Launching a bare metal restore for Windows.

If performing a BMR for a Linux machine from the Core Console, then this task is also a step in Performing a bare metal restore for Linux machines It is part of the process for Launching a bare metal restore for a Linux machine using the command line.

Complete the steps in this procedure to select a recovery point on the Core to restore to the physical or virtual BMR target machine.

To select a recovery point and initiate BMR

1. Navigate to the AppAssure Core Console and, in the list of protected machines, click the name of the protected server you want to restore to bare metal.

   The Summary tab for the selected machine appears.

2. Click the Recovery Points tab.

3. In the list of recovery points, click the right angle bracket > symbol to expand the recovery point that you want to restore.

4 In the expanded details for that recovery point, from the Actions menu, click **Restore**.

The Restore Machine Wizard appears.

5 Select **Recover to any target machine using a boot CD**.

6 Select **I already have a boot CD running on the target machine**.

The authentication fields become accessible.

7 Enter the information about the machine to which you want to connect as described in the following table.

Table 121. Target machine information

Text Box	Description
IP Address	The IP address of the machine to which you want to restore. This is identical to the IP address displayed in the URC.
Authentication Key	The specific password to connect to the selected server. This is identical to the Authentication Key displayed in the URC.

8 Click **Next**.

If the connection information you entered matches the URC, and if the Core and the target server can identify each other properly on the network, then the volumes for the selected recovery point are loaded, and the Disk Mapping page appears, In this case, your next step is to map volumes.

NOTE: While AppAssure supports FAT32 and ReFS partitions, at present, only full restore and BMR are supported as a driver limitation exists with ReFS, so restore is implemented in user mode, VM export, and so on. If a Core is protecting at least one agent volume that contains the ReFS file system, it should be installed on Window 8, Windows 8.1, Windows Server 2012, and Windows Server 2012 R2 machines, which provide native support of ReFS. Otherwise, functionality will be limited and operations that involve such things as mounting a volume image will not work. The AppAssure Core Console will present applicable error messages in these occurrences.

Bare metal restore of Storage Spaces disks configuration (a feature of Windows 8.1) is also not supported in this release. For details, see the *Dell AppAssure Installation and Upgrade Guide*.

Mapping volumes for a bare metal restore

Once connected to the Universal Recovery Console, you will need to map volumes between those listed in the recovery point and volumes existing on the target hardware to perform the restore.

AppAssure attempts to automatically map volumes. If you accept the default mapping, then the disk on the destination machine is cleaned and re-partitioned and any previously existing data is deleted. The alignment is performed in the order the volumes are listed in the recovery point, and the volumes are allocated to the disks appropriately according to size, and so on. Assuming there is enough space on the target drive, no partitioning is required when using automatic disk alignment. A disk can be used by multiple volumes. If you manually map the drives, note that you cannot use the same disk twice.

For manual mapping, you must have the new machine correctly formatted already before restoring it. The destination machine must have a separate partition for each volume in the recovery point, including the system reserved volume. For more information, see Launching a bare metal restore for Windows.

This task is a step in Performing a bare metal restore for Windows machines. It is part of the process for Launching a bare metal restore for Windows.

If performing a BMR for a Linux machine from the Core Console, then this task is also a step in Performing a bare metal restore for Linux machines. It is part of the process for Launching a bare metal restore for Linux.

Complete the steps in this procedure to map a volume.

To map volumes for a bare metal restore

1 If you want to map volumes automatically, do the following. If you want to map volumes manually, proceed to Step 2

   a On the Disk Mapping page of the Restore Machine Wizard, select the Automatically Map Volumes tab.

   b In the Disk Mapping area, under Source Volume, verify that the source volume is selected, and that the appropriate volumes are listed beneath it and are selected.

   > **NOTE:** Typically for a BMR, you should restore, at minimum, the system reserved volume and the system volume (usually, but not always, the C:\ volume).

   c Optionally, if you do not wish to restore a listed volume, clear the option under Source volume. At least one volume must be selected to perform the BMR.

   d If the destination disk that is automatically mapped is the correct target volume, select **Destination Disk** and ensure that all appropriate volumes are selected.

   e Click **Restore**, and then proceed to Step 3.

2 If you want to map volumes manually, do the following:

   a On the Disk Mapping page of the Restore Machine Wizard, select the Manually Map Volumes tab.

   > **NOTE:** If no volumes exist on the drive of the machine on which you are performing a BMR, you will not be able to see this tab or manually map volumes.

   b In the Volume Mapping area, under Source Volume, verify that the source volume is selected, and that the appropriate volumes are listed beneath it and are selected.

   c Under Destination, from the drop-down menu, select the appropriate destination that is the target volume to perform the bare metal restore of the selected recovery point, and then click **Restore**.

3 In the confirmation dialog box, review the mapping of the source of the recovery point and the destination volume for the restore. To perform the restore, click **Begin Restore**.

> **CAUTION: If you select Begin Restore, all existing partitions and data on the target drive will be removed permanently, and replaced with the contents of the selected recovery point, including the operating system and all data.**

Loading drivers using the Universal Recovery Console

When creating a boot CD, you can add necessary drivers to the ISO image. After you boot into the target machine, you also can load storage or network drivers from within the Universal Recovery Console (URC). This feature lets you add any forgotten drivers that were not included in the ISO image but are required for a successful bare metal restore.

This task is a step in Performing a bare metal restore for Windows machines. It is part of the process for Launching a bare metal restore for Windows.

Complete the steps in the following procedure to load drivers using the URC.

To load drivers using the Universal Recovery Console

1 On an internet-connected machine, download and save the necessary drivers onto a portable media device, such as a USB drive.

2 Remove the media from the connected machine and insert it into the boot target server.

3 In the URC on the target server, on the Console tab, click **Load driver**.

4 In the Select driver window, navigate to the driver location, select the driver, and click **Open**.

5 Repeat as necessary for each driver you want to load.

Injecting drivers to your target server

If you are restoring to dissimilar hardware, you must inject storage controller, RAID, AHCI, chipset and other drivers if they are not already on the boot CD. These drivers make it possible for the operating system to operate all devices on your target server successfully once you reboot the system following the restore process.

If you are unsure which drivers are required by your target server, click the System Info tab in the Universal Recovery Console. This tab shows all system hardware and device types for the target server to which you want to restore.

NOTE: Your target server automatically contains some generic Windows 7 PE 32-bit drivers which will work for some systems.

This task is a step in Performing a bare metal restore for Windows machines. It is part of the process for Launching a bare metal restore for Windows.

Complete the following task to inject drivers to your target server.

To inject drivers to your target server

1. Download the drivers from the manufacturer's Web site for the server and unpack them.
2. Compress each driver into a .zip file using an appropriate compression utility (for example, WinZip) and copy it to the target server.
3. In the Universal Recovery Console, click **Driver Injection.**
4. Navigate through the filing system to locate the compressed driver file and select the file.
5. If you clicked Driver Injection in step 3, click **Add Driver**. If you clicked Load driver in step 3, click **Open**.

   The selected drivers are injected and will be loaded to the operating system after you reboot the target server.
6. Repeat Step 3 through Step 5, as appropriate, until all drivers have been injected.

Verifying a bare metal restore

Once you perform a bare metal restore, you can verify the progress of the restore. When the action is completed successfully, you can start the restored server. Some troubleshooting steps are included if you encounter difficulties connecting to the Universal Recovery Console to complete the restore, and to repair startup problems with the restored machine.

You can perform the following tasks:

- Viewing the recovery progress
- Starting a restored target server
- Troubleshooting connections to the Universal Recovery Console
- Repairing startup problems

This process is a step in Performing a bare metal restore for Windows machines.

Viewing the recovery progress

Complete the steps in this procedure to view the progress of restoring data from a recovery point (including bare metal restore) initiated from the AppAssure Core Console.

This task is a step in Performing a bare metal restore for Windows machines. It is part of the process for Verifying a bare metal restore.

To view the recovery progress

1. After you initiate the process restoring data from a recovery point, while the task is in process, you can view its progress from the Running Tasks drop-down menu on the Core Console.
2. Optionally, you can view detailed information in the Events tab. Fore more information about monitoring AppAssure events, see Viewing tasks, alerts, and events.

Starting a restored target server

Complete the steps in this procedure to start the restored target server.

NOTE: Before starting the restored target server, you should verify that the recovery was successful. For more information, see Viewing the recovery progress.

This task is a step in Performing a bare metal restore for Windows machines. It is part of the process for Verifying a bare metal restore.

To start a restored target server

1. Navigate back to the target server, and verify that the AppAssure Universal Recovery Console is active.
2. Eject the boot CD (or disconnect physical media with the boot CD image) from the restored server.
3. In the Universal Recovery Console, from the Console tab, click **Reboot**.
4. Specify to start the operating system normally.
5. Log on to the machine. The system should be restored to its state captured in the recovery point.

Troubleshooting connections to the Universal Recovery Console

The following are troubleshooting steps for connecting to the boot CD image as part of the process for Selecting a recovery point and initiating BMR.

This task is a step in Performing a bare metal restore for Windows machines. It is part of the process for Verifying a bare metal restore.

If an error displays indicating that the Core could not connect to the remote server, then any of several possible causes are likely.

- Verify that the IP address and Current Password displayed in the URC is identical to the information you entered in the Recovery Console Instance dialog box.
- To reach the server on which to restore data, the Core must be able to identify the server on the network. To determine if this is possible, you can open a command prompt on the Core and ping the IP address of the target BMR server. You can also open a command prompt on the target server and ping the IP address of the AppAssure Core.
- Verify that the network adapter settings are compatible between Core and target BMR server.

Repairing startup problems

Complete the steps in this procedure to repair startup problems. Keep in mind that if you restored to dissimilar hardware, you must have injected storage controller, RAID, AHCI, chipset and other drivers if they are not already on the boot CD. These drivers make it possible for the operating system to operate all devices on your target server successfully. For more information, see Injecting drivers to your target server.

This task is a step in Performing a bare metal restore for Windows machines. It is part of the process for Verifying a bare metal restore.

Complete the following procedure to repair startup problems on your target server.

To repair startup problems

1. Open the Universal Recovery Console by reloading the boot CD.
2. In the Universal Recovery Console, click **Driver Injection**.
3. In the Driver Injection dialog, click **Repair Boot Problems**.

   The startup parameters in the target server boot record are automatically repaired.
4. In the Universal Recovery Console, click **Reboot**.

16

Retention and archiving

This chapter describes how to manage aging data with retention policies and then archive it for long-term storage. It includes the following sections:

- Managing retention policies
- Understanding archives
- Archiving to a cloud

Periodic backup snapshots of all of its protected machines accumulate on the Core over time. Retention policies are used to retain backup snapshots for longer periods of time and to help with the management of these backup snapshots. After the aged data reaches its predetermined limit (for example, one year), you can either delete the data or archive it to one of several locations, including a cloud account. The following topics describe how to set up retention policies and how to archive data from a protected machine.

Managing retention policies

The retention policy is enforced by a nightly rollup process that helps in aging and deleting old backups. For information about configuring retention policies, see Customizing retention policy settings for a protected machine.

Configuring Core default retention policy settings

The retention policy for the Core specifies how long the recovery points for an agent machine are stored in the repository. This retention policy is enforced by a rollup process which is performed as one component of running nightly jobs. Then, recovery points beyond the age specified in the retention policy are “rolled up” (combined) into fewer recovery points that cover a less granular period of time. Applying the retention policy on a nightly basis results in the ongoing rollup of aging backups. This eventually results in the deletion of the oldest recovery points, based on the requirements specified in that retention policy.

Different retention settings can be configured for source and target cores.

NOTE: This topic is specific to customizing retention policy settings on the AppAssure Core. When you save customized retention policy settings on the Core, you establish the default retention policy settings which can be applied to all machines protected by this Core. For more information on customizing retention policy settings for individual protected machines, see Customizing retention policy settings for a protected machine.

To configure Core default retention policy settings

1 Navigate to the AppAssure Core Console, click the Configuration tab, and then click **Retention Policy**.

The Retention Policy screen appears, displaying the retention policy options for the Core.

2 Specify the primary setting that determines how long initial backup snapshots are retained, and then proceed to define a cascading set of rollup requirements that determines the intervals between when recovery points should be rolled up.

The retention policy options are described in the following table.

Table 122.

Text Box	Description
Keep all Recovery Points for *n* [*retention time period*]	Specifies the retention period for the recovery points. Enter a number to represent the retention period and then select the time period. The default is *3* days. You can choose from: Days, Weeks, Months, or Years
…and then keep one Recovery Point per hour for *n* [*retention time period*]	Provides a more granular level of retention. It is used as a building block with the primary setting to further define how long recovery points are maintained. Enter a number to represent the retention period and then select the time period. The default is *2* days. You can choose from: Days, Weeks, Months, or Years
…and then keep one Recovery Point per day for *n* [*retention time period*]	Provides a more granular level of retention. It is used as a building block to further define how long recovery points are maintained. Enter a number to represent the retention period and then select the time period. The default is *4* days. You can choose from: Days, Weeks, Months, or Years
…and then keep one Recovery Point per week for *n* [*retention time period*]	Provides a more granular level of retention. It is used as a building block to further define how long recovery points are maintained. Enter a number to represent the retention period and then select the time period. The default is *3* weeks. You can choose from: Weeks, Months, or Years
…and then keep one Recovery Point per month for *n* [*retention time period*]	Provides a more granular level of retention. It is used as a building block to further define how long recovery points are maintained. Enter a number to represent the retention period and then select the time period. The default is *2 months*. You can choose from: Months or Years
…and then keep one Recovery Point per year for *n* [*retention time period*]	Enter a number to represent the retention period and then select the time period. You can choose from: Years

The Newest Recovery Point text box displays the most recent recovery point. The oldest recovery point would be determined by the retention policy settings.

The following is an example of how the retention period is calculated.

Keep all recovery points for *3 days*.

…and then keep one recovery point per hour for *3 days*

…and then keep one recovery point per day for *4 days*

…and then keep one recovery point per week for *3 weeks*

…and then keep one recovery point per month for *2 months*

…and then keep one recovery point per month for *1 year*

Newest Recovery Point is set to the current day, month, and year.

In this example, the oldest recovery point would be 1 year, 4 months, and 6 days old.

3 Under Settings, in the **Number of simultaneous Rollups** text field, enter a numeric value.

This setting determines how many rollup operations can be performed at the same time. Setting the number above 1 will result in a shorter time to complete the rollup process, but will place a heavier load on the Core while rollups are occurring.

NOTE: As a rule of thumb, set this value to 1. If rollup operations take too long, increment by one digit, and check system performance to ensure the change is constructive in your environment.

4 Click **Apply**.

The retention policy you defined be applied during the nightly rollup.

You will also be able to apply these settings when specifying the retention policy for any individual agent machine. For more information about setting retention policies for an agent machine, see Customizing retention policy settings for a protected machine.

Customizing retention policy settings for a protected machine

The retention policy for an agent machine specifies how long recovery points are stored in the repository. Typically, each agent uses the default retention policy established for the core unless you specify a custom retention policy, as described in this procedure.

Starting with version 5.4.1, AppAssure includes the ability to set disparate retention policies between an agent on the source core and the corresponding replicated agent on the target core.

Use this procedure to define a custom retention policy for an agent, including a replicated agent.

NOTE: In environments upgrading from AppAssure release 5.3.x to release 5.4.1 or later, if you want to customize a retention policy for any replicated agent, you must first upgrade the source and target cores to release 5.4.1, and then perform the Integrity Check job on each repository in that target core. Completing this job is likely to take a substantial amount of time, based on the size of your repository and the underlying storage system. For information about this job, see About the repository Integrity Check job. For information on performing this job, see Running the Integrity Check job on a repository.

This task is also a step in the Modifying cluster node settings.

To customize retention policy settings for a protected machine

1 From the AppAssure Core Console, navigate to the machine that you want to modify.

2 Click the Configuration tab.

The Settings page displays.

3 In the Nightly Jobs pane, click **Change**.

The Nightly Jobs dialog box displays.

4 To specify the time intervals for retaining the backup data as needed, select the **Rollup** option and then click **Settings**.

The Configuration dialog box for retention policy displays.

5 If customizing retention policies settings for a replicated agent, and if you see a caution notifying you to perform an Integrity Check on your repository, proceed with this step. Otherwise, skip to Step 6.

a If you are prepared to perform the job, click **Check Integrity**

b Click **Yes** to confirm the Integrity Check job.

CAUTION: This could take a substantial amount of time, based on the size of your repository. During this time, you can perform no other actions (snapshots, replication, virtual export, and so on) in the repository. For information about this job, see About the repository Integrity Check job.

- Once the Check Integrity job completes all child job successfully, return to this procedure and continue with the next step.

6 In the Configuration dialog box, do one of the following:

- To use the default retention policy for this agent, select the **Use Core default retention policy** option, and then click **Save**. The default policy is applied to this agent.
- To define a custom retention policy for this agent, select the **Use custom retention policy** option, and then continue with the next step.

7 Enter the custom schedule for retaining the recovery points as described in the following table.

Table 123.

Text Box	Description
Keep all Recovery Points for *n* [*retention time period*]	Specifies the retention period for the recovery points. Enter a number to represent the retention period and then select the time period. The default is *3* days. You can choose from: Days, Weeks, Months, and Years
…and then keep one Recovery Point per hour for *n* [*retention time period*]	Provides a more granular level of retention. It is used as a building block with the primary setting to further define how long recovery points are maintained. Enter a number to represent the retention period and then select the time period. The default is *2 days*. You can choose from: Days, Weeks, Months, and Years
…and then keep one Recovery Point per day for *n* [*retention time period*]	Provides a more granular level of retention. It is used as a building block to further define how long recovery points are maintained. Enter a number to represent the retention period and then select the time period. The default is *4 days*. You can choose from: Days, Weeks, Months, and Years
…and then keep one Recovery Point per week for *n* [*retention time period*]	Provides a more granular level of retention. It is used as a building block to further define how long recovery points are maintained. Enter a number to represent the retention period and then select the time period. The default is *3 weeks*. You can choose from: Weeks, Months, and Years
…and then keep one Recovery Point per month for *n* [*retention time period*]	Provides a more granular level of retention. It is used as a building block to further define how long recovery points are maintained. Enter a number to represent the retention period and then select the time period. The default is *2 months*. You can choose from: Months and Years
…and then keep one Recovery Point per year for *n* [*retention time period*]	Enter a number to represent the retention period and then select the time period. You can choose from: Years

The Newest Recovery Point text box displays the most recent recovery point. The oldest recovery point would be determined by the retention policy settings.

The following is an example of how the retention period is calculated.

Keep all recovery points for *3 days*.

…and then keep one recovery point per hour for *3 days*

…and then keep one recovery point per day for *4 days*

…and then keep one recovery point per week for *3 weeks*

…and then keep one recovery point per month for *2 months*

…and then keep one recovery point per month for *1 year*

Newest Recovery Point is set to the current day, month, and year.

In this example, the oldest recovery point would be 1 year, 4 months, and 6 days old.

8 Click **Save**.

Understanding archives

Retention policies enforce the periods for which backups are stored on short-term (fast and expensive) media. Sometimes certain business and technical requirements mandate extended retention of these backups, but use of fast storage is cost prohibitive. Therefore, this requirement creates a need for long-term (slow and cheap) storage. Businesses often use long-term storage for archiving both compliance and non-compliance data. The archive feature in AppAssure is used to support the extended retention for compliance and non-compliance data; and it is also used to seed replication data to a remote replica core.

Creating an archive

Complete the steps in this procedure to create an archive.

To create an archive

1 Navigate to the AppAssure Core Console, and then click the Tools tab.

2 From the Archive option, click **Create**.

The Add Archive Wizard opens.

3 On the Create page of the Add Archive Wizard, select one of the following options from the Location Type drop-down list:

- Local
- Network
- Cloud

4 Enter the details for the archive as described in the following table based on the location type you selected in Step 3.

Table 124.

Option	Text Box	Description
Local	Output location	Enter the location for the output. It is used to define the location path where you want the archive to reside; for example, d:\work\archive.
Network	Output location	Enter the location for the output. It is used to define the location path where you want the archive to reside; for example, \\servername\sharename.
	User Name	Enter a user name. It is used to establish logon credentials for the network share.
	Password	Enter a password for the network path. It is used to establish logon credentials for the network share.

Table 124.

Option	Text Box	Description
Cloud	Account	Select an account from the drop-down list. **NOTE:** To select a cloud account, you must first have added it in the Core Console. For more information, see Adding a cloud account.
	Container	Select a container associated with your account from the drop-down menu.
	Folder Name	Enter a name for the folder in which the archived data is to be saved. The default name is AppAssure-5-Archive-[DATE CREATED]-[TIME CREATED]

5 Click **Next**.

6 On the Machines page of the wizard, select which protected machine or machines contains the recovery points you want to archive.

7 Click **Next**.

8 On the Options page, enter the information described in the following table.

Text Box	Description
Maximum Size	Large archives of data can be divided into multiple segments. Select the maximum amount of space you want to reserve for creating the archive by doing one of the following: • Select **Entire Target** to reserve all available space in the path provided on the destination provided in Step 4. (for example, if the location is D:\work\archive, all of the available space on the D: drive is reserved). • Select the blank text box, use the up and down arrows to enter an amount, and then select a unit of measurement from the drop-down list to customize the maximum space you want to reserve. **NOTE:** Amazon cloud archives are automatically divided into 50 GB segments. Microsoft Azure cloud archives are automatically divided into 200 GB segments.
Recycle action	Select one of the following recycle action options: • **Do not reuse.** Does not overwrite or clear any existing archived data from the location. If the location is not empty, the archive write fails. • **Replace this Core.** Overwrites any pre-existing archived data pertaining to this core but leaves the data for other cores intact. • **Erase completely.** Clears all archived data from the directory before writing the new archive. • **Incremental.** Lets you add recovery points to an existing archive. It compares recovery points to avoid duplicating data that already exists in the archive.
Comment	Enter any additional information that is necessary to capture for the archive. The comment will be displayed if you import the archive later.
Use compatible format	Select this option to archive your data in a format that is compatible with previous versions of cores. **NOTE:** The new format offers better performance; however it is not compatible with older cores.

9 Click **Next**.

10 On the Date Range page, enter the Start Date and Expiration Date of the recovery points to be archived.

- To enter a time, click on the time shown (default, 8:00 AM) to reveal the slide bars for selecting hours and minutes.
- To enter a date, click the text box to reveal the calendar, and then click on the preferred day.

11 Click **Finish**.

Setting a scheduled archive

The Scheduled Archive feature lets you set a time when an archive of a selected machine will be automatically created and saved to the specified location. This accommodates situations where you would want frequent archives of a machine to be saved, without the inconvenience of needing to create the archives manually. Complete the steps in the following procedure to schedule automatic archiving.

To set a scheduled archive

1 Navigate to the AppAssure Core Console, and then click the Tools tab.

2 From the Archive option, click **Scheduled**.

3 On the Scheduled Archive page, click **Add**.

The Add Archive Wizard opens.

4 On the Location page of the Add Archive Wizard, select one of the following options from the Location Type drop-down list:

- Local
- Network
- Cloud

5 Enter the details for the archive as described in the following table based on the location type you selected in Step 4.

Table 125.

Option	Text Box	Description
Local	Output location	Enter the location for the output. It defines the location path where you want the archive to reside; for example, d:\work\archive.
Network	Output location	Enter the location for the output. It defines the location path where you want the archive to reside; for example, \\servername\sharename.
	User Name	Enter a user name. It establishes logon credentials for the network share.
	Password	Enter a password for the network path. It establishes logon credentials for the network share.
Cloud	Account	Select an account from the drop-down list. **NOTE:** To select a cloud account, you must first have added it in the Core Console. For more information, see Adding a cloud account.
	Container	Select a container associated with your account from the drop-down menu.
	Folder Name	Enter a name for the folder in which the archived data is to be saved. The default name is AppAssure-5-Archive-[DATE CREATED]-[TIME CREATED]

6 Click **Next**.

7 On the Machines page of the wizard, select which protected machines contain the recovery points you want to archive.

8 Click **Next**.

9 On the Options page, select one of the following Recycle Actions from the drop-down list:

- **Replace this Core.** Overwrites any existing archived data pertaining to this core but leaves the data for other cores intact.
- **Erase completely.** Clears all archived data from the directory before writing the new archive.
- **Incremental.** Lets you add recovery points to an existing archive. It compares recovery points to avoid duplicating data that already exists in the archive.

10 On the Schedule page, select one of the following Send data frequency options:

- Daily
- Weekly
- Monthly

11 Enter the information described in the following table based on your selection from Step 10.

Option	Text Box	Description
Daily	At time	Select the hour of the day you want to create a daily archive.
Weekly	At day of week	Select a day of the week on which to automatically create the archive.
	At time	Select the hour of the day you want to create a daily archive.
Monthly	At day of months	Select the day of the month on which to automatically create the archive.
	At time	Select the hour of the day you want to create a daily archive.

12 To pause archiving for resuming at a later time, select **Initial pause archiving**.

> **NOTE:** You may want to pause the scheduled archive if you need time to prepare the target location before archiving resumes. If you do not select this option, archiving begins at the scheduled time.

13 Click **Finish**.

Pausing or resuming a scheduled archive

There may be times when you want to pause a scheduled archive job, such as if you need to make changes to the destination archive location. Also, if you opted to initially pause archiving when you performed the Setting a scheduled archive procedure, you likely want to resume the scheduled archive at a later time. Complete the steps in the following procedure to pause or resume scheduled archive.

To pause or resume scheduled archive

1. Navigate to the AppAssure Core Console, and then click the Tools tab.
2. From the Archive option, click **Scheduled**.
3. On the Scheduled Archive page, do one of the following:
   - Select the preferred archive, and then click one of the following actions as appropriate:
     - Pause
     - Resume
   - Next to the preferred archive, click the drop-down menu, and then click one of the following actions as appropriate:
     - Pause
     - Resume

   The status of the archive displays in the Schedule column.

Editing a scheduled archive

AppAssure lets you change the details of a scheduled archive. To edit a scheduled archive, complete the steps in the following procedure.

To edit a schedule archive

1. Navigate to the AppAssure Core Console, and then click the Tools tab.
2. From the Archive option, click **Scheduled**.
3. On the Scheduled Archive page, click the drop-down menu next to the archive you want to change, and then click **Edit**.

   The Add Archive Wizard opens.
4. On the Location page of the Add Archive Wizard, select one of the following options from the Location Type drop-down list:
   - Local
   - Network
   - Cloud
5. Enter the details for the archive as described in the following table based on the location type you selected in Step 4.

Table 126.

Option	Text Box	Description
Local	Output location	Enter the location for the output. It is used to define the location path where you want the archive to reside; for example, d:\work\archive.

Table 126.

Option	Text Box	Description
Network	Output location	Enter the location for the output. It is used to define the location path where you want the archive to reside; for example, \\servername\sharename.
	User Name	Enter a user name. It is used to establish logon credentials for the network share.
	Password	Enter a password for the network path. It is used to establish logon credentials for the network share.
Cloud	Account	Select an account from the drop-down list. NOTE: To select a cloud account, you must first have added it in the Core Console. For more information, see Adding a cloud account.
	Container	Select a container associated with your account from the drop-down menu.
	Folder Name	Enter a name for the folder in which the archived data is to be saved. The default name is AppAssure-5-Archive-[DATE CREATED]-[TIME CREATED]

6 Click **Next**.

7 On the Machines page of the wizard, select which protected machines contain the recovery points you want to archive.

8 Click **Next**.

9 On the Schedule page, select one of the following Send data frequency options:

- Daily
- Weekly
- Monthly

10 Enter the information described in the following table based on your selection from Step 9.

Option	Text Box	Description
Daily	At time	Select the hour of the day you want to create a daily archive.
Weekly	At day of week	Select a day of the week on which to automatically create the archive.
	At time	Select the hour of the day you want to create a daily archive.
Monthly	At day of months	Select the day of the month on which to automatically create the archive.
	At time	Select the hour of the day you want to create a daily archive.

11 To pause archiving for resuming at a later time, select **Initial pause archiving**.

NOTE: You may want to pause the scheduled archive if you need time to prepare the target location before archiving resumes. If you do not select this option, archiving begins at the scheduled time.

12 Click **Finish**.

Checking an archive

You can scan an archive for structural integrity by performing an archive check. This check verifies the presence of all necessary files within the archive. To perform an archive check, complete the steps in the following procedure.

To check an archive

1 Navigate to the AppAssure Core Console, and then click the Tools tab.

2 From the Archive option, click **Check Archive**.

3 The Check Archive dialog box appears.

4 For **Location type**, select one of the following options from the drop-down list:

- Local
- Network
- Cloud

5 Enter the details for the archive as described in the following table based on the location type you selected in Step 4.

Table 127.

Option	Text Box	Description
Local	Location	Enter the path for the archive.
Network	Location	Enter the path for the archive.
	User Name	Enter the user name. It is used to establish logon credentials for the network share.
	Password	Enter the password for the network path. It is used to establish logon credentials for the network share.
Cloud	Account	Select an account from the drop-down list. **NOTE:** To select a cloud account, you must first have added it in the Core Console. For more information, see Adding a cloud account.
	Container	Select a container associated with your account from the drop-down menu.
	Folder Name	Enter the name of the folder in which the archived data is saved; for example, AppAssure-5-Archive-[DATE CREATED]-[TIME CREATED]

6 To also perform a structure integrity check, select **Structure integrity**.

7 Click **Check File**.

Importing an archive

When you want to recover archived data, you must import the entire archive to a specified location. Afterwards, you are able to browse the data. To import an archive, complete the steps in the following procedure.

To import an archive

1 Navigate to the AppAssure Core Console, and then select the Tools tab.

2 From the Archive option, click **Import**.

3 For **Location type**, select one of the following options from the drop-down list:

- Local
- Network
- Cloud

4 Enter the details for the archive as described in the following table based on the location type you selected in Step 3.

Table 128.

Option	Text Box	Description
Local	Location	Enter the path for the archive.
Network	Location	Enter the path for the archive.
	User Name	Enter the user name. It is used to establish logon credentials for the network share.
	Password	Enter the password for the network path. It is used to establish logon credentials for the network share.
Cloud	Account	Select an account from the drop-down list. **NOTE:** To select a cloud account, you must first have added it in the Core Console.
	Container	Select a container associated with your account from the drop-down menu.
	Folder Name	Enter the name of the folder in which the archived data is saved; for example, AppAssure-5-Archive-[DATE CREATED]-[TIME CREATED]

5 Click **Check File** to validate the existence of the archive to import.

6 In the dialog box, verify the name of the source core shown from the Core drop-down list.

7 Select the agents to be imported from the archive.

8 Select the repository in which to save the archived data.

NOTE: The repository selected must be the same repository in which all current recovery points for the selected agent are stored.

9 Click **Restore** to import the archive.

Archiving to a cloud

When data reaches the end of a retention period, you may want to extend that retention by creating an archive of the aged data. When you archive data, there is always the matter of where to store it. AppAssure lets you upload your archive to a variety of cloud providers directly from the Core Console. Compatible clouds include Microsoft Azure, Amazon, Rackspace, and any OpenStack-based provider.

Exporting an archive to a cloud using AppAssure involves the following procedures:

- **Add your cloud account to the AppAssure Core Console.** For more information, see Adding a cloud account.
- **Archive your data and export it to your cloud account.** For more information, see Creating an archive.
- **Retrieve archived data by importing it from the cloud location.** For more information, see Importing an archive.

17

Managing cloud accounts

This chapter describes how to define links to existing cloud storage provider accounts, and how to manage those cloud accounts for use with AppAssure. For example, you can archive AppAssure data to the cloud, or import archived data from the cloud. This chapter includes the following sections:

- Adding a cloud account
- Editing a cloud account
- Configuring cloud account settings
- Removing a cloud account

AppAssure lets you archive data to a variety of cloud providers, or import archived data stored in a cloud account. Compatible clouds include Microsoft Azure, Amazon, Rackspace, and any OpenStack-based provider.

You can add an existing cloud account to the AppAssure Core console. Once added, you can edit the information, configure the account connection options, or remove the account from AppAssure.

Adding a cloud account

Before you can export your archived data to a cloud, you must add the account information for your cloud provider to the AppAssure Core Console. To add a cloud account, complete the steps in the following procedure.

To add a cloud account

1 On the AppAssure Core Console, click the Tools tab.

2 In the left menu, click **Clouds**.

3 On the Clouds page, click **Add New Account**.

The Add New Account dialog box opens.

4 Select a compatible cloud provider from the Cloud Type drop-down list.

5 Enter the details described in the following table based on the cloud type selected in Step 4.

Table 129. Cloud account details

Cloud Type	Text Box	Description
Microsoft Azure	Storage Account Name	Enter the name of your Microsoft Azure storage account.
	Access Key	Enter the access key for your account.
	Display Name	Create a display name for this account in AppAssure; for example, Microsoft Azure 1.
Amazon S3	Access Key	Enter the access key for your Amazon cloud account.
	Secret Key	Enter the secret key for this account.
	Display Name	Create a display name for this account in AppAssure; for example, Amazon 1.

Cloud Type	Text Box	Description
Powered by OpenStack	User Name	Enter the user name for you OpenStack-based cloud account.
	API Key	Enter the API key for your account.
	Display Name	Create a display name for this account in AppAssure; for example, OpenStack 1.
	Tenant ID	Enter your tenant ID for this account.
	Authentication URL	Enter the authentication URL for this account.
Rackspace Cloud Block Storage	User Name	Enter the user name for your Rackspace cloud account.
	API Key	Enter the API key for this account.
	Display Name	Create a display name for this account in AppAssure; for example, Rackspace 1.

6 Click **Add**.

The dialog box closes, and your account appears on the Clouds page of the Core Console.

Editing a cloud account

If you need to change the information to connect to your cloud account, for example to update the password or edit the display name, you can do so on the Tools tab of the Core Console. Complete the steps in the following procedure to edit a cloud account.

To edit a cloud account

1 On the AppAssure Core Console, click the Tools tab.

2 In the left menu, click **Clouds**.

3 Next to the cloud account you want to edit, click the drop-down menu, and then click **Edit**.

The Edit Account window opens.

4 Edit the details as necessary, and then click **Save**.

NOTE: You cannot edit the cloud type.

Configuring cloud account settings

The cloud configuration settings let you determine the number of times AppAssure should attempt to connect to your cloud account, and how much time should pass for those attempts before they time out. Complete the steps in the following procedure to configure the connection settings for your cloud account.

To configure cloud account settings

1 On the AppAssure Core Console, click the Configuration tab.

2 In the left menu, click **Settings**.

3 On the Settings page, scroll down to Cloud Configuration.

4 Click the drop-down menu next to the cloud account you want to configure, and then do one of the following:

- Click **Edit**.

  The Cloud Configuration dialog box appears.

a Use the up and down arrows to edit either of the following options:

- **Request Timeout.** Displayed in minutes and seconds, it determines the amount of time AppAssure should spend on a single attempt to connect to the cloud account when there is a delay. Connection attempts will cease after the entered amount of time.
- **Retry Count.** Determines the number of attempts AppAssure should conduct before determining that the cloud account cannot be reached.
- **Write Buffer Size.** Determines the buffer size reserved for writing archived data to the cloud.
- **Read Buffer Size.** Determines the block size reserved for reading archived data from the cloud.

b Click **Next**.

- Click **Reset**.

  Returns the configuration to the following default settings:

  - **Request Timeout:** 01:30 (minutes and seconds)
  - **Retry Count:** 3 (attempts)

Removing a cloud account

If you discontinue your cloud service, or decide to stop using it for a particular Core, you may want to remove your cloud account from the Core Console. Complete the steps in the following procedure to remove a cloud account.

To remove a cloud account

1 On the AppAssure Core Console, click the Tools tab.

2 In the left menu, click **Clouds**.

3 Next to the cloud account you want to edit, click the drop-down menu, and then click **Remove**.

4 In the Delete Account window, click **Yes** to confirm that you want to remove the account.

5 If the cloud account is currently in use, a second window asks you if you still want to remove it. Click **Yes** to confirm.

> **NOTE:** Removing an account that is currently in use causes all archive jobs scheduled for this account to fail.

18

Working with Linux machines

This chapter describes how to protect, configure, and manage the protected Linux machines in your AppAssure environment. It includes the following sections:

- Working with Linux recovery points
- Exporting data to a Linux-based VirtualBox virtual machine
- Restoring volumes for a Linux machine using the command line
- Performing a bare metal restore for Linux machines
- Managing a Linux boot image
- Managing Linux partitions
- Launching a bare metal restore for Linux
- Verifying the bare metal restore from the command line

The AppAssure Agent software is compatible with multiple Linux-based operating systems (for details, see the *Dell AppAssure Installation and Upgrade Guide*). The AppAssure Core is compatible only with Windows machines. While you can manage protected Linux machines from the AppAssure Core Console, several procedures for Linux machines have steps that differ from their Windows counterparts. Additionally, you can perform some actions directly on a protected Linux machine by using the command line aamount utility.

Working with Linux recovery points

The recommended and supported way to mount and unmount recovery points from a protected Linux machine is to use the aamount utility. For more information, see the following procedures:

- Mounting a recovery point volume on a Linux machine
- Unmounting a recovery point on a Linux machine

These procedures specifically address using aamount to mount and unmount Linux recovery points. For managing Linux recovery points in any other way, see Managing snapshots and recovery points, as all other management can be conducted from the Core Console.

Mounting a recovery point volume on a Linux machine

Using the aamount utility in AppAssure, you can remotely mount a volume from a recovery point as a local volume on a Linux machine using the aamount utility.

NOTE: When performing this procedure, do not attempt to mount recovery points to the /tmp folder, which contains the aavdisk files.

To mount a recovery point volume on a Linux machine

1. Create a new directory for mounting the recovery point (for example, you can use the `mkdir` command).
2. Verify the directory exists (for example, by using the `ls` command).

3 Run the AppAssure aamount utility as root, or as the super user, for example:

```
sudo aamount
```

4 At the AppAssure mount prompt, enter the following command to list the protected machines.

```
lm
```

5 When prompted, enter the IP address or hostname of your AppAssure Core server.

6 Enter the logon credentials for the Core server, that is, the user name and password.

A list of the machines that are protected by the AppAssure server will display. Each machine is identified by the following: line item number, host/IP address, and an ID number for the machine.

For example: 293cc667-44b4-48ab-91d8-44bc74252a4f

7 Enter the following command to list the recovery points that are available for a specified machine:

```
lr <line_number_of_machine>
```

> **NOTE:** Note that you can also enter the machine ID number in this command instead of the line item number.

A list of the base and incremental recovery points for the machine appears. The list includes the line item number, date and timestamp, location of volume, size of recovery point, and an ID number for the volume, which includes a sequence number at the end to identify the recovery point.

For example, 293cc667-44b4-48ab-91d8-44bc74252a4f:2

8 Enter the following command to select and mount the specified recovery point at the specified mount point/path.

```
m <volume_recovery_point_ID_number> <volume-letter> [flag] <path>
```

The flag in the command determines how to mount the recovery point. You can use one of the following options:

- [r] - mount read-only (default). This flag lets you mount a recovery point but does not let you make changes to it.
- [w] - mount writable. This flag lets you mount the recovery point and lets you make changes.
- [v] - mount with previous writes. Mounting with the “v” flag lets you mount the recovery point and include any changes that were made during the previous writable mount but are not present in the recovery point.
- [n] - do not mount nbd to <path>. A nbd (network block device) makes a socket connection between the Core and the protected machine when you perform a local mount. This flag lets you mount the recovery point without mounting the nbd, which is useful if you want to manually check the file system of the recovery point.

> **NOTE:** You can also specify a line number in the command instead of the recovery point ID number to identify the recovery point. In that case, you would use the machine line number (from the lm output), followed by the recovery point line number and volume letter, followed by the path, such as, m <machine_line_number> <recovery_point_line_number> <volume_letter> <path>. For example, if the `lm` output lists three protected machines, and you enter the `lr` command for number 2 and you mount the twenty-third recovery point volume b to /tmp/mount_dir, then the command would be:
> `m 2 23 b /tmp/mount_dir`

9 To verify that the mount was successful, enter the following command, which should list the attached remote volume:

```
l
```

Unmounting a recovery point on a Linux machine

Complete the steps in this procedure to unmount a recovery point on a Linux machine.

To unmount a recovery point on a Linux machine

1. Run the AppAssure aamount utility as root, or as the super user, for example:

   ```
   sudo aamount
   ```

2. At the AppAssure mount prompt, enter the following command to list the protected machines.

   ```
   lm
   ```

3. When prompted, enter the IP address or hostname of your AppAssure Core server.
4. Enter the logon credentials (user name and password) for the Core server.

   A list of the machines that are protected by the AppAssure server will display.

5. Enter the following command to list the recovery points that are available for a specified machine:

   ```
   lr <line_number_of_machine>
   ```

   NOTE: Note that you can also enter the machine ID number in this command instead of the line item number.

   A list of the base and incremental recovery points for the machine will display and includes. The list includes the line item number, date/timestamp, location of volume, size of recovery point, and an ID number for the volume that includes a sequence number at the end, which identifies the recovery point.

   For example, 293cc667-44b4-48ab-91d8-44bc74252a4f:2

6. Run the `l or list` command to obtain a list of mounted Network Block Device (NBD)-devices. If you mount any recovery point, you will get a path to NBD-device after executing the `l or list` command.
7. Enter the following command to unmount a recovery point.

   ```
   unmount <path_of_nbd-device>
   ```

8. Run the `l or list` command to verify that the unmount of the recovery point was successful.

Exporting data to a Linux-based VirtualBox virtual machine

With AppAssure you can export data to a Windows- or Linux-based virtual machine (VM) using VirtualBox. An export includes all of the backup information from a recovery point as well as the parameters defined for the protection schedule for your machine.

NOTE: Virtual Box Version 4.2.18 or higher is supported. Export to a Linux-based VirtualBox VM requires an SSH connection from the Core to the Linux machine.

NOTE: To export to a Windows-based VirtualBox VM, see Exporting data to a VirtualBox virtual machine.

CAUTION: **AppAssure does not support exporting machines that have extended partitions.**

Performing a one-time VirtualBox export

Complete the steps in this procedure to perform a one-time export to VirtualBox.

To perform a one-time VirtualBox export

1 In the AppAssure Core Console, navigate to the Linux machine you want to export.

2 On the Summary tab, in the Actions drop-down menu for that machine, click **Export**, and then select **One-time**.

The Export Wizard displays on the Protected Machines page.

3 Select a machine for export, and then click **Next**.

4 On the Recovery Points page, select the recovery point that you want to export, and then click **Next**.

5 On the Destination page in the Export Wizard, in the Recover to Virtual machine drop-down menu, select **VirtualBox**, and then click **Next**.

6 On the Virtual Machine Options page, select **Remote Linux Machine**.

7 Enter information about the virtual machine as described in the following table.

Table 130. Remote Linux machine settings

Option	Description
VirtualBox Host Name	Enter an IP address or host name for the VirtualBox server. This field represents the IP address or host name of the remote VirtualBox server.
Port	Enter a port number for the machine. This number represents the port through which the Core communicates with this machine.
Virtual Machine Name	Enter a name for the virtual machine being created. **NOTE:** The default name is the name of the source machine.
Target Path	Specify a target path to create the virtual machine. **NOTE:** It is recommended that you create a root folder from root so that the virtual machine runs from root. If you do not use root, you will need to create the destination folder manually on the target machine prior to setting up the export. You will also need to manually attach or load the virtual machine after the export.
User Name	User name of the account on the target machine, for example, root.
Password	Password for the user account on the target machine.
Memory	Specify the memory usage for the virtual machine by clicking one of the following: • **Use the same amount of RAM as source machine** • **Use a specific amount of RAM, and then specify the amount in MB** The minimum amount is 1024 MB and the maximum allowed by the application is 65536 MB. The maximum amount of memory usage is limited by the amount of RAM available to the host machine.

8 On the Volumes page, select the volumes of data to export, and then click **Next**.

9 On the Summary page, click **Finish** to complete the wizard and to start the export.

NOTE: You can monitor the status and progress of the export by viewing the Virtual Standby or Events tab.

Performing a continual (Virtual Standby) VirtualBox export

Complete the steps in this procedure to create a Virtual Standby and perform a continual export to VirtualBox.

To perform a continual (virtual standby) VirtualBox export

1 In the AppAssure Core Console, do one of the following:

- On the Virtual Standby tab, click **Add** to launch the Export Wizard. On the Protected Machines page of the Export Wizard, select the protected machine you want to export, and then click **Next**.
- Navigate to the machine you want to export, and, on the Summary tab in the Actions drop-down menu for that machine, click **Export** > **Virtual Standby**.

2 On the Destination page of the Export Wizard, in the Recover to a Virtual Machine drop-down menu, select **VirtualBox**.

3 On the Virtual Machine Options page, select **Remote Linux Machine**.

4 Enter information about the virtual machine as described in the following table.

Table 131. Remote Linux machine settings

Option	Description
VirtualBox Host Name	Enter an IP address or host name for the VirtualBox server. This field represents the IP address or host name of the remote VirtualBox server.
Port	Enter a port number for the machine. This number represents the port through which the Core communicates with this machine.
Virtual Machine Name	Enter a name for the virtual machine being created. **NOTE:** The default name is the name of the source machine.
Target Path	Specify a target path to create the virtual machine. **NOTE:** It is recommended that you create a root folder from root so that the virtual machine runs from root. If you do not use root, you will need to create the destination folder manually on the target machine prior to setting up the export. You will also need to manually attach or load the virtual machine after the export.
User Name	User name of the account on the target machine, for example, root.
Password	Password for the user account on the target machine.
Memory	Specify the memory usage for the virtual machine by clicking one of the following: • **Use the same amount of RAM as source machine** • **Use a specific amount of RAM, and then specify the amount in MB** The minimum amount is 1024 MB and the maximum allowed by the application is 65536 MB. The maximum amount of memory usage is limited by the amount of RAM available to the host machine.

5 Select **Perform initial one-time export** to perform the virtual export immediately instead of after the next scheduled snapshot.

6 Click **Next**.

7 On the Volumes page, select the volumes of data to export, and then click **Next**.

8 On the Summary page, click **Finish** to complete the wizard and to start the export.

NOTE: You can monitor the status and progress of the export by viewing the Virtual Standby or Events tab.

Restoring volumes for a Linux machine using the command line

In AppAssure, you can restore volumes on your protected Linux machines using the command line aamount utility.

NOTE: This process was previously referred to as Rollback.

When performing this procedure, do not attempt to mount recovery points to the /tmp folder, which contains the aavdisk files.

Restoring volumes is also supported for your protected machines within the AppAssure Core Console. See Restoring volumes from a recovery point for more information.

CAUTION: To restore the system or root (/) partition or entire operating system, see Performing a bare metal restore for Linux machines.

To restore volumes for a Linux machine using the command line

1. Run the AppAssure aamount utility as root, for example:

   ```
   sudo aamount
   ```

2. At the AppAssure mount prompt, enter the following command to list the protected machines.

   ```
   lm
   ```

3. When prompted, enter the IP address or hostname of your AppAssure Core server.
4. Enter the logon credentials, that is, the user name and password, for this server.

   A list displays showing the machines protected by this AppAssure server. It lists the agent machines found by line item number, host/IP address, and an ID number for the machine (for example: 293cc667-44b4-48ab-91d8-44bc74252a4f).

5. Enter the following command to list the currently mounted recovery points for the specified machine:

   ```
   lr <machine_line_item_number>
   ```

   NOTE: Note that you can also enter the machine ID number in this command instead of the line item number.

   A list displays that shows the base and incremental recovery points for that machine. This list includes a line item number, date/timestamp, location of volume, size of recovery point, and an ID number for the volume that includes a sequence number at the end (for example, ”293cc667-44b4-48ab-91d8-44bc74252a4f:2”), which identifies the recovery point.

6. Enter the following command to select a recovery point to restore:

   ```
   r <volume_recovery_point_ID_number> <path>
   ```

   This command restores the volume image specified by the ID from the Core to the specified path The path for the restore is the path for the device file descriptor and is not the directory to which it is mounted.

NOTE: You can also specify a line number in the command instead of the recovery point ID number to identify the recovery point. In that case, you would use the agent/machine line number (from the lm output), followed by the recovery point line number and volume letter, followed by the path, such as, r <machine_line_item_number> <recovery_point_line_number> <volume_letter> <path>. In this command, <path> is the file descriptor for the actual volume.

For example, if the `lm` output lists three agent machines, and you enter the `lr` command for number `2`, and you want to restore the `23` recovery point volume `b` to the volume that was mounted to the directory `/mnt/data`, the command would be:

```
r2 23 b /mnt/data
```

It is possible to restore to /, but only when performing a Bare Metal Restore while booted with a Live DVD. For more information, see Launching a bare metal restore for Linux.

7 When prompted to proceed, enter `y` for Yes.

Once the restore proceeds, a series of messages will display to notify you of the status.

8 Upon a successful restore, the aamount utility will automatically mount and re-attach the kernel module to the restored volume if the target was previously protected and mounted. If not, you will need to mount the restored volume to the local disk and then should verify that the files are restored (for example, you can use the `sudo mount` command and then the `ls` command.)

Performing a bare metal restore for Linux machines

In AppAssure you can perform a Bare Metal Restore (BMR) for a Linux machine, including a restore of the system volume. When you restore a Linux machine, you will roll back to the boot volume recovery point. BMR functionality is supported using the command line aamount utility and from within the Core Console UI.

△ **CAUTION: Before you begin the BMR process, be sure that any Linux machine you want to restore does not include an EXT2 boot partition. Any BMR performed on a machine with this type of partition results in a machine that does not start. To perform a BMR in this case, you would have needed to convert any EXT2 partitions to EXT3 or EXT4 before you began protecting and backing up the machine.**

CAUTION: When you boot a restored Linux machine for the first time after a BMR, AppAssure takes a base image of the restored machine. Depending on the amount of data on the machine, this process takes more time than taking an incremental snapshot. For more information about base images and incremental snapshots, see Understanding protection schedules.

To perform a bare metal restore for Linux machines, perform the following tasks.

- **Manage a Linux boot image**. This Linux Live DVD boot ISO image is used to start up the destination drive, from which you can access the Universal Recovery Console to communicate with backups on the Core. See Managing a Linux boot image.
    - If you require physical media to start up the destination Linux machine, you will need to transfer the ISO image to media. See Transferring the Live DVD ISO image to media.
    - In all cases, you will need to load the boot image into the destination server and start the server from the boot image. See Loading the Live DVD and starting the target machine.
- **Manage Partitions**. You may need to create or mount partitions before performing a BMR on a Linux machine. See Managing Linux partitions.
    - The Linux system on which you are performing a BMR must have the same partitions as the source volumes in the recovery point. You may need to create additional partitions on the target system, if required. See Creating partitions on the destination drive.
    - If you are performing a manual BMR, you must first mount partitions. See Mounting partitions from the command line. Steps to mount partitions are included in the process to perform a BMR from the command line. See Launching a bare metal restore for a Linux machine using the command line.

      If you are using auto-partitioning for BMR within the Core Console, you do not need to mount partitions. AppAssure will restore the same partitions as those included in the recovery point(s) being restored.
- **Launch a Bare Metal Restore for Linux**. Once the destination machine is started from the Live DVD boot image, you can launch the BMR. The tasks required depend on whether you will perform this from the AppAssure user interface or from the command line using the aamount utility. See Launching a bare metal restore for Linux.
    - If using the Core Console, you will need to initiate a restore from a recovery point on the Core. See Selecting a recovery point and initiating BMR.
    - If using the Core Console, you will need to map the volumes from the UI. See Mapping volumes for a bare metal restore.
    - Optionally, if restoring from the command line, you can start the screen utility to enhance your ability to scroll and see commands in the terminal console. For more information, see Starting the Screen utility.
    - If using aamount, all tasks will be performed at the command line. For more information, see Launching a bare metal restore for a Linux machine using the command line.

- **Verifying a Bare Metal Restore**. After starting the bare metal restore, you can verify and monitor your progress. See Verifying the bare metal restore from the command line.
    - You can monitor the progress of your restore. See Viewing the recovery progress.
    - Once completed, you can start the restored server. See Starting a restored target server.
    - Troubleshoot the BMR process. See Troubleshooting connections to the Universal Recovery Console and Repairing startup problems.

Prerequisites for performing a bare metal restore for a Linux machine

Before you can begin the process of performing a bare metal restore for a Linux machine, you must ensure that the following conditions and criteria exist:

- **Backups of the machine you want to restore.** You must have a functioning AppAssure Core containing recovery points of the protected server you want to restore.
- **Hardware to restore (new or old, similar or dissimilar).** The target machine must meet the installation requirements for an agent; for details, see the *Dell AppAssure Installation and Upgrade Guide*.
- **Live DVD boot image**. Obtain the Linux Live DVD ISO image, which includes a bootable version of Linux. Download it from the Dell AppAssure License Portal at https://licenseportal.com. If you have any issues downloading the Live DVD, contact Dell AppAssure support.
- **Image media and software.** If using physical media, you must have a blank CD or DVD and disk burning software, or software to create an ISO image.
- **Compatible storage drivers and network adapter drivers.** If restoring to dissimilar hardware, then you must have compatible storage drivers and network adapter drivers for the target machine, including RAID, AHCI, and chipset drivers for the target operating system, as appropriate.
- **Storage space and partitions, as appropriate.** Ensure that there is enough space on the hard drive to create destination partitions on the target machine to contain the source volumes. Any destination partition should be at least as large as the original source partition.
- **Restore path.** Identify the path for the restore, which is the path for the device file descriptor. To identify the path for the device file descriptor, use the `fdisk` command from a terminal window.

Managing a Linux boot image

A bare metal restore for Linux requires a Live DVD boot image, which you download from the Dell AppAssure License Portal. You will use this image to start the destination Linux machine. Based on the specifics of your environment you may need to transfer this image to physical media such as a CD or DVD. You must then virtually or physically load the boot image, and start the Linux server from the boot image.

NOTE: The Live DVD was previously known as the Live CD.

You can perform the following tasks:

- Downloading a boot ISO image for Linux
- Transferring the Live DVD ISO image to media
- Loading the Live DVD and starting the target machine

Managing a Linux boot image is a step in Performing a bare metal restore for Linux machines.

Downloading a boot ISO image for Linux

The first step when performing a bare metal restore (BMR) for a Linux machine is to download the Linux Live DVD ISO image from the Dell AppAssure License Portal. The Live DVD functions with all Linux file systems supported by AppAssure, and includes a bootable version of Linux, a screen utility, and the AppAssure Universal Recovery Console (URC) interface. The AppAssure Universal Recovery Console is an environment that is used to restore the system drive or the entire server directly from the AppAssure Core.

NOTE: The International Organization for Standardization (ISO) is an international body of representatives from various national organizations that sets file system standards. The ISO 9660 is a file system standard that is used for optical disk media for the exchange of data and supports various operating systems. An ISO image is the archive file or disk image, which contains data for every sector of the disk as well as the disk file system.

You must download the Live DVD ISO image that matches your version of AppAssure. The current version of Live DVD is available from the Dell AppAssure License Portal at https://licenseportal.com. If you need a different version, contact Dell AppAssure Support.

NOTE: For more information about the Dell AppAssure License Portal, see the *Dell AppAssure License Portal User Guide*.

This task is a step in Performing a bare metal restore for Linux machines. It is part of the process for Managing a Linux boot image.

Complete the steps in this procedure to download the Live DVD ISO image.

To download a Boot ISO image for Linux

1. Log into the Dell AppAssure License Portal at https://licenseportal.com.
2. Access the **Downloads** area.
3. Scroll down to Linux Based Applications and, from the Linux Live DVD section, click **Download**.
4. Save the Live DVD ISO image. If you are restoring a virtual machine, you can save it to a network location, and set the VM to start up from a CD or DVD drive associated with the ISO image.
5. If restoring from a physical machine, burn the Boot CD ISO image onto a compact disc (CD) or digital video disk (DVD) from which the target machine can be started. For more information, see Transferring the Live DVD ISO image to media.

Transferring the Live DVD ISO image to media

When you download the Linux Live DVD file, it is stored as an ISO image in the path you specified. You must be able to boot the target Linux machine from the Live DVD image.

You can burn the boot CD ISO image onto compact disc (CD) or digital video disk (DVD) media.

When you start the machine from the Live DVD, the Universal Recovery Console launches automatically.

If performing a BMR on a virtual machine, this step is not required. Simply load the ISO image in a drive and edit settings for that VM to start from that drive.

This task is a step in Performing a bare metal restore for Linux machines. It is part of the process for Managing a Linux boot image.

Loading the Live DVD and starting the target machine

After you obtain the Live DVD ISO image, you need to start the Linux machine from the newly created Live DVD.

This task is a step in Performing a bare metal restore for Linux machines. It is part of the process for Managing a Linux boot image.

To load a Live DVD and start the target machine

1. Navigate to the new server and load the Live DVD image from the appropriate location. Specify that the server will start from the Live DVD image.
2. Start the machine.

   An AppAssure splash screen displays and a terminal window opens, displaying the IP address and authentication password for the machine.

   NOTE: A new temporary password is generated each time the machine is started with the Live DVD image.

3. Write down the IP address and the authentication password displayed on the introduction screen. You will need this information later during the data recovery process to log back on to the console.

Once the target Linux machine is started with the Live DVD, this machine is ready for the user to connect to it from the Core to begin the bare metal restore process. You can perform this process using any one of two methods:

- Launching a restore from the AppAssure Core Console. For more information, see Launching a bare metal restore for Linux.
- Launching a Restore from the command Line using the aamount utility. For more information, see Launching a bare metal restore for a Linux machine using the command line.

Managing Linux partitions

When performing a BMR, the destination drive onto which you will be restoring data must have the same partitions as in the recovery point you are restoring. You may need to create partitions to meet this requirement.

You can launch the restore from the command line using the aamount utility, or you can launch the restore from the AppAssure Core Console. If restoring using the user interface, you must first mount the partitions.

You can perform the following tasks:

- Creating partitions on the destination drive
- Formatting partitions on the destination drive
- Mounting partitions from the command line

Managing Linux partitions is a step in Performing a bare metal restore for Linux machines.

Creating partitions on the destination drive

Often, when performing a BMR, the destination drive is a new volume that may consist of a single partition. The drive on the destination machine must have the same partition table as in the recovery point, including the size of the volumes. If the destination drive does not contain the same partitions, you must create them before performing the bare metal restore. Use the fdisk utility to create partitions on the destination drive equal to the partitions on the source drive.

This task is a step in Performing a bare metal restore for Linux machines. It is part of the process for Managing Linux partitions.

To create partitions on the destination drive

1 Optionally, you can start the Screen utility. Once started, it remains active until you reboot the machine.

> **NOTE:** For more information, see Starting the Screen utility.

2 From the command line, enter the following command and then press **Enter** to change privileges to run as administrator and then list existing disk partitions:

```
sudo fdisk -l
```

A list of all volumes appears.

This example assumes the volume you want to partition is /dev/sda. If your volume is different (for example, for older drives, you may see /dev/hda), change commands accordingly.

3 To create a new boot partition, enter the following command and then press **Enter**:

```
sudo fdisk /dev/sda
```

4 To create a new boot partition, enter the following command and then press **Enter**:

```
n
```

5 To create a new primary partition, enter the following command and then press **Enter**:

```
p
```

6 To specify partition number, enter the partition number and then press **Enter**. For example, to specify partition 1, type 1 and then press **Enter**.

7 To use the first sector, 2048, press **Enter**.

8 Allocate an appropriate amount to the boot partition by entering the plus sign and the allocation amount and then press **Enter**.

For example, to allocate 500 M for the boot partition, type the following and then press **Enter**:

```
+500M
```

9 To toggle a bootable flag for the boot partition (to make the partition bootable), type the following command and then press **Enter**:

```
a
```

10 To assign a bootable flag for the appropriate partition, type the number of the partition and then press **Enter**. For example, to assign a bootable flag for partition 1, type 1 and then press **Enter**.

11 To save all changes in the fdisk utility, type the following command and then press **Enter**:

```
w
```

Formatting partitions on the destination drive

After creating partitions on a new volume on the destination drive, you must format the partitions before they can be mounted. If this situation applies to you, follow this procedure to format partitions in Ext3, Ext4, or XFS formats.

This task is a step in Performing a bare metal restore for Linux machines. It is part of the process for Managing Linux partitions.

To format partitions on the destination drive

1 Optionally, you can start the Screen utility. Once started, it remains active until you reboot the machine.

> **NOTE:** For more information, see Starting the Screen utility.

2 From the command line, enter the following command and then press **Enter** to change privileges to run as administrator and then list existing disk partitions:

```
sudo fdisk -l
```

A list of all volumes appears.

This example assumes the partition you want to format is /dev/sda1. If your volume is different (for example, for older drives, you may see /dev/hda), change commands accordingly.

3 Select one of the following command based on the format you want to use for the destination partition:

- To format a partition in Ext3 format, enter the following command and then press **Enter**:

  ```
  sudo mkfs.ext3 /dev/sda1
  ```

- To format a partition in Ext4 format, enter the following command and then press **Enter**:

  ```
  sudo mkfs.ext4 /dev/sda1
  ```

- To format a partition in xfs format, enter the following command and then press **Enter**:

  ```
  sudo mkfs.xfs /dev/sda1
  ```

The selected partition is formatted accordingly.

4 Optionally, if you need to format other partitions, repeat this procedure.

Mounting partitions from the command line

If performing a BMR using the AppAssure Core Console, you must first mount the appropriate partitions on the destination machine. Perform this from the command line in the Universal Recovery Console.

This task is a step in Performing a bare metal restore for Linux machines. It is part of the process for Managing Linux partitions.

Complete the steps in this procedure to mount partitions on the Linux machine before performing a restore.

To mount partitions from the command line

1. From the command line, enter the following command and then press **Enter** to change privileges to run as administrator and then list existing disk partitions:

   ```
   sudo fdisk -l
   ```

   A list of all volumes appears.

2. Format all partitions you will need to perform the BMR to the mount directory. These must match the volumes that are in the recovery point. For example, if the volume you want to mount is called sda1, and the mount directory is mnt, then type the following command and then press **Enter**:

3. Mount all partitions you will need to perform the BMR to the mount directory. These must match the volumes that are in the recovery point. For example, if the volume you want to mount is called sda1, and the mount directory is mnt, then type the following command and then press **Enter**:

   ```
   mount /dev/sda1 /mnt
   ```

4. Repeat Step 3 as necessary until you have mounted all required volumes.

Once volumes are mounted, you can perform a restore to the destination Linux machine from the AppAssure Core Console. See Launching a bare metal restore for Linux.

Launching a bare metal restore for Linux

Before launching a bare metal restore (BMR) for a Linux machine, certain conditions are required.

To restore a recovery point saved on the Core, you must have the appropriate hardware in place. For more information, see Prerequisites for performing a bare metal restore for a Linux machine.

The BMR destination Linux machine must be started using the Live DVD boot image. For more information, see Managing a Linux boot image.

The number of volumes on the Linux machine to be restored must match the number of volumes in the recovery point. You must also decide whether to restore from the AppAssure Core Console, or from the command line using aamount. For more information, see Managing Linux partitions.

If restoring from the Core Console UI, the first step in launching a BMR is to select the appropriate recovery point, then initiate the restore to the hardware by specifying the IP address and temporary password you obtained from the Universal Recovery Console. You must then map the drives and start the restore.

To launch a BMR from the AppAssure Core Console, perform the following tasks.

- Selecting a recovery point and initiating BMR
- Mapping volumes for a bare metal restore

If restoring from the command line using the aamount utility, then you must first set appropriate privileges, mount volumes, execute aamount, obtain information about the Core from the list of machines, connect to the core, obtain a list of recovery points, select the recovery point you want to roll back onto bare metal, and launch the restore.

Optionally, you may want to start the Screen utility.

To launch a BMR from the command line, perform the following tasks.

- Starting the Screen utility
- Launching a bare metal restore for a Linux machine using the command line

This process is a step in Performing a bare metal restore for Linux machines.

Starting the Screen utility

Included on the Live DVD is Screen, a utility which is available when you boot from the Live DVD into the Universal Recovery Console. Screen allows users to manage multiple shells simultaneously over a single Secure Shell (SSH) session or console window. This allows you to perform one task in a terminal window (such as verify mounted volumes) and, while that is running, open or switch to another shell instance to perform another task (such as to run the aamount utility).

The Screen utility also has its own scroll-back buffer, which enables you to scroll the screen to view larger amounts of data, such as a list of recovery points.

NOTE: This utility is provided for convenience; use of the Screen utility is optional.

The screen utility starts on the machine booted with the Live DVD by default. However, if you have closed this application, you must start the Screen utility from the Live DVD using the procedure below.

To start the Screen utility

- If the machine was booted from the Live DVD, then in the terminal window, type **screen** and press **Enter**.

  The Screen utility starts.

Launching a bare metal restore for a Linux machine using the command line

Once the Live DVD ISO image is accessible on the machine on which you want to perform a BMR, and the number and size of volumes matches between the target machine and the recovery point you want to restore to bare metal, then you can launch a restore from the command line using the aamount utility.

If you want to perform a BMR using the AppAssure Core Console UI, see Selecting a recovery point and initiating BMR.

NOTE: When performing this procedure, do not attempt to mount recovery points to the /tmp folder, which contains the aavdisk files.

This task is a step in Performing a bare metal restore for Linux machines. It is part of the process for Launching a bare metal restore for a Linux machine using the command line.

Complete the steps in this procedure to select a recovery point on the Core to roll back to the physical or virtual BMR target machine.

To perform a bare metal restore for a Linux machine using the command line

1. To run the AppAssure aamount utility as root, type the following command and then press **Enter**:

   ```
   sudo aamount
   ```

2. To list the protected machines, type the following command and then press **Enter**:

   ```
   lm
   ```

3. When prompted, enter the connection information for the AppAssure Core as described in the following table, pressing **Enter** after each required command:

Table 132. AppAssure Core connection information

Text Box	Description	Required
AppAssure Core IP address or hostname	The IP address or hostname of the AppAssure Core.	Yes
Domain	The domain of the AppAssure Core. This is optional.	No
User	The user name for an administrative user on the Core	Yes
Password	The password used to connect the administrative user to the Core.	Yes

A list displays showing the machines protected by the AppAssure Core. It lists the machines found by line item number, the host display name or IP address, and an ID number for the machine.

4 To list the recovery points for the machine that you want to restore, type the list recovery points command using the following syntax and then press **Enter**:

```
lr <machine_line_item_number>
```

NOTE: You can also enter the machine ID number in this command instead of the line item number.

A list displays the base and incremental recovery points for that machine. This list includes:

- A line item number
- Date and time stamp
- A lettered list of volumes within the recovery point
- Location of the volume
- Size of the recovery point
- An ID number for the volume that includes a sequence number at the end, which identifies the recovery point

5 To select the recovery point for a restore, enter the following command and then press **Enter**:

```
r <recovery_point_ID_number> <path>
```

CAUTION: You must ensure that the system volume is not mounted.

NOTE: If you started the machine from the Live DVD, then the system volume is not mounted.

This command rolls back the volume image specified by the ID from the Core to the specified path. The path for the restore is the path for the device file descriptor and is not the directory to which it is mounted.

NOTE: You can also specify a line number in the command instead of the recovery point ID number to identify the recovery point. In that case, use the agent/machine line number (from the `lm` output), followed by the recovery point line number and volume letter (from the lettered list of volumes within the recovery point), followed by the path. For example:

```
r <machine_line_item_number> <base_image_recovery_point_line_number>
<volume_letter> <path>
```

For example, type:

```
r 1 24 a /dev/sda1
```

In this command, *<path>* is the file descriptor for the actual volume.

6 When prompted to proceed, enter **y** for Yes and then press **Enter**.

After the restore begins, a series of messages will display that notify you of the restore completion status.

> **NOTE:** If you receive an exception message, the details regarding that exception can be found in the aamount.log file. The aamount.log file is located in /var/log/appassure.

7 Upon a successful restore, exit aamount by typing `exit` and then press **Enter**.

8 Your next step is to verify the restore. For more information, see Verifying the bare metal restore from the command line.

Verifying the bare metal restore from the command line

Dell recommends performing the following steps to verify a bare metal restore completed from the command line.

- Performing a file system check on the restored volume
- Using the command line to make a restored Linux machine bootable

This task is a step in Performing a bare metal restore for Linux machines.

Performing a file system check on the restored volume

Once you execute a bare metal restore from the command line, you should perform a file system check on the restored volume to ensure the data restored from the recovery point was not corrupted.

This task is a step in Performing a bare metal restore for Linux machines. It is part of the process for Verifying the bare metal restore from the command line.

Perform the task below to perform a file system check on the restored volume.

To perform a file system check on the restored volume

1 From the command line in the Universal Recovery Console of the Linux machine you have restored, to verify whether the appropriate partitions are mounted, type the following command and then press **Enter**:

```
df
```

2 If the restored volume is not mounted, then skip to Step 3. If the restored volume is mounted, unmount it by typing the following command and then pressing **Enter**:

```
umount <volume>
```

3 Run a file system check on the restored volumes by typing the following command and then press Enter:

```
fsck -f <volume>
```

If the fsck returns clean, the file system is verified.

4 Mount the appropriate volumes once again by typing the following command in format `mount <volume> <folder>`, and then press **Enter**.

For example, if the volume path is prod/sda1 and the folder you want to mount to is mnt, then type the following and then press **Enter**:

```
mount /dev/sda1 /mnt
```

Using the command line to make a restored Linux machine bootable

Once you complete a clean file system check on the restored volume, you must create bootable partitions.

GNU Grand Unified Bootloader (GRUB) is a boot loader that allows administrators to configure which operating system or specific kernel configuration is used to start the system. After a BMR, the configuration file for GRUB must be modified so that the machine uses the appropriate universally unique identifier (UUID) for the root volume. Before this step you must mount the root and boot volumes, and check the UUIDs for each. This ensures that you can boot from the partition.

NOTE: This procedure applies to Linux machines that use GRUB1 or GRUB2. When using this procedure, ensure that the boot partition is healthy and protected.

CAUTION: When you boot a restored Linux machine for the first time after a BMR, AppAssure takes a base image of the restored machine. Depending on the amount of data on the machine, this process takes more time than taking an incremental snapshot. For more information about base images and incremental snapshots, see Understanding protection schedules.

This task is a step in Performing a bare metal restore for Linux machines. It is part of the process for Verifying the bare metal restore from the command line.

Perform the task below to create bootable partitions using the command line.

To use the command line to make a restored Linux machine bootable

1 You must mount the root volume first and then the boot volume. Mount each restored volume by using the following commands:

   a To mount the root volume, type the following command and then press **Enter**:

   ```
   mount /<restored volume[root]> /mnt
   ```

   For example, if /dev/sda2 is the root volume, then type **mount /dev/sda2 /mnt** and then press **Enter**.

   b To mount the boot volume, type the following command and then press **Enter**:

   ```
   mount /<restored volume[boot]> /mnt/boot
   ```

   For example, if /dev/sda1 is the boot volume, then type **mount /dev/sda1 /mnt/boot** and then press **Enter**.

   NOTE: Some system configurations may include the boot directory as part of the root volume.

2 If the volume size is increasing — that is, if the destination volume on the new Linux machine is larger than the volume was in the recovery point — then you must delete any existing bitmap data files.

3 Obtain the Universally Unique Identifier (UUID) of the new volumes by using the `blkid` command. Type the following and then press **Enter**:

   ```
   blkid [volume]
   ```

   NOTE: You can also use the `ls -l /dev/disk/by-uuid` command.

4 Obtain the old UUID of the partition or partitions from the mounted recovery points `/etc/fstab` file and compare it to the UUIDs for the root (for Ubuntu and CentOS), boot (for CentOS and RHEL), or data partitions by typing the following command and then press **Enter**:

   ```
   less /mnt/etc/fstab
   ```

5 Obtain the old UUID of the partition or partitions from the mounted recovery points `/etc/mtab` file and compare it to the UUIDs for the root (for Ubuntu and CentOS), boot (for CentOS and RHEL), and data partitions by typing the following command and then press **Enter**:

```
less /mnt/etc/mtab
```

6 If performing a BMR on a brand new disk on the destination machine, comment out the swap partition in fstab in your root volume.

7 Modifying fstab and mtab paths should occur on the restored volume, not the Live DVD. There is no need to modify paths on the Live DVD. Prepare for the installation of Grand Unified Bootloader (GRUB) by typing the following commands. Following each command, press **Enter**:

```
mount --bind /dev /mnt/dev
```

```
mount --bind /proc /mnt/proc
```

```
mount --bind /sys /mnt/sys
```

8 Locate the `grub.conf` file in your mounted volume, and open it using a text editor.

The location of grub.conf differs depending on your OS version and the version of GRUB installed. The most likely locations include **<root path>/boot/grub/grub.conf**, **<root path>/boot/grub/grub.cfg** or **<root path>/etc/grub.conf**.

9 In grub.conf, locate all lines containing "root=<root device uuid>" and replace it with the correct UUID for the root volume. You can also use the root device path. As in the examples above, if the root device path is /dev/sda2, then change all instances to root=/dev/sda2.

10 Do one of the following:

- If you are not using LVM, remove all "rd_LVM_LV=" entries in the grub.conf file, save the file, and exit the text editor.
- If you are using LVM, enter the device name. For example:

  ```
  root=/dev/mapper/vg_TEST-root
  rd_LVM_LV=vg_TEST/sqp_1rd_LVM_LV=vg_TEST/root
  ```

11 Change root directory by typing the following command and then press **Enter**:

```
chroot /mnt /bin/bash
```

12 If using SLES, install GRUB by typing the following commands, pressing **Enter** after each:

```
grub-install --recheck /dev/sda
```

```
grub-install /dev/sda
```

13 For Linux distributions other than SLES, install GRUB by typing the following command and then press **Enter**:

```
grub-install/dev/sda
```

> **NOTE:** If installing on SUSE, when installing GRUB, no parameters are required. For example, the command to install GRUB on SUSE is simply grub-install and then press **Enter**.

14 After you complete installation, run one of the following updates:

- For SLES:

```
grub-install.unsupported --recheck /dev/sda
grub-install.unsupported /dev/sda
update-grub
```

NOTE: If the `update-grub` command does not exist on your Linux distribution, omit this option.

- For other distributions:

```
grub-install /dev/sda
update-grub
```

NOTE: If the `update-grub` command does not exist on your Linux distribution, omit this option.

15 Remove the Live DVD disk from the CD-ROM or DVD drive and restart the Linux machine.

19

Understanding the Local Mount Utility

This chapter describes how to download, install, and use the Local Mount Utility (LMU) to mount an explore recovery points from a machine that does not host the AppAssure Core. It includes the following topics:

- Downloading the Local Mount Utility
- Installing the Local Mount Utility
- Using the Local Mount Utility
- Using AppAssure Core and protected machine options

The LMU is a downloadable application that lets you mount a recovery point on a remote AppAssure Core from any machine. The light-weight utility includes the *aavdisk* and *aavstor* drivers, but it does not run as a service. When you install the utility, by default, it is installed in the directory C:\Program Files\AppRecovery\Local Mount Utility and a shortcut appears on the machine's desktop.

While the utility was designed for remote access to an AppAssure Core machine, you also can install the LMU on the same machine as an AppAssure Core. When it runs on a Core, the application recognizes and displays all mounts from that Core, including mounts performed through the AppAssure Core Console. Likewise, mounts performed on the LMU also appear in the console.

When the LMU is installed on a Core machine, it provides access to Mailbox Restore, the Dell AppAssure application used to restore Microsoft Exchange data stores and items. For more information about Mailbox Restore, see the *Dell AppAssure Mailbox Restore User for Microsoft Exchange User Guide*.

Downloading the Local Mount Utility

There are two ways to download the Local Mount Utility. You can download the web installer version of the software directly from the AppAssure Core Console, which is practical if you are installing the LMU on the Core. You can also download either 32-bit or 64-bit executable installer files for the LMU from the Dell AppAssure License Portal.

This section includes the following topics:

- Downloading the LMU from the AppAssure Core Console
- Downloading the LMU from the Dell AppAssure License Portal

Downloading the LMU from the AppAssure Core Console

Complete the following steps to download the Local Mount Utility from the AppAssure Core Console.

To download the Local Mount Utility from the AppAssure Core Console

1. From the machine on which you want to install the LMU, access the AppAssure Core Console by entering the console URL into your browser and logging on with your user name and password.
2. From the AppAssure Core Console, click the Tools tab.
3. On the Tools tab, click **Downloads**.
4. Under Local Mount Utility, click the **Download web installer** link.
5. From the Opening LocalMountUtility-Web.exe window, click **Save File**.

   The file saves to the local Downloads folder. In some browsers, the folder automatically opens or a pop-up message gives you the option to run the installation.

Downloading the LMU from the Dell AppAssure License Portal

If you have already registered your AppAssure Core software in the Dell AppAssure License Portal, you can log into the License Portal at https://licenseportal.com and download the LMU software to the machine on which you want to install it.

NOTE: For more information about the Dell AppAssure License Portal, including obtaining a license key or registering and creating a License Portal account, see the *Dell AppAssure License Portal User Guide*.

Complete the following steps to download the Local Mount Utility from the Dell AppAssure License Portal.

To download the Local Mount Utility from the Dell AppAssure License Portal

1. From the machine on which you want to install the LMU, log into the Dell AppAssure License Portal at https://licenseportal.com.
2. Access the **Downloads** area.
3. Scroll down the list of Windows Based Applications. From the Local Mount Utility section, based on the architecture of the machine on which you are installing the utility (64-bit systems or 32-bit systems on the x86 architecture), click **Download**.
4. Save the executable installer in an appropriate location (for example, Downloads).

Installing the Local Mount Utility

The Local Mount Utility can be installed using the web installer, or an executable installation file specific to the architecture of the machine on which it is being installed. The process is the same regardless of which installer you use. Follow the procedure below to install the Local Amount Utility.

To install the Local Mount Utility

1. From the machine on which you want to install the LMU, locate the executable installation file.

   If you downloaded the installer from the Core, the install file uses the web installer, which is named LocalMountUtility-Web.exe. If you downloaded from the License Portal, the installer name includes the architecture of your operating system, for example LocalMountUtility-X32.exe or LocalMountUtility-X64.exe.

2 Launch the installer program using one of the following methods.

- From the Downloads folder, right-click on the executable file and click **Open**.
- Or, if the web browser opened a message after downloading with the option to run the executable file, click **Run**.

Depending on the configuration of your machine, the User Account Control window or the Open File - Security Warning window could appear.

3 If the User Account Control window or the Open File - Security Warning window appears, click **Yes** or **Run** to let the program make changes to the machine.

4 In the Setup dialog box, from the language field, select the appropriate language and then click **OK**.

The installer prepares the installation.

5 Choose from one of the following:

- If this machine has an earlier version of the AppAssure Local Mount Utility software installed, you will see a message asking if you want to upgrade to the current version.

  a Click **Yes**.

  The AppAssure Local Mount Utility Installation Wizard launches, and the Progress page of the wizard appears. The application downloads to the destination folder, with progress displayed in the progress bar. When finished, the wizard automatically advances to the Completed page.

  b Skip to Step 10.

- If this is the first time the AppAssure LMU software is being installed on this machine, proceed to Step 6.

6 On the AppAssure Local Mount Utility Installation Wizard Welcome page, click **Next** to continue to the License Agreement page.

7 On the License Agreement page, select **I accept the terms in the license agreement**, and then click **Next**.

8 On the Prerequisites page, install any necessary prerequisites and then click **Next**.

9 On the Installation Options page, complete the following tasks:

a Select a destination folder for the LMU by clicking the **folder icon**.

> **NOTE:** The default destination folder is
>
> `C:\Program Files\AppRecovery\LocalMountUtility`.

b Optionally, to also install Mailbox Restore, select **Mailbox Restore**.

> **CAUTION: If you already installed Mailbox Restore on this machine with your installation of the AppAssure Core, clear this option so that you do not install it a second time.**

c Select whether to **Allow Local Mount Utility to automatically send diagnostic and usage information to Dell.**

d Click **Install**.

The Progress page of the wizard appears, and the application downloads to the destination folder, with progress displayed in the progress bar. When finished, the wizard automatically advances to the Completed page.

10 On the Completed page, click **Finish** to close the wizard.

Using the Local Mount Utility

The LMU lets you mount, explore and dismount the recovery points of any Core you add to the utility. There is no limit to the number of cores you can add. For more information about using the LMU, see the following procedures:

- Adding a Core machine to the Local Mount Utility
- Mounting a recovery point using the Local Mount Utility
- Exploring a mounted recovery point using the Local Mount Utility
- Dismounting a recovery point using the Local Mount Utility

Adding a Core machine to the Local Mount Utility

To mount a recovery point, you must add a Core machine to the LMU. There is no limit as to how many Cores you can add.

Complete the following procedure to set up the LMU by adding a core.

To add a core to the Local Mount Utility

1 From the machine on which the LMU is installed, launch the LMU by double-clicking the desktop icon.

2 If the User Account Control window appears, click **Yes** to let the program to make changes to the machine.

3 In the upper-left corner of the AppAssure Local Mount Utility window, click **Add Core**.

4 In the Add Core dialog box, enter the requested credentials described in the following table.

Table 133. AppAssure Core credentials

Option	Description
Host name	The name of the Core from which you want to mount recovery points. **NOTE:** If installing the LMU on a core, the LMU automatically adds the localhost machine.
Port	The port number used to communicate with the Core. The default port number is 8006.
Use my Windows user credentials	Select this option if the credentials you use to access the Core are the same as your Windows credentials.
Use specific credentials	Select this option if the credentials you use to access the Core are different from your Windows credentials.
User name	The user name used to access the Core machine. **NOTE:** This option is only available if you choose to use specific credentials.
Password	The password used to access the Core machine. **NOTE:** This option is only available if you choose to use specific credentials.

5 Click **Connect**.

6 If adding multiple cores, repeat Step 3 through Step 5 as necessary.

Mounting a recovery point using the Local Mount Utility

Before mounting a recovery point, the local mount utility (LMU) must connect to the Core on which the recovery point is stored. As described in the procedure Adding a Core machine to the Local Mount Utility, the number of cores that can be added to the LMU is unlimited; however, the application can connect to only one core at a time. For example, if you mount a recovery point of a machine protected by one core and then mount a recovery point of another machine protected by a different core, the LMU automatically disconnects from the first core to establish a connection with the second core.

Complete the following procedure to mount a recovery point on a remote core using the LMU.

To mount a recovery point using the Local Mount Utility

1 From the machine on which the LMU is installed, launch the LMU by double-clicking the desktop icon.

2 From the main AppAssure Local Mount Utility window, expand the Core in the navigation tree to reveal the protected machines.

3 From the navigation tree, select the protected machine from which you want to mount a recovery point.

The recovery points appear in the main frame.

4 Expand the recovery point you want to mount to reveal individual disk volumes or databases.

5 Right-click the recovery point you want to mount, and select one of the following options:

- Mount
- Mount writable
- Mount with previous writes
- Advanced mount

a If you selected Advanced Mount, then from the Advanced Mount window, complete the options described in the following table.

Table 134. Advanced Mount options

Option	Description
Mount point path	Click the **Browse** button to select a path for the recovery points other than the default mount point path.
Mount type	Select one of the following options: • Mount read-only • Mount writable • Mount read-only with previous writes

b Click **Mount**.

The LMU automatically opens the folder containing the mounted recovery point.

NOTE: If you select a recovery point that is already mounted, the Mounting dialog will prompt whether to dismount the recovery point.

Exploring a mounted recovery point using the Local Mount Utility

Complete the following procedure to explore a recovery point that has remained mounted from a previous session.

NOTE: This procedure is not necessary if you are exploring a recovery point immediately after mounting it, as the folder containing the recovery point automatically opens upon completion of the mounting procedure.

To explore a mounted recovery point using the Local Mount Utility

1. From the machine on which the LMU is installed, launch the LMU by double-clicking the desktop icon.
2. From the main Local Mount Recovery screen, click **Active mounts**.

   The Active Mounts window opens and displays all mounted recovery points.
3. Click **Explore** beside the recovery point from which you want to recover to open the folder of deduplicated volumes.

Dismounting a recovery point using the Local Mount Utility

Complete the following procedure to dismount a recovery point on a remote core using the LMU.

To dismount a recovery point using the Local Mount Utility

1. From the machine on which the LMU is installed, double-click the Local Mount Utility desktop icon to launch the program.
2. From the main Local Mount Recovery screen, click **Active mounts**.

   The Active Mounts window opens and displays all mounted recovery points.
3. Do one of the following:
   - To dismount one recovery point, select a recovery point you want to dismount, and then click **Dismount**.
   - To dismount all mounted recovery points, click **Dismount all**, and then click **Yes** in the Dismount All dialog box to confirm.
4. To close the Active mounts window, click the X in the upper-right corner.
5. To minimize the LMU application, click the X in the upper-right corner of the Local Mount Utility window.
6. To close the LMU application, right-click the AppAssure Local Mount Utility icon in the LMU tray menu, and select **Exit**.

Using the Local Mount Utility tray menu

The LMU tray menu is located in your desktop task bar. Right-click the icon to reveal the options described in the following table:

Table 135. Tray menu options

Option	Description
Browse Recovery Points	Opens the LMU main window.
Active Mounts	Opens the Active Mounts dialog box on top of the LMU main window.
Options	Opens the Options dialog box on top of the LMU main window. From the Options dialog box, you can change the Default mount point directory and the default Core credentials for the LMU user interface.
About	Reveals the Local Mount Utility licensing information.
Exit	Closes the LMU application. **NOTE:** Clicking the X in the upper corner of the main window of the LMU minimizes the application to the tray; it does not exit the application.

Using AppAssure Core and protected machine options

By right-clicking the AppAssure Core or a protected machine in the main LMU screen, you can access certain options, as described in the following topics:

- Accessing localhost options
- Accessing remote core optionsAccessing protected machine options

Accessing localhost options

Complete the step in this procedure to access localhost options.

To access localhost options

- Right-click the AppAssure Core or protected machine, and then click **Reconnect to core**.

  Information from the Core is updated and refreshed; for example, recently added protected machines.

Accessing remote core options

Complete the steps in this procedure to access remote core options.

To access remote core options

- Right-click the AppAssure Core or protected machine, and then select one of the remote core options as described in the following table.

  Table 136. Remote core options

Option	Description
Reconnect to core	Refreshes and updates information from the Core, such as recently added protected machines.
Remove core	Deletes the Core from the Local Mount Utility.
Edit core	Opens the Edit Core window, where you can change the host name, port, and credentials.

Accessing protected machine options

Complete the steps in this procedure to access protected machine options.

To access protected machine options

- Right-click the AppAssure protected machine, and then click **Refresh recovery points**.

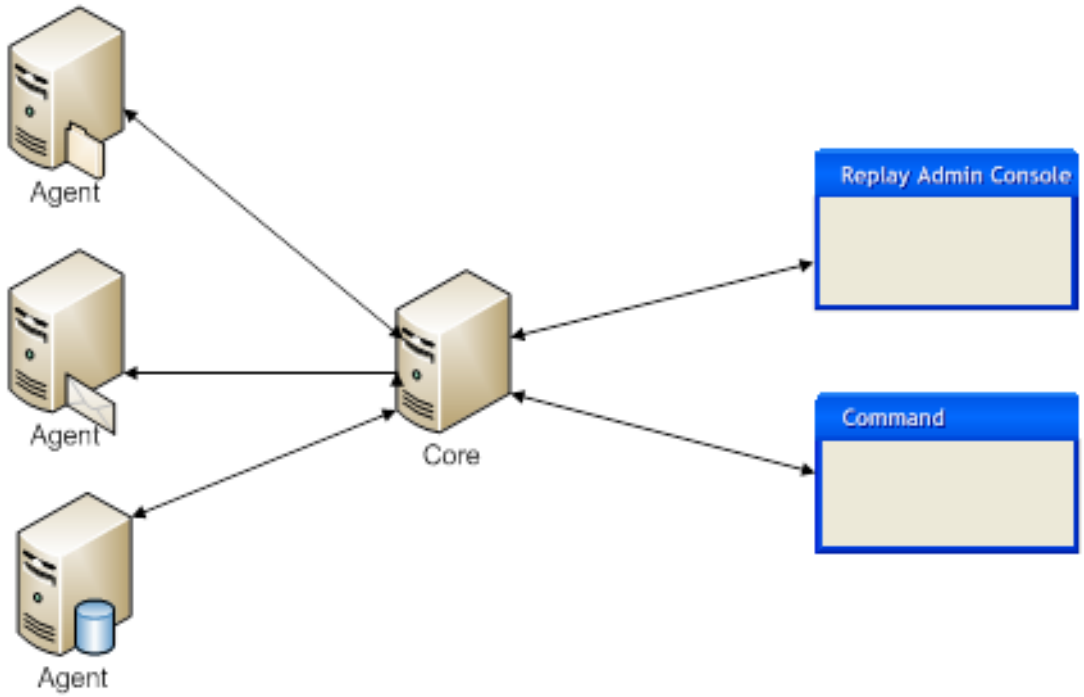
  The list of recovery points for the selected protected machine updates.

A

Understanding the AppAssure Command Line Management Utility

The AppAssure backup and disaster recovery product consists of several software components, including the AppAssure Agent, the AppAssure Core, and the AppAssure Command Line Management utility. The AppAssure Agent is responsible for volume snapshots and fast transfer of the data to the Core. The AppAssure Core, in turn, stores the snapshots along with a wide variety of enhanced features, such as bare metal restore (BMR) to dissimilar hardware, Virtual Standby, and replication, among others. To provide third-party access to manage system functionality, AppAssure includes a command line tool called AppAssure Command Line Management utility (aacmd). It permits scripting of the AppAssure Core management functions.

Figure 9.



AppAssure Command Line Management is a Windows command line utility that lets users interact with the AppAssure Core server. It offers some of the same functions that are provided by the GUI element in the AppAssure Core Console. For example, AppAssure Command Line Management utility can mount recovery points or force a snapshot.

The AppAssure Command Line Management utility is embedded in every installation of the AppAssure Core. To open the AppAssure Command Line Management utility, go to the location of the AppAssure installed folders and double-click the `aacmd.exe` file.

In Command Line Mode, action flags can be passed to the AppAssure Command Line Management utility through a selection of command options and qualifiers to perform limited management functions.

This section includes the following topics:

- Commands
- Localization

Commands

This section describes the commands and options available for the AppAssure Command Line Management utility. The following commands are available for use:

- Archive
- CancelActiveJobs
- CreateRepository
- Dismount
- Force
- ForceAttach
- ForceChecksum
- ForceLogTruncation
- ForceMount
- ForceReplication
- ForceRollup
- Help
- List
- Mount
- Pause [snapshot | vmexport | replication]
- Protect
- ProtectCluster
- RemoveAgent
- RemovePoints
- RestoreArchive
- Resume [snapshot | vmexport | replication]
- StartExport
- UpdateRepository
- Version

Archive

Businesses often use long-term storage to archive both compliant and non-compliant data. The archive feature in AppAssure supports extended retention for compliant and non-compliant data. The administrator can save an archive on the local storage or network location by specifying the `-path` parameter and credentials.

Usage

The usage for the command is as follows:

```
/archive -core [host name] -user [user name] -password [password] -all | -
protectedserver [name | IP address] -path [location] -startdate [time string] -
enddate [time string] -archiveusername [name] -archivepassword [password] -comment
[text]
```

Command Options

The following table describes the options available for the `Archive` command:

Table 137. Archive command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-all`	Archive all recovery points for all protected machines on the Core.
`-protectedserver`	Protected machine with recovery points to be archived. You can specify several machine names enclosed in double quotes and separated by spaces.
`-path`	Path where archived data should be placed; for example: d:\work\archive or network path \\servername\sharename.
`-startdate`	Start date for selecting recovery points by creation date. Value must be enclosed in double quotes.
`-enddate`	Optional. End date for selecting recovery points by creation date. Value must be enclosed in double quotes. The current time system is used by default.
`-archiveusername`	Optional. User name for the remote machine. Required for network path only.
`-archivepassword`	Optional. Password to the remote machine. Required for network path only.
`-comment`	Optional. Comment text must be enclosed in double quotes; for example: `-comment "comment goes here..."`.

Examples:

Archive all recovery points with creation dates starting from 04/30/2012 02:55 PM for all machines on the Core:

```
>aacmd /archive -core 10.10.10.10 -user administrator -password 23WE@#$sdd -path
d:\work\archive -startdate "04/30/2012 02:55 PM" -all
```

Archive recovery points that fall within a date range for two protected machines:

```
>aacmd /archive -core 10.10.10.10 -user administrator -password 23WE@#$sdd -
protectedserver "10.20.30.40" "20.20.10.1" -path d:\work\archive -startdate
"04/30/2012 02:55 PM" -enddate "05/31/2012 11:00 AM"
```

CancelActiveJobs

Use the `CancelActiveJobs` command to cancel the execution of all in-progress jobs of a specific type, such as transfer or replication.

Usage

The usage for the command is as follows:

```
/cancelactivejobs -core [host name] -user [user name] -password [password] -jobtype [job type filter]]
```

Command Options

The following table describes the options available for the `CancelActiveJobs` command:

Table 138. CancelActiveJobs command options

Option	Description
`-?`	Display help on the command.
`-core`	Optional. Remote core host machine IP address (with an optional port number). By default, the connection is made to the core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote core host machine. If you specify a password, you must also provide a user name. If none is provided, the logged-in user's credentials are used.
`-protectedserver`	Determines the protected machine on which the jobs should be canceled.
`-all`	Select and cancel events of specified type for all protected servers.
`-jobtype`	Optional. Specifies job type filter. Available values are: • 'transfer' (data transfer) • 'repository' (repository maintenance) • 'replication' (local and remote replications) • 'backup' (backup and restore) • 'bootcdbuilder' (create boot CDs) • 'diagnostics' (upload logs) • 'exchange' (Exchange Server files check) • 'export' (recovery point export) • 'pushinstall' (deploy agents) • 'rollback' (recovery point rollbacks) • 'rollup' (recovery point rollups) • 'sqlattach' (agent attachability checks) • 'mount' (mount repository) By default, all jobs of the specified type are canceled.

Example:

Cancel all transfer jobs on Core 10.10.10.10:

```
>aacmd /cancelactivejobs -core 10.10.10.10:8006 -user administrator -password 23WE@#$sdd -jobtype transfer
```

CreateRepository

Use the `CreateRepository` command to create a new repository on a local machine as well as on a share location.

Usage

The usage for the command when creating a repository on a local location is as follows:

```
/createrepository -name [repository name] -size [size allocated for repository] -datapath [data path of repository] -metadatapath [metadata path of repository] -core [host name] -user [user name] -password [password]
```

The usage for the command when creating a repository on a share location is as follows:

```
/createrepository -name [repository name] -size [size allocated for repository] -uncpath [path for data and metadata] -shareusername [user name for share location] -sharepassword [password for share user name] -concurrentoperations [number of operations to occur at one time\ -core [host name] -user [user name] -password [password]
```

Command Options

The following table describes the options available for the `CreateRepository` command:

Table 139. CreateRepository command options

Option	Description
`-?`	Display help on the command.
`-core`	Optional. Remote core host machine IP address (with an optional port number). By default, the connection is made to the core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-protectedserver`	Determines the protected machine on which the jobs should be canceled.
`-all`	Select and cancel events of specified type for all protected servers.
`-name`	Repository name.
`-size`	Size of repository storage location. Available units are b, Kb, Mb, Gb, Tb, and Pb.
`-datapath`	For local location only. Determines data path of repository storage location.
`-metadatapath`	For local location only. Determines metadata path of repository storage location.
`-uncpath`	For share location only. Determines data and metadata paths of repository storage location.
`-shareusername`	For share location only. Determines the user name to the share location.
`-sharepassword`	For share location only. Determines password to share location.
`-comment`	Optional. Description of repository.
`-concurrentoperations`	Optional. Maximum number of operations that can be pending at one time. Value by default: 64.

Examples:

Create repository at local location:

```
>aacmd /createrepository -name "Repository 1" -size 200 Gb -datapath d:\repository -
metadatapath d:\repository -core 10.10.10.10:8006 -user administrator -password
23WE@#$sdd
```

Create repository at share location:

```
>aacmd /createrepository -name "Repository 1" -size 200 Gb -uncpath
\\share\repository -shareusername login -sharepassword pass123 -comment "First
repository." -concurrentoperations 8 -core 10.10.10.10:8006 -user administrator
-password 23WE@#$sdd
```

Dismount

Use the `Dismount` command to dismount a mounted recovery point specified by the `Path` command, dismount points for the selected agent by the `-protectedserver` parameter, or dismount all mounted recovery points (`-all`).

Usage

The usage for the command is as follows:

```
/dis[mount] -core [host name] -user [user name] -password [password] [-all |
-protectedserver [name | IP address] | -path [location]
```

Command Options

The following table describes the options available for the `Dismount` command:

Table 140. Dismount command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-all`	Dismount all mounted recovery points.
`-protectedserver`	Dismount all mounted recovery points for current agent.
`-path`	Dismount selected mount point.

Example:

Dismount a recovery point that was mounted to folder c:\mountedrecoverypoint:

```
>aacmd /dismount -core 10.10.10.10 -user administrator -password 23WE@#$sdd -path
c:\mountedRecoveryPoint
```

Force

The `Force` command forces a snapshot of a specified protected server. Forcing a snapshot lets you force a data transfer for the current protected machine. When you force a snapshot, the transfer will start immediately or will be added to the queue. Only the data that has changed from a previous recovery point will be transferred. If there is no previous recovery point, all data on the protected volumes will be transferred.

Usage

The usage for the command is as follows:

```
/force [snapshot] default | [base] [-all | -protectedserver [name | IP address]]
-core [host name] -user [user name] -password [password]
```

Command Options

The following table describes the options available for the `Force` command:

Table 141. Force command options

Option	Description
`-?`	Display this help message.
`-force`	Optional. Type of snapshot to create. Available values: 'snapshot' (incremental snapshot) and 'base' (base image snapshot). By default, an incremental snapshot is performed.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-all`	Force snapshots for all machines on the core.
`-protectedserver`	Force a snapshot for a specific protected machine.

Example:

Force a snapshot for all machines on the Core:

```
>aacmd /force snapshot -core 10.10.10.10 -user administrator -password 23WE@#$sdd -
all
```

ForceAttach

The `ForceAttach` command lets you force a SQL database files attachability check. When you force an attachability check, the transfer starts immediately.

Usage

The usage for the command is as follows:

```
/forceattach -core [host name] -user [user name] -password [password]
-protectedserver [name | IP address] -rpn [number | numbers] | -time [time string]
```

Command Options

The following table describes the options available for the `ForceAttach` command:

Table 142. ForceAttach command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-protectedserver`	Optional. Protected machine against which to perform the attachability check.
`-rpn`	Optional. The sequential number of a recovery point against which to perform checks (run command /list rps to obtain the numbers). To perform checks against multiple recovery points with a single command, you can specify several numbers separated by spaces.
`-time`	Optional. Select a recovery point by its creation time. You must specify the exact time in the format “mm/dd/yyyy hh:mm tt” (for example, “2/24/2012 09:00 AM”). Keep in mind to specify the date and time values of the time zone set on your PC.

Example:

Perform attachability checks for recovery points with numbers 5 and 7:

```
>aacmd /forceattach -core 10.10.10.10 -user administrator -password 23WE@#$sdd -
protectedserver 10.10.5.22 -rpn 5 7
```

ForceChecksum

The `ForceChecksum` command lets you force an integrity check of any Exchange Message Databases (MDBs) present on the specified recovery point or points. When you force a checksum check, the command begins immediately.

Usage

The usage for the command is as follows:

```
/forcechecksum -core [host name] -user [user name] -password [password] -
protectedserver [name | IP address] -rpn [number | numbers] -time [time string]
```

Command Options

The following table describes the options available for the `ForceChecksum` command:

Table 143. ForceChecksum command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.

Table 143. ForceChecksum command options

Option	Description
-user	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
-protectedserver	Optional. Protected machine against which to perform the checksum check.
-rpn	Optional. The sequential number of a recovery point against which to perform checks (run command /list rps to obtain the numbers). To perform checks against multiple recovery points with a single command, you can specify several numbers separated by spaces.
-time	Optional. Select a recovery point by its creation time. You must specify the exact time in the format “mm/dd/yyyy hh:mm tt” (for example, “2/24/2012 09:00 AM”). Keep in mind to specify the date and time values of the time zone set on your PC.

Example:

Perform a checksum check for recovery points with numbers 5 and 7:

```
>aacmd /forcechecksum -core 10.10.10.10 -user administrator -password 23WE@#$sdd -protectedserver 10.10.5.22 -rpn 5 7
```

ForceLogTruncation

Forcing log truncation lets you perform this job one time, on-demand. It immediately truncates the logs for the specified SQL Server agent machine.

Usage

The usage for the command is as follows:

```
/[forcelogtruncation | flt] -core [host name] -user [user name] -password [password] -protectedserver [name | IP address]
```

Command Options

The following table describes the options available for the ForceLogTruncation command:

Table 144. ForceLogTruncation command options

Option	Description
-?	Display this help message.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
-protectedserver	Optional. Protected machine against which to perform log file truncation.

Example:

Force log truncation for a protected server:

```
>aacmd /forcelogtruncation -core 10.10.10.10 -user administrator -password
23WE@#$sdd -protectedserver 10.10.20.20
```

ForceMount

Use the `ForceMount` command to conduct an one-time recovery point mountability check. This determines whether or not the specified recovery point or recovery points can be mounted and used to restore backed up data. You must list either one or more specific recovery points on which to conduct the check, or a time range during which the recovery points were created.

Usage

The usage for the command is as follows:

```
/forcemount -core [host name] -user [user name] -password [password]
-protectedserver [name | IP address] -rpn [number | numbers] | -time [time string]
```

Command Options

The following table describes the options available for the `ForceMount` command:

Table 145. ForceMount command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-protectedserver`	Optional. Protected machine against which to perform log file truncation.
`-rpn`	Optional. The sequential number of a recovery point against which to perform checks (run command /list rps to obtain the numbers). To perform checks against multiple recovery points with a single command, you can specify several numbers separated by spaces.
`-time`	Optional. Select a recovery point by its creation time. You must specify the exact time in the format “mm/dd/yyyy hh:mm tt” (for example, “2/24/2012 09:00 AM”). Keep in mind to specify the date and time values of the time zone set on your PC.

Example:

Perform mountability checks for recovery points with numbers 5 and 7:

```
>aacmd /forcemount -core 10.10.10.10 -user administrator -password 23WE@#$sdd -
protectedserver 10.10.20.20 -rpn 5 7
```

ForceReplication

Use the `ForceReplication` command to force a one-time transfer of replicated data from the source core to the target core. You can replicate one specific protected server or replicate all protected servers. The protected servers must be already configured for replication.

Usage

The usage for the command is as follows:

```
/[forcereplication |frep] -core [host name] -user [user name] -password [password] -targetcore [host name] -all | -protectedserver [name | IP address]
```

Command Options

The following table describes the options available for the `ForceReplication` command:

Table 146. ForceReplication command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used
`-targetcore`	Host name of the target core against which replication should be forced.
`-protectedserver`	Optional. Protected machine against which to perform log file truncation.
`-all`	Force replication for all machines being replicated to the target core.

Example:

Force replication for a protected server on a specific target core:

```
>aacmd /forcereplication -target core 10.10.10.10 -protectedserver 10.20.30.40
```

ForceRollup

Use the `ForceRollup` command to force the rollup of recovery points on a protected machine.

Usage

The usage for the command is as follows:

```
/[forcerollup | fro] -core [host name] -user [user name] -password [password] -protectedserver [name | IP address]
```

Command Options

The following table describes the options available for the `ForceRollup` command:

Table 147. ForceRollup command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used
`-protectedserver`	Optional. Protected machine against which to perform log file truncation.

Example:

Force rollup for agent 10.10.10.1 on the Core:

```
>aacmd /forcerollup -core 10.10.10.10 - user administrator -password 23WE@#$sdd -
protectedserver 10.10.10.1
```

Help

The `Help` command displays a list of the available commands and their definitions. It also provides copyright and version details.

Usage

The usage for the command is as follows:

```
/help
```

Example:

Request Command Line help:

```
>aacmd /help
```

List

The `List` command returns information about all recovery points, active jobs, completed jobs, failed jobs, invalid (failed) recovery points, valid (passed) recovery points, mounts, protected servers, volumes, virtualized servers, unprotected volumes, clusters, protection groups, SQL databases, Exchange databases, replicated servers, and repositories for the specified agent or list of servers currently protected by the Core. The most recent records return by default. You can list all records or specify how many records display by using a number parameter. This parameter should contain the letter ***l*** for the latest recovery points and ***f*** for the first recovery point. Each recovery point has its own number, which the administrator can use for mounting.

Usage

The usage for the command is as follows:

```
/list [rps | passed | failed | mounts | volumes | protectedservers | activejobs |
completed jobs | failedjobs | virtualizedservers | unprotectedvolumes | clusters |
protectiongroups | sqldatabases | exchangemailstores | replicatedservers |
repositories] -protectedserver [name | IP address] -core [host name] -user [user
name] -password [password] -number [all | l<number> | f<number> | <number>] -jobtype
```

Command Options

The following table describes the options available for the `List` command:

Table 148. List command options

Option	Description
`-?`	Display this help message.
`-list`	Select one of the following options: • all recovery points ('rps') • valid recovery points ('passed') • invalid recovery points ('failed') • mounts ('mounts') • protected volumes ('volumes') • unprotected volumes ('unprotectedvolumes') • protected machines ('protectedservers') • active jobs ('activejobs') • failed jobs ('failedjobs') • completed jobs ('completedjobs') • virtualized servers ('virtualizedservers') • clusters ('clusters') • protection groups ('protectiongroups') • SQL Server databases ('sqldatabases') • MS Exchange databases ('exchangemailstores') • replicated servers ('replicatedservers') • repositories (‘repositories’)
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-all`	For show jobs only. Display al events of a specific type (active/failed/completed) on the core server.
`-protectedserver`	Protected machine with recovery points to display.

Table 148. List command options

Option	Description
-number	Optional. Number of data items to display. Use only with the following specifiers: 'rps', 'activejobs', 'completedjobs', 'failedjobs'. Available values are: • all (fetch all data items) • l[number] or [number] (fetches top ## data items) • f[number] (fetches first ## data items) Only takes effect when displaying recovery points and jobs.
-jobtype	Optional. Filter output by job type. Available values include: • 'transfer' (data transfer) • 'repository' (repository maintenance) • 'replication' (local and remote replications) • 'backup' (backup and restore) • 'bootcdbuilder' (create boot CDs) • 'diagnostics' (upload logs) • 'exchange' (Exchange Server files check) • 'export' (recovery point export) • 'pushinstall' (deploy agents) • 'rollback' (recovery point rollbacks) • 'rollup' (protected machine rollups) • 'sqlattach' (agent attachability checks) • 'mount' (mount repository)

Examples:

List the 30 most recent recovery points:

```
>aacmd /list rps -core 10.10.10.10 -user administrator -password 23WE@#$sdd
-protectedserver 10.10.5.22 -number l30
```

View all failed data transfer jobs performed by a protected machine:

```
>aacmd /list failed jobs -core 10.10.10.10 -user administrator -password 23WE@#$sdd
-protectedserver 10.10.5.22 -number all -jobtype transfer
```

Mount

The `Mount` command mounts a snapshot of one or more drives. You can specify whether the mount should be read, write, or read-only with previous writes. The default selection is read-only.

Usage

The usage for the command is as follows:

```
/mount -core [host name] -user [user name] -password [password] -protectedserver
[name | IP address] -mounttype [read | write | readOnlyWithPreviousWrites] -drives
[drive names] -volumes [volume names] -path [location] -rpn [number | numbers] | -
time [time string]
```

Command Options

The following table describes the options available for the `Mount` command:

Table 149. Mount command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-protectedserver`	Protected machine with a recovery point or points to be mounted.
`-mounttype`	Optional. Specifies a mount mode. Available values are `'read' (read-only)`, `'readOnlyWithPreviousWrites'` (read-only with previous writes), `'write'` (writable). The default mode is `read-only`.
`-volumes`	Optional. List of volume names to mount. If not specified, all volumes are mounted. Values must be enclosed in double quotes and separated by spaces; for example: "c:" "d:". Do not use trailing slashes in volume names.
`-path`	Path to a folder on the core server to which the recovery point should be mounted. If one does not exist, a folder is automatically created.
`-rpn`	Optional. The sequential number of a recovery point to mount (use `/list rps` command to get the numbers). Specify several space-separated numbers to mount multiple recovery points with a single command. In this case data from each recovery point will be stored in a separate child folder. Note: if neither option -time nor -rpn is specified then the most recent recovery point that successfully passed integrity check will be mounted.
`-time`	Optional. Determines recovery point or points to be selected for mount. Available values include: 'latest', 'passed', exact time in the format "mm/dd/yyyy hh:mm tt" (for instance, "2/24/2012 09:00 AM"). Keep in mind to specify date time values of the time zone set on your PC. If neither the -time option nor the -rpn option is specified, then the most recent recovery point that successfully passed an integrity check is mounted.
`-localdrive`	Optional. Perform mount to user disk on local PC.

Examples:

Mount the most recent recovery points containing volumes “c:\” and “d:\” in the read-only mode:

```
>aacmd /mount -core 10.10.10.10 -user administrator -password 23WE@#$sdd -
protectedserver 10.10.5.22 -path c:\mountedrecoverypoint -mounttype read -volumes
"c:" "d:"
```

Mount recovery points with numbers 2 and 7:

```
>aacmd /mount -core 10.10.10.10 -user administrator -password 23WE@#$sdd -
protectedserver 10.10.5.22 -path c:\mountedrecoverypoint -rpn 2 7
```

Pause [snapshot | vmexport | replication]

An administrator can pause snapshots, export to virtual machines, or replicate a Core. The `Pause` command accepts three parameters: `snapshot`, `vmexport`, and `replication`. Only one parameter can be specified. A snapshot can be paused until a certain time, if a time parameter is specified.

A user can pause replication in three ways:

- On a source Core for all Agents (`-[outgoing]`).

  The administrator must specify the remote machine name with the outgoing replication pairing to pause outgoing replication on the source Core:

  ```
  >aacmd /Pause replication /o 10.10.12.10
  ```

- On the source Core for a single Agent (`-protectedserver`):

  ```
  >aacmd /Pause replication /protectedserver 10.10.12.97
  ```

- On target Core (`-incoming`).

  If the local Core is a target Core, the administrator can pause replication by specifying the source Core using the incoming parameter:

  ```
  >aacmd /Pause replication /i 10.10.12.25
  ```

Usage

The usage for the command is as follows:

```
/pause [snapshot | vmexport | replication] -core [host name] -user [user name]
-password [password] -all | -protectedserver [name | IP address] -incoming [host
name] | outgoing [host name] -time [time string]
```

Command Options

The following table describes the options available for the `Pause` command:

Table 150. Pause command options

Option	Description
`-?`	Display this help message.
`-pause`	[`snapshots`], [`replication`] or [`vmexport`].
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-all`	Pause all agents on the selected Core.
`-protectedserver`	Pause current protected server.
`-incoming`	Host name of the remote core that replicates to the core machine.
`-outgoing`	Host name of the remote target core to which data is replicated.
`-time`	The time in the format 'Day-Hours-Minutes' when the snapshots will be resumed (only for snapshots pause).

Examples:

Pause creating snapshots for a specific protected server:

```
>aacmd /pause snapshot -core 10.10.10.10 -user administrator -password 23WE@#$sdd
-protectedserver 10.10.10.4
```

Pause creating snapshots for a protected machine and resume it after three days, 20 hours, and 50 minutes:

```
>aacmd /pause snapshot -core 10.10.10.10 -user administrator -password 23WE@#$sdd
-protectedserver 10.10.10.4 -time 3-20-50
```

Pause export to virtual machine for all protected machines on the core:

```
>aacmd /pause vmexport -core 10.10.10.10 /user administrator -password 23WE@#$sdd -
all
```

Pause outgoing replication on the core for a specific protected machine:

```
>aacmd /pause replication –core 10.10.10.10 -user administrator -password 23WE@#$sdd
–protectedserver 10.10.1.76
```

Pause outgoing replication for all protected machines on the target core:

```
>aacmd /pause replication -core 10.10.10.10 -user administrator -password -
23WE@#$sdd –outgoing 10.10.1.63
```

Pause incoming replication for all machines on the target core:

```
>aacmd /pause replication –core 10.10.10.10 -user administrator -password 23WE@#$sdd
–incoming 10.10.1.82
```

Protect

The `Protect` command adds a server under protection by a core.

Usage

The usage for the command is as follows:

```
/protect -core [host name] -user [user name] -password [password] -repository [name]
-agentname [name | IP address] -agentusername [user name] -agentpassword [password]
-agentport [port] -volumes [volume names]
```

Command Options

The following table describes the options available for the `Protect` command:

Table 151. Protect command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-repository`	Name of a repository on the Core to which the protected machine data should be stored. The name must be enclosed in double quotes.
`-agentname`	Name or IP address of the server you want to protect.
`-agentusername`	User name for the server to be protected.
`-agentpassword`	Password for the server to be protected.

Table 151. Protect command options

Option	Description
-agentport	Protected server port number.
-volumes	List of volumes to protect. Values must be enclosed in double quotes and separated by a space. Do not use trailing slashes in volume names; for example: “c:” “d:”.

Example:

Protect specific volumes of a server with the Core:

```
>aacmd /protect -core 10.10.10.10 -username administrator -password 23WE@#$sdd -
repository “Repository 1” -agentname 10.10.9.120 -agentport 5002 -agentusername
administrator agentpassword 12345 -volumes “c:” “d:”
```

ProtectCluster

The `ProtectCluster` command adds a cluster under protection by a core.

Usage

The usage for the command is as follows:

```
/protectcluster -core [host name] -user [user name] -password [password] -repository
[name] -clustername [name | IP address] -clusterusername [user name] -
clusterpassword [password] -clusterport [port] -clustervolumes [volume names] -
clusternodes [cluster nodes collection]
```

Command Options

The following table describes the options available for the `ProtectCluster` command:

Table 152. ProtectCluster command options

Option	Description
-?	Display this help message.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
-repository	Name of a repository on the Core to which the protected machine data should be stored. The name must be enclosed in double quotes.
-clustername	Name or IP address of the cluster you want to protect.
-clusterusername	User name for the cluster to be protected.
-clusterpassword	Password for the cluster to be protected.
-clusterport	Protected cluster server port number.
-clustervolumes	List of volumes to protect. Values must be enclosed in double quotes and separated by a space. Do not use trailing slashes in volume names; for example: “c:” “d:”.
-clusternodes	List of the cluster nodes and the volumes you want to protect on each node.

Example:

Protect specific volumes of a cluster server with the Core:

```
>aacmd /protectcluster -core 10.10.10.10 -username administrator -password
23WE@#$sdd -repository "Repository 1" -clustername 10.10.8.150 -clusterport 8006 -
clusterusername clusterAdmin clusterpassword password -volumes
"C:\ClusterStorage\Volume1" -clusternodes nodeName 10.10.8.150 volumes "c:" nodeName
10.10.8.151 volumes "c:"
```

RemoveAgent

The `RemoveAgent` command lets you remove a protected machine from the protection of a Core and optionally delete the recovery points of the removed machine. If you do not delete the recovery points, AppAssure retains and labels them as a recovery points only machine.

Usage

The usage for the command is as follows:

```
/removeagent -core [host name] -user [user name] -password [password] -
protectedserver [name | IP address] -deleterecoverypoints
```

Command Options

The following table describes the options available for the `RemoveAgent` command:

Table 153. RemoveAgent command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-protectedserver`	The name or IP address of the server you want to remove from protection.
`-deleterecoverypoints`	Optional. Deletes all recovery points for the machine you want to remove.

Example:

Remove a machine from protection and delete the associated recovery points:

```
>aacmd /removeagent -protectedserver 10.10.1.1 -deleterecoverypoints
```

RemovePoints

The `RemovePoints` command lets you delete specific recovery points of a protected machine.

Usage

The usage for the command is as follows:

```
/removepoints -core [host name] -user [user name] -password [password] -
protectedserver [name | IP address] -rpn [number | numbers] | -time [time string]
```

Command Options

The following table describes the options available for the `RemovePoints` command:

Table 154. RemovePoints command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-protectedserver`	The name or IP address of the server for which you want to delete recovery points
`-rpn`	Optional. The sequential number of a recovery point to be deleted (use `/list rps` command to get the numbers). Specify several space-separated numbers to delete multiple recovery points with a single command.
`-time`	Optional. Determines which recovery point or points to delete by creation time. Specify the exact time in the format "mm/dd/yyyy hh:mm tt" (for example, "2/24/2012 09:00 AM"). Keep in mind to specify the date time values of the time zone set on your PC.

Example:

Delete the recovery points with number 5 and 7:

```
>aacmd /removepoints -core 10.10.10.10 -user administrator -password 23WE@#$sdd -
protectedserver 10.10.5.22 -rpn 5 7
```

RestoreArchive

This command restores an archive from a local archive or share and places the restored data in a specified repository.

Usage

The usage for the command is as follows:

```
/restorearchive -core [host name] -user [user name] -password [password] -all | -
protectedserver [name | IP address] -repository [name] -archiveusername [name] -
archivepassword [password] -path [location]
```

Command Options

The following table describes the options available for the `RestoreArchive` command:

Table 155. RestoreArchive command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-all`	Restore data for all protected machines from the archive files.
`-protectedserver`	Protected machine with recovery points to restore. You can specify several machine names enclosed in double quotes and separated by spaces.
`-repository`	Name of a repository on the Core to which the restored recovery points should be placed. The name must be enclosed in double quotes.
`-archiveusername`	Optional. User name for the remote machine. Required for network path only.
`-archivepassword`	Optional. Password to the remote machine. Required for network path only.
`-path`	Location of the archived data to be restored; for example: d:\work\archive or network path \\servename\sharename.

Examples:

Restore archived data for all protected servers:

```
>aacmd /restorearchive -core 10.10.10.10 -username administrator -password
23WE@#$sdd -all -repository repository1 -path d:\work\archive
```

Restore archived data for specific protected servers:

```
>aacmd /restorearchive -core 10.10.10.10 -username administrator -password
23WE@#$sdd -protectedserver "10.10.20.30" "20.10.10.5" -repository repository1 -path
d:\work\archive
```

Resume [snapshot | vmexport | replication]

The administrator can resume snapshots, export to a virtual machine, and replicate. You must specify your need to resume by a parameter. The following parameters are valid: `snapshot`, `vmexport`, and `replication`. See Pause [snapshot | vmexport | replication] for more details.

Usage

The usage for the command is as follows:

```
/resume [snapshot | vmexport | replication] -core [host name] -user [user name]
-password [password] -all | -protectedserver [name | IP address] -incoming [host
name] | outgoing [host name] -time [time string]
```

Command Options

The following table describes the options available for the `Resume` command:

Table 156. Resume command options

Option	Description
`-?`	Display this help message.
`-restore`	[`snapshots`], [`replication`] or [`vmexport`].
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-all`	Resume all agents on the selected Core.
`-protectedserver`	Resume current protected server.
`-incoming`	Host name of the remote core that replicates to the core machine.
`-outgoing`	Host name of the remote target core to which data is replicated.

Examples:

Resume snapshots for specific protected server:

```
>aacmd /resume snapshot -core 10.10.10.10 -user administrator -password 23WE@#$sdd
-protectedserver 10.10.10.4
```

Resume export to a virtual machine for all protected machines on the core:

```
>aacmd /resume vmexport -core 10.10.10.10 -user administrator -password 23WE@#$sdd -
all
```

Resume outgoing replication on the core for a specific protected machine:

```
>aacmd /resume replication -core 10.10.10.10 -user administrator -password
23WE@#$sdd -protectedserver 10.10.1.76
```

Resume outgoing replication for all protected machines on the target core:

```
>aacmd /resume replication -core 10.10.10.10 -user administrator -password
23WE@#$sdd -outgoing 10.10.1.63
```

Resume incoming replication for all machines on the target core:

```
>aacmd /resume replication -core 10.10.10.10 -user administrator -password
23WE@#$sdd -incoming 10.10.1.82
```

StartExport

The `StartExport` command forces a one-time export of data from a protected machine to a virtual server. You can export to an ESXi, VMware Workstation, Hyper-V, or VirtualBox virtual machine. If exporting to ESXi, you must specify thick or thin disk provisioning.

Usage

The usage for the command is as follows:

```
/startexport -exporttype [esxi | vm | hyperv | vb] -core [host name] -user [user name] -password [password] -protectedserver [name | IP address] -volumes [volume names] -rpn [recovery point number | numbers] | -time [time string] -vmname [virtual machine name] -hostname [virtual host name] -hostport [virtual hostport number] -hostusername [virtual host user name] -hostpassword [virtual host password] [-ram [total megabytes] | -usesourceram] -diskprovisioning [thin | thick] -diskmapping [automatic | manual | withvm] -targetpath [location] -pathusername [user name] -pathpassword [password] [-uselocalmachine]
```

Command Options

The following table describes the options available for the `StartExport` command:

Table 157. StartExport command options

Option	Description
`-?`	Display this help message.
`-exporttype`	Perform export of data from protected server to an ESXi server ('esxi'), VMware Workstation server ('vm'), Hyper-V server ('hyperv'), or VirtualBox server ('vb').
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-protectedserver`	Protected machine with recovery points to be exported.
`-volumes`	Optional. List of volume names to be exported. If not specified, all volumes will be exported. Values must be enclosed in double quotes and separated with spaces; for example: “c:” “d:”. Do not use trailing slashes in volume names.
`-rpn`	Optional. The sequential number of a recovery point to be exported (use Get-RecoveryPoints command to get the numbers). If neither the ‘time’ nor the ‘rpn’ option is specified, then the most recent recovery point is exported.
`-time`	Optional. Determines the recovery point or points to be selected for export. You need to specify the exact time in the format "mm/dd/yyyy hh:mm tt" (for example, "2/24/2012 09:00 AM"). Be sure to specify the date time values of the time zone set on your PC. Note: if neither the 'time' nor the 'rpn' option is specified, then the most recent recovery point is exported.
`-vmname`	ESXi/VMware Workstation export only. Windows name of the virtual machine.
`-hostname`	ESXi/Hyper-V export only. Virtual server host name.
`-linuxhostname`	VirtualBox export only. Virtual server host name.
`-hostport`	ESXi/Hyper-V export only. Virtual server port number.
`-hostusername`	ESXi/Hyper-V export only. User name for the virtual server host.

Table 157. StartExport command options

Option	Description
-hostpassword	ESXi/Hyper-V export only. Password to the virtual server host.
-ram	ESXi/Hyper-V export only. Allocate a specific amount of RAM on the virtual server.
-usesourceram	ESXi/Hyper-V export only. Optional. Allocate the same amount of RAM on the virtual server as there is on the source protected machine.
-diskprovisioning	ESXi export only. Optional. The amount of disk space to be allocated on the virtual machine. Specify 'thick' to make the virtual disk as large as the original drive on the protected server, or 'thin' to allocate the amount of actual disk space occupied on the original drive + some extra megabytes. The default selection is 'thin' provisioning.
-diskmapping	ESXi export only. Optional. Available values: 'auto', 'manual', 'withvm'. The default value is auto-mapping.
-targetpath	VMware Workstation or VirtualBox export only. Local or network path (or Linux path for VirtualBox export only) to the folder where the virtual machine files should be stored.
-pathusername	VMware Workstation export only. User name for network machine. Only required when you specify network path in -targetpath.
-pathpassword	VMware Workstation export only. Password for network machine. Only required when you specify network path in -targetpath.
-uselocalmachine	Hyper-V export only. Optional. Connect to the local Hyper-V server. In this case, options 'hostname', 'hostport', 'hostusername' and 'hostpassword' are ignored.

Examples:

Export data to an ESXi virtual machine with a specific name and the same amount of RAM and disk size as the source protected server:

```
>aacmd /startexport -exporttype esxi -core 10.10.10.10 -user administrator -password
23WE@#$sdd -protectedserver 10.10.5.22 -vmname Win2008-Smith -hostname 10.10.10.23 -
hostport 443 -hostusername root -hostpassword 12QWsdxc@# -usesourceram -
diskprovisioning thick
```

Create a VMware Workstation machine file on the local drive with protected data from recovery point #4:

```
>aacmd /startexport -exporttype vmstation -core 10.10.10.10 -user administrator -
password 23WE@#$sdd -protectedserver 10.10.5.22 -rpn 4 -vmname Win2008-Smith -
targetpath c:\virtualmachines -ram 4096
```

Create a Hyper-V machine files to be stored on a remote machine:

```
>aacmd /startexport -exporttype hyperv -core 10.10.10.10 -user administrator -
password 23WE@#$sdd -protectedserver 10.10.5.22 -vmlocation \\WIN7-
Bobby\virtualmachines -hostname 10.10.10.23 -hostport 443 -hostusername root -
hostpassword 12QWsdxc@# -ram 4096
```

UpdateRepository

The `UpdateRepository` command adds a new storage location to an existing repository.

Usage

The usage for the command is as follows:

```
/updaterepository -name [repository name] -size [size of the repository] [-datapath [data path] -metadatapath [metadata path] | [-uncpath [UNC path] -shareusername [share user name] -sharepassword [share password] -core [host name] -user [user name] -password [password]
```

Command Options

The following table describes the options available for the `UpdateRepository` command:

Table 158. UpdateRepository command options

Option	Description
`-?`	Display this help message.
`-name`	Repository name.
`-size`	Size of repository storage location. Available units are b, Kb, Mb, Gb, Tb, and Pb.
`-datapath`	For local location only. Determines data path of repository storage location.
`-metadatapath`	For local location only. Determines metadata path of repository storage location.
`-uncpath`	For share location only. Determines data and metadata paths of repository storage location.
`-shareusername`	For share location only. Determines user name to share location.
`-sharepassword`	For share location only. Determines password to share location.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.

Examples:

Create a new storage location in a local repository:

```
>aacmd /updaterepository -name "Repository 1" -size 200Gb -datapath d:\repository -metadatapath d:\repository -core 10.10.10.10:8006 -username administrator -password 23WE@#$sdd
```

Create a storage location at a shared location:

```
>aacmd /updaterepository -name "Repository 1" -size 200Gb -uncpath \\share\repository -shareusername login -sharepassword 23WE@#$sdd -core 10.10.10.10:8006 -username administrator -password 23WE@#$sdd
```

Version

The `Version` command displays information about the version of the AppAssure software installed on the specified server. If you do not specify a core or protected server, the information returned applies to the Core on which you are currently working.

Usage

The usage for the command is as follows:

```
/[version | ver] -protectedserver [name | IP address]
```

Command Options

The following table describes the options available for the `Version` command:

Table 159. Version command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default, the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you must also provide a password. If none is provided, then the credentials for the logged-on user are used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none is provided, then the credentials for the logged-on user are used.
`-protectedserver`	Optional. The protected machine for which you want to view version information. If you do not specify a protect machine, the return is information about the Core machine on which you are working.

Example:

Display information about the version of AppAssure installed on the current AppAssure Core:

```
>aacmd.exe /version
```

Localization

When running on the same machine on which AppAssure Core is installed, the the AppAssure Command Line Management utility bases its display language on the language set for the AppAssure Core. In this release, supported languages include English, Chinese (Simplified), French, Korean, German, Japanese, Portuguese (Brazil), and Spanish.

If the AppAssure Command Line Management utility is installed on a separate machine, English is the only language supported.

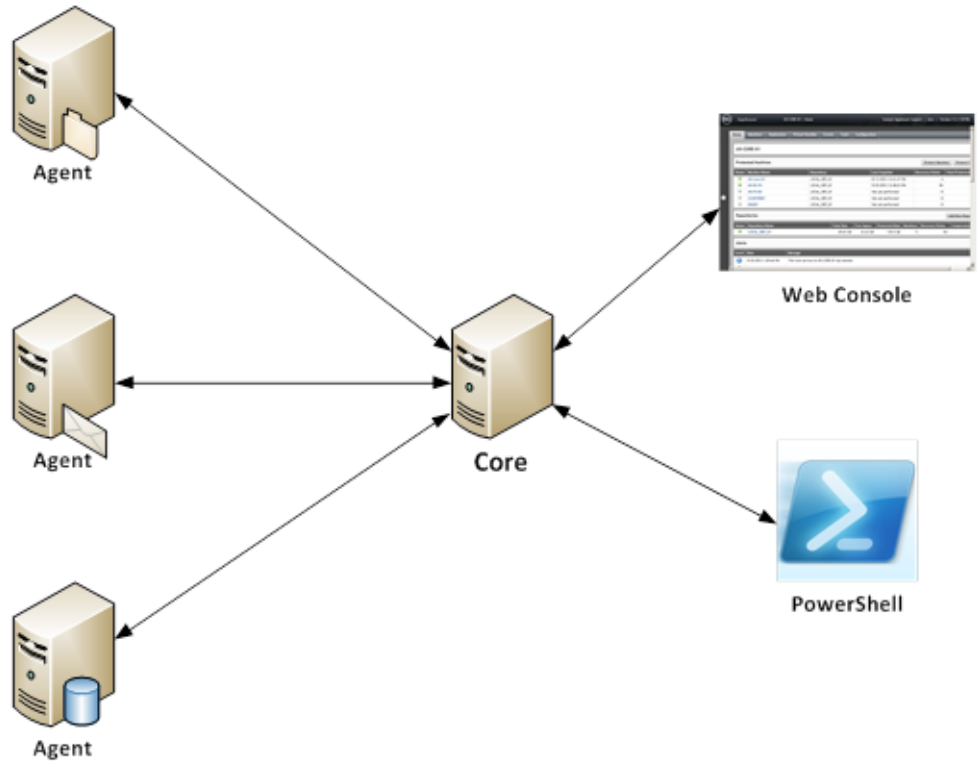
B

Understanding the AppAssure PowerShell module

The AppAssure backup and disaster recovery product consists of several software components, including the AppAssure Agent, the AppAssure Core, and the AppAssure PowerShell Module.

- The AppAssure Agent is responsible for taking volume snapshots and for fast transfer of the data to the AppAssure Core.
- The AppAssure Core stores snapshot data for machines protected by the Agent software, and offers other enhanced features such as bare metal restore to dissimilar hardware, virtual export and replication.
- The AppAssure PowerShell Module is a Windows utility that lets users interact with the AppAssure Core server through the use of PowerShell scripts. This module offers some of the same functionality that is provided by the AppAssure Core Console graphic user interface. For example, the AppAssure PowerShell Module can mount AppAssure recovery points or force a snapshot of a protected machine.

Figure 10. PowerShell interacts with the AppAssure Core



Windows PowerShell is a Microsoft .NET Framework-connected environment designed for administrative automation. This section describes the AppAssure PowerShell module and the cmdlets that can be used by administrators to script certain functions without interaction with the AppAssure Core graphic user interface.

This section includes the following topics:

- Prerequisites for using PowerShell
- Working with commands and cmdlets

NOTE: You can also run PowerShell scripts as pre and post scripts. For more information and sample scripts, see Extending AppAssure jobs using scripting.

Prerequisites for using PowerShell

Before using the AppAssure PowerShell module, you must have Windows PowerShell 2.0 or later installed. Due to new features introduced in PowerShell 3.0, including easier access to object properties, PowerShell Web access, and support for REST calls, Dell recommends using PowerShell 3.0 or later.

NOTE: Make sure to place the powershell.exe.config file in the PowerShell home directory. For example, C:\WindowsPowerShell\powershell.exe.config

powershell.exe.config

```
<?xml version="1.0"?>
<configuration>
      <startup useLegacyV2RuntimeActivationPolicy="true">
            <supportedRuntime version="v4.0.30319"/>
            <supportedRuntime version="v2.0.50727"/>
      </startup>
</configuration>
```

Launching PowerShell and importing the module

Unlike other system modules, the AppAssure PowerShell Module is not loaded by default. For each session, you can open Windows PowerShell with administrative privileges, and then import the module. Complete the steps in this procedure to launch PowerShell and import the AppAssure PowerShell Module.

To launch PowerShell and import the AppAssure PowerShell module

1. Open an elevated command prompt for Windows PowerShell. For example, type Windows PowerShell in the Start menu, and for the resulting Windows PowerShell application, right-click and select **Run as administrator**.

   Windows PowerShell opens in a new command window.

2. Enter the following command and then press Enter:

   ```
   Import-Module "AppAssurePowerShellModule"
   ```

   The AppAssure PowerShell module is imported for your current session. You can begin to run cmdlets in the existing command window.

Working with commands and cmdlets

Cmdlets are specialized commands in a Windows PowerShell script that perform a single function. A cmdlet is typically expressed as a verb-noun pair. The result returned by a cmdlet is an object.

You can pipeline PowerShell commands, which enables the output of one cmdlet to be piped as input to another cmdlet. As a simple example, you can request the list of commands in the AppAssure PowerShell module, and sort that list by name. The example script for this is:

```
Get-Command -module appassurepowershellmodule | sort-object name
```

Getting cmdlet help and examples

Once you have opened PowerShell and imported the AppAssure PowerShell module, you can request additional information at any time by using the Get-Help <command_name> cmdlet. For example, to get information about the virtual machine export cmdlet, enter the following cmdlet and then press Enter:

```
Get-Help Start-VMExport
```

The object returned includes the command name, synopsis, syntax, and any options you can use with the command.

Another method to get help for a specific cmdlet is to type the command name followed by `-?`. For example:

```
Start-VMExport -?
```

You can also request examples for a cmdlet by executing the following command:

```
>Get-Help Start-VMExport -examples
```

AppAssure PowerShell module cmdlets

This section describes the cmdlets and options available in the AppAssure PowerShell Module. The available cmdlets are listed in the following table.

Table 160. Cmdlets in the AppAssure PowerShell Module

Cmdlet name	Description
Get-ActiveJobs	Retrieve a collection of active jobs
Get-Clusters	Retrieve a collection of protected clusters
Get-CompletedJobs	Retrieve a collection of completed jobs
Get-ExchangeMailStores	Retrieve a collection of Exchange mail stores
Get-Failed	Get information about failed recovery points
Get-FailedJobs	Retrieve a collection of failed jobs
Get-Mounts	Show all mounted recovery points
Get-Passed	Get information about passed recovery points
Get-ProtectedServers	Get information about protected servers
Get-ProtectionGroups	Retrieve a collection of protection groups
Get-RecoveryPoints	Get information about recovery points
Get-ReplicatedServers	Get information about replicated servers
Get-Repositories	Get information about repositories
Get-SqlDatabases	Retrieve a collection of SQL databases
Get-UnprotectedVolumes	Retrieve a collection of unprotected volumes
Get-VirtualizedServers	Get information about virtualized servers

Table 160. Cmdlets in the AppAssure PowerShell Module

Cmdlet name	Description
Get-Volumes	Get information about volumes
New-Base	Force base image snapshot
New-Mount	Mount recovery points
New-Repository	Create new repository
New-Snapshot	Force snapshot
Push-Replication	Force replication
Push-Rollup	Force rollup
Remove-Mount	Dismount recovery point
Remove-Mounts	Dismount all mounted recovery points
Resume-Replication	Resume replication
Resume-Snapshot	Resume snapshot
Resume-VMExport	Resume virtual machine export
Start-Archive	Archive recovery points
Start-AttachabilityCheck	Force attachability check for protected MS SQL databases
Start-EsxiExport	Force export to an ESXi server
Start-HypervExport	Force export to a Hyper-V server
Start-LogTruncation	Force log truncation
Start-MountabilityCheck	Force mountability check for protected Exchange mail stores
Start-Protect	Put a server under protection
Start-ProtectCluster	Put a cluster under protection
Start-RestoreArchive	Restore archive with recovery points
Start-VBExport	Force export to a VirtualBox server
Start-VMExport	Force export to a VMWare Workstation server
Stop-ActiveJobs	Cancel active jobs
Suspend-Replication	Pause replication
Suspend-Snapshot	Pause snapshot
Suspend-VMExport	Pause virtual machine export
Update-Repository	Add extent to repository

Get-ActiveJobs

The `Get-ActiveJobs` command returns all active jobs from the Core. The `-jobtype` parameter could be used to observe specific jobs.

Usage

The usage for the command is as follows:

```
Get-ActiveJobs -core [host name] -user [user name] -password [password] -all |
-protectedserver [server name or IP address] -number [all | f[number] |l[number] |
number] -jobtype [type] -time [time]
```

Command Options

The following table describes the options available for the `Get-ActiveJobs` command:

Table 161. Get-ActiveJobs command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Show jobs for a specific protected machine, indicated by IP address.
`-all`	Show all jobs, including those performed by the Core and all protected servers.
`-number`	Optional. Determine how many records to display. available values are: all (display all jobs); l[number] or [number] (fetches ## most recent jobs sorted by execution and time); f[number] (displays first ## recovery jobs sorted by execution and time). By default, the 20 most recent jobs are shown.
`-jobtype`	Optional. Specifies the job type filter. Available values are: `'transfer'` (data transfer), `'repository'` (repository maintenance), `'replication'` (local and remote replications), `'backup'` (backup and restore), `'bootcdbuilder'` (create boot CDs), `'diagnostics'` (upload logs), `'exchange'` (Exchange Server files check), `'export'` (recovery point export), `'pushinstall'` (deploy agents), `'rollback'` (restoring from a recovery point), `'rollup'` (recovery point rollups), `'sqlattach'` (agent attachability checks), and `'mount'` (mount repository). By default, all jobs of the specified type are returned.
`-time`	Optional. Filter output by date and time for the job started. Available types of input include: #d or DD (where # is a number for the period of time of days before now until now) #h or #H (where # is number for the period of hours before now until now) “time date 1”, “time date 2” (to show a custom range of time from a specific date appearing before the comma to a specific date following the comma).

Example:

Lists all active jobs on the local Core:

```
>Get-activejobs -all
```

Get-Clusters

The `Get-Clusters` command returns information about server clusters protected in the Core.

Usage

The usage for the command is as follows:

```
Get-Clusters -core [host name] -user [user name] -password [password]
```

Command Options

The following table describes the options available for the `Get-Clusters` command:

Table 162. Get-Clusters command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.

Example:

List server clusters protected on the local Core:

```
>Get-Clusters
```

Get-CompletedJobs

The `Get-CompletedJobs` command returns a list of jobs completed on the Core. The `-jobtype` parameter could be used to observe specific jobs.

Usage

The usage for the command is as follows:

```
Get-CompletedJobs -core [host name] -user [user name] -password [password] -all |
-protectedserver [server name or IP address] -number [all | f[number] |l[number] |
number] -jobtype [type] -time [time]
```

Command Options

The following table describes the options available for the `Get-CompletedJobs` command:

Table 163. Get-CompletedJobs command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Show jobs for a specific protected machine, indicated by IP address.
`-all`	Show all jobs, including those performed by the Core and all protected servers.

Table 163. Get-CompletedJobs command options

Option	Description
-number	Optional. Determine how many records to display. available values are: all (display all jobs); l[number] or [number] (fetches ## most recent jobs sorted by execution and time); f[number] (displays first ## recovery jobs sorted by execution and time). By default, the 20 most recent jobs are shown.
-jobtype	Optional. Specifies the job type filter. Available values are: 'transfer' (data transfer), 'repository' (repository maintenance), 'replication' (local and remote replications), 'backup' (backup and restore), 'bootcdbuilder' (create boot CDs), 'diagnostics' (upload logs), 'exchange' (Exchange Server files check), 'export' (recovery point export), 'pushinstall' (deploy agents), 'rollback' (restoring from a recovery point), 'rollup' (recovery point rollups), 'sqlattach' (agent attachability checks), and 'mount' (mount repository). By default, all jobs of the specified type are returned.
-time	Optional. Filter output by date and time for the job started. Available types of input include: #d or DD (where # is a number for the period of time of days before now until now) #h or #H (where # is number for the period of hours before now until now) “time date 1”, “time date 2” (to show a custom range of time from a specific date appearing before the comma to a specific date following the comma).

Example:

Lists all active jobs on the local Core:

```
>Get-CompletedJobs -all
```

Lists all completed create repository jobs on the local Core:

```
>Get-CompletedJobs -jobtype repository
```

Get-ExchangeMailStores

The Get-ExchangeMailStores command returns information about male stores on Exchange servers Protected by the Core.

Usage

The usage for the command is as follows:

```
Get-ExchangeMailStores -core [host name] -user [user name] -password [password]
-protectedserver [server name or IP address]
```

Command Options

The following table describes the options available for the Get-ExchangeMailStores command:

Table 164. Get-ExchangeMailStores command options

Option	Description
-?	Display this help message.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.

Table 164. Get-ExchangeMailStores command options

Option	Description
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.
-protectedserver	Show jobs for a specific protected machine, indicated by IP address.

Example:

Lists Exchange mail stores for Exchange server for the local Core:

```
>Get-ExchangeMailStores -protectedserver 10.10.10.10
```

Get-Failed

The `Get-Failed` command returns information about failed recovery points on the local Core.

Usage

The usage for the command is as follows:

```
Get-Failed -core [host name] -user [user name] -password [password] -all |
-protectedserver [server name or IP address] -number [all | f[number] |l[number] |
number]
```

Command Options

The following table describes the options available for the `Get-Failed` command:

Table 165. Get-Failed command options

Option	Description
-?	Display this help message.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.
-protectedserver	Show jobs for a specific protected machine, indicated by IP address.
-number	Optional. Determine how many records to display. available values are: all (display all jobs); l[number] or [number] (fetches ## most recent jobs sorted by execution and time); f[number] (displays first ## recovery jobs sorted by execution and time). By default, the 20 most recent jobs are shown.

Example:

Lists all failed recovery points:

```
>Get-failed -protectedserver 10.10.10.10
```

Get-FailedJobs

The `Get-FailedJobs` command returns all failed jobs from the local Core.

Usage

The usage for the command is as follows:

```
Get-FailedJobs -core [host name] -user [user name] -password [password] -all |
-protectedserver [server name or IP address] -number [all | f[number] |l[number] |
number] -jobtype [type] -time [time]
```

Command Options

The following table describes the options available for the `Get-FailedJobs` command:

Table 166. Get-FailedJobs command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Show jobs for a specific protected machine, indicated by IP address.
`-all`	Show all jobs, including those performed by the Core and all protected servers.
`-number`	Optional. Determine how many records to display. available values are: all (display all jobs); l[number] or [number] (fetches ## most recent jobs sorted by execution and time); f[number] (displays first ## recovery jobs sorted by execution and time). By default, the 20 most recent jobs are shown.
`-jobtype`	Optional. Specifies the job type filter. Available values are: `'transfer'` (data transfer), `'repository'` (repository maintenance), `'replication'` (local and remote replications), `'backup'` (backup and restore), `'bootcdbuilder'` (create boot CDs), `'diagnostics'` (upload logs), `'exchange'` (Exchange Server files check), `'export'` (recovery point export), `'pushinstall'` (deploy agents), `'rollback'` (restoring from a recovery point), `'rollup'` (recovery point rollups), `'sqlattach'` (agent attachability checks), and `'mount'` (mount repository). By default, all jobs of the specified type are returned.
`-time`	Optional. Filter output by date and time for the job started. Available types of input include: #d or DD (where # is a number for the period of time of days before now until now) #h or #H (where # is number for the period of hours before now until now) “time date 1”, “time date 2” (to show a custom range of time from a specific date appearing before the comma to a specific date following the comma).

Example:

Lists all failed jobs on the local Core:

```
>Get-FailedJobs -all
```

Lists all failed create backup jobs on the local Core:

```
>Get-FailedJobs -type backup
```

Get-Mounts

The `Get-Mounts` command returns all recovery points mounted on the local Core.

Usage

The usage for the command is as follows:

```
Get-Mounts -core [host name] -user [user name] -password [password] -protectedserver
[server name or IP address]
```

Command Options

The following table describes the options available for the `Get-Mounts` command:

Table 167. Get-Mounts command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Show jobs for a specific protected machine, indicated by IP address.

Example:

Show all mounted recovery points:

```
>Get-Mounts -core 10.10.10.10:8006 -user administrator -password 23WE@#$sdd
-protectedserver 10.10.5.22
```

Get-Passed

The `Get-Passed` command returns information about recovery points that have passed verification checks on the Core.

Usage

The usage for the command is as follows:

```
Get-Passed -core [host name] -user [user name] -password [password] -protectedserver
[server name or IP address] -number [all | f[number] |l[number] | number]
```

Command Options

The following table describes the options available for the `Get-Passed` command:

Table 168. Get-Passed command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.

Example:

Lists all recovery points on the local Core the passed verification checks:

```
>Get-Passed -protectedserver 10.10.10.10
```

Get-ProtectedServers

The `Get-ProtectedServers` command information about machines protected on the local Core.

Usage

The usage for the command is as follows:

```
Get-ProtectedServers -core [host name] -user [user name] -password [password]
```

Command Options

The following table describes the options available for the `Get-ProtectedServers` command:

Table 169. Get-ProtectedServers command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.

Table 169. Get-ProtectedServers command options

Option	Description
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.

Example:

Lists all machines currently protected on the local Core:

```
>Get-ProtectedServers
```

Get-ProtectionGroups

The Get-ProtectionGroups command returns information about protection groups on the local Core.

Usage

The usage for the command is as follows:

```
Get-ProtectionGroups -core [host name] -user [user name] -password [password] -all |
-protectedserver [server name or IP address]
```

Command Options

The following table describes the options available for the Get-ProtectionGroups command:

Table 170. Get-ProtectionGroups command options

Option	Description
-?	Display this help message.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.
-protectedserver	Show jobs for a specific protected machine, indicated by IP address.

Example:

Lists protection groups on the local Core:

```
>Get-ProtectionGroups -protectedserver 10.10.10.10
```

Get-RecoveryPoints

The `Get-RecoveryPoints` command returns information about recovery points for machines protected on the local Core.

Usage

The usage for the command is as follows:

```
Get-RecoveryPoints -core [host name] -user [user name] -password [password]
-protectedserver [server name or IP address] -number [all | f[number] |l[number] |
number]
```

Command Options

The following table describes the options available for the `Get-RecoveryPoints` command:

Table 171. Get-RecoveryPoints command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Show jobs for a specific protected machine, indicated by IP address.
`-number`	Optional. Determine how many records to display. available values are: all (display all jobs); l[number] or [number] (fetches ## most recent jobs sorted by execution and time); f[number] (displays first ## recovery jobs sorted by execution and time). By default, the 20 most recent jobs are shown.

Example:

Lists recovery points for machines protected on the local Core:

```
>Get-RecoveryPoints -protectedserver 10.10.10.10
```

Get-ReplicatedServers

The `Get-ReplicatedServers` command returns information about machines replicated on the Core.

Usage

The usage for the command is as follows:

```
Get-ReplicatedServers -core [host name] -user [user name] -password [password]
```

Command Options

The following table describes the options available for the `Get-ReplicatedServers` command:

Table 172. Get-ReplicatedServers command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.

Example:

Lists all replicated servers on the local Core:

```
>Get-ReplicatedServers
```

Get-Repositories

The `Get-Repositories` command returns information about repositories on the Core.

Usage

The usage for the command is as follows:

```
Get-Repositories -core [host name] -user [user name] -password [password]
```

Command Options

The following table describes the options available for the `Get-Repositories` command:

Table 173. Get-Repositories command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.

Example:

Lists repositories on the local Core:

```
>Get-Repositories
```

Get-SqlDatabases

The `Get-SqlDatabases` command returns a list of SQL databases from the specified protected machine.

Usage

The usage for the command is as follows:

```
Get-SqlDatabases -core [host name] -user [user name] -password [password]
-protectedserver [server name or IP address]
```

Command Options

The following table describes the options available for the `Get-SqlDatabases` command:

Table 174. Get-SqlDatabases command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Show jobs for a specific protected machine, indicated by IP address.

Example:

Lists all SQL databases jobs on the local Core:

```
>Get-SqlDatabases -protectedserver 10.10.10.10
```

Get-UnprotectedVolumes

The `Get-UnprotectedVolumes` command returns information about volumes that are available for protection but not currently protected on the Core.

Usage

The usage for the command is as follows:

```
Get-UnprotectedVolumes
-core [host name] -user [user name] -password [password] -protectedserver [server
name or IP address]
```

Command Options

The following table describes the options available for the `Get-UnprotectedVolumes` command:

Table 175. Get-UnprotectedVolumes command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Show jobs for a specific protected machine, indicated by IP address.

Example:

Lists all volumes available for protection (but not get protected) on the specified agent machine:

```
>Get-UnprotectedVolumes -protectedserver 10.10.10.10
```

Get-VirtualizedServers

The `Get-VirtualizedServers` command returns information about virtualized servers.

Usage

The usage for the command is as follows:

```
Get-VirtualizedServers -core [host name] -user [user name] -password [password]
```

Command Options

The following table describes the options available for the `Get-VirtualizedServers` command:

Table 176. Get-VirtualizedServers command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.

Example:

Lists all virtualized servers on the local Core:

```
>Get-VirtualizedServers
```

Get-Volumes

The `Get-Volumes` command returns information about volumes on a specified machine that is protected by the Core.

Usage

The usage for the command is as follows:

```
Get-Volumes -core [host name] -user [user name] -password [password]
-protectedserver [server name or IP address]
```

Command Options

The following table describes the options available for the `Get-Volumes` command:

Table 177. Get-Volumes command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Show jobs for a specific protected machine, indicated by IP address.

Example:

Lists all volumes on the specified machine:

```
>Get-Volumes -protectedserver 10.10.10.10
```

New-Base

The `New-Base` command forces a new base image resulting in a data transfer for the current protected machine. When you force a base image, the transfer will start immediately or will be added to the queue. Only the data that has changed from a previous recovery point will be transferred. If there is no previous recovery point, all data on the protected volumes will be transferred.

Usage

The usage for the command is as follows:

```
New-Base [[-all] | -protectedserver [machine name]] -core [host name] -user [user
name] -password [password]
```

Command Options

The following table describes the options available for the `New-Base` command:

Table 178. New-Base command options

Option	Description
`-?`	Display this help message.
`-all`	Base image for all agents.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Force for the current protected machine's name.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.

Example:

Force base image for all protected machines:

```
>New-Base -all
```

New-Mount

The `New-Mount` command mounts a snapshot of one or more drives.

Usage

The usage for the command is as follows:

```
New-Mount -core [host name] -user [user name] -password [password]
-protectedserver [machine name] -mounttype [read | write |
readonlywithpreviouswrites] -drives [drive names] -path [location] -time [MM/DD/YYYY
hh:mm:ss tt | passed | latest] -rpn [number]
```

Command Options

The following table describes the options available for the `New-Mount` command:

Table 179. New-Mount command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-protectedserver`	The protected server IP address or machine name (depends on how the particular machine was protected.
`-time`	Optional. The timestamp of the Recovery Point to mount. This should be in the format that is specified by the OS on the current PC. The administrator is able to get the latest recovery point by specifying latest or last checked recovery point by passed parameter value. By default the latest time option is chosen.

Table 179. New-Mount command options

Option	Description
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.
-path	Path on the Core machine to which recovery points will be mounted.
-mounttype	Optional. Specifies a mount mode. Available options are 'read', 'readOnlyWithPreviousWrites' (read-only with previous writes), 'write' (writable). Default mode is read-only.
-volumes	Optional. Space-separated list of volume names to mount. If the volume's name contains spaces or special characters, it has to be specified using double quotes. If not specified, all volumes will be mounted.
-drivers	Optional. Comma-separated list of volume names to mount. If not specified, all volumes will be mounted. **NOTE:** This option is obsolete, use '-volumes' instead.
-rpn	Optional. Recovery point number for the mount. You can obtain this using the get-mounts command. Specify several numbers for the rpn parameter to mount different points with a single command. **NOTE:** If you set an array of points to mount, each point will be located in a separate child directory. The name describes the time when the recovery point was created. When you call dismount, all child directories will be removed. You should remove the parent directory manually.

Example:

```
>New-Mount -core 10.10.10.10:8006 -user administrator -password 23WE@#$sdd
-protectedserver 10.10.5.22 -path C:\MountedRecoveryPoint -mounttype read -volumes c
"d, ko"
```

Mount an array of recovery points:

```
>New-Mount -rpn 10 52 41 -protectedserver localhost -path "D:/Folder for mount"
```

Mount a recovery point with certain time created:

```
>New-Mount -protectedserver 10.10.5.56 -path "D:/Folder for mount" -time "8/24/2012
11:46 AM"
```

New-Repository

The `New-Repository` command creates a new repository in the AppAssure Core. The size specified must be between 250 Mb and 16 Tb.

Usage

The usage for the command is as follows:

```
New-Repository | –name [name] -size [size] -datapath [location] -metadatapath [location]
```

Command Options

The following table describes the options available for the `New-Repository` command:

Table 180. New-Repository command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-name`	Repository name.
`-size`	Size of repository extent. Available units are: b, Kb, Mb, Gb, Tb, Pb.
`-datapath`	For local location only. Determines data path of repository extent.
`-metadatapath`	For local location only. Determines metadata path of repository extent.
`-uncpath`	For share location only. Determines data and metadata paths of repository extent.
`-shareusername`	For share location only. Determines login to share location.
`-sharepassword`	For share location only. Determines password to share location.
`-comment`	Optional. Description of repository.
`-concurrent Operations`	Optional. Maximum number of operations that can be pending at one time. Value by default: 64.

Example:

Create new repository of minimum size in local drive E:

```
>New-Repository –name Repository2 -size 250Mb -datapath e:\Repository\Data -metadatapath e:\Repository\Metadata
```

New-Snapshot

The `New-Snapshot` command forces a snapshot resulting in a data transfer for the current protected machine. When you force a snapshot, the transfer will start immediately or will be added to the queue. Only the data that has changed from a previous recovery point will be transferred. If there is no previous recovery point, all data on the protected volumes will be transferred.

Usage

The usage for the command is as follows:

```
New-Snapshot [-all] | -protectedserver [machine name]] -core [host name] -user [user name] -password [password]
```

Command Options

The following table describes the options available for the `New-Snapshot` command:

Table 181. New-Snapshot command options

Option	Description
`-?`	Display this help message.
`-all`	Force all protected machines.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Force for the current protected machine's name.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.

Example:

Force a snapshot for all protected machines:

```
>New-Snapshot -all
```

Push-Replication

The `Push-Replication` command forces replication for one or more protected machines.

Usage

The usage for the command is as follows:

```
Push-Replication -core [host name] -user [user name] -password [password]
-targetcore [host name] -all | -protectedserver [machine name | IP address]
```

Command Options

The following table describes the options available for the `Push-Replication` command:

Table 182. Push-Replication command options

Option	Description
`-?`	Display this help message.
`-all`	Force replication for all machines being replicated to the target Core.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Protected machine name on the target Core against which to force replication.
`-user`	Optional. Login for the remote Core host machine. If you specify a login, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.

Example:

Push replication for a single protected machine:

```
>Push-Replication -core 10.10.10.10:8006 -user administrator -password 23WE@#$sdd
-targetcore 10.10.10.20:8006 -protectedserver 10.10.5.22
```

Push replication for all protected machines:

```
>Push-Replication -all
```

Push-Rollup

The `Push-Rollup` command forces rollup for a protected machine.

Usage

The usage for the command is as follows:

```
Push-Rollup -core [host name] -user [user name] -password [password]
-protectedserver [machine name | IP address]
```

Command Options

The following table describes the options available for the `Push-Rollup` command:

Table 183. Push-Rollup command options

Option	Description
`-?`	Display this help message.
`-all`	Force all protected machines.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.

Table 183. Push-Rollup command options

Option	Description
-protectedserver	Force for the current protected machine's name.
-user	Optional. Login for the remote Core host machine. If you specify a login, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.

Example:

Push rollup for a single protected machine:

```
>Push-Rollup -core 10.10.10.10:8006 -user administrator -password 23WE@#$sdd -
protectedserver 10.10.5.22
```

Push rollup for all protected machines:

```
>Push-Rollup -all
```

Remove-Mount

The Remove-Mount command dismounts a mounted recovery point specified by the /Path. Dismount points for the selected machine using the -protectedserver parameter or dismount points for all the mounted recovery points by using the -all parameter.

Usage

The usage for the command is as follows:

```
Remove-Mount -core [host name] -user [user name] -password [password]
[-protectedserver [machine name] | -path [mount path]]
```

Command Options

The following table describes the options available for the Remove-Mount command:

Table 184. Remove-Mount command options

Option	Description
-?	Display this help message.
-all	Dismount all mounted recovery points.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a log on. If none are provided, then the logged-on user's credentials will be used.
-path	Dismount selected mount point.
-protectedserver	Dismount all mounted recovery points for the current protected machine.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.

Example:

Dismount the recovery point specified by the path:

```
>Remove-Mount -core 10.10.10.10:8006 -user administrator -password 23WE@#$sdd -path
C:\mountedRecoveryPoint
```

Remove-Mounts

The `Remove-Mounts` command dismounts all mounted recovery points.

Usage

The usage for the command is as follows:

```
Remove-Mounts -core [host name] -user [user name] -password [password]
```

Command Options

The following table describes the options available for the `Remove-Mounts` command:

Table 185. Remove-Mounts command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.

Example:

Dismount all recovery points on the specified Core:

```
>Remove-Mounts -core 10.10.10.10:8006 -user administrator -password 23WE@#$sdd
```

Resume-Replication

The `Resume-Replication` command lets you resume replication. See Suspend-Replication for more details.

Usage

The usage for the command is as follows:

```
Resume-Replication -core [host name] -user [user name] -password [password] -all | -protectedserver [machine name | IP address] -incoming [host name] | -outgoing [host name]
```

Command Options

The following table describes the options available for the `Resume-Replication` command:

Table 186. Resume-Replication command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-all`	All protected servers.
`-protectedserver`	Resume replication for the specified machine.
`-incoming`	Host name of the remote Core that replicates to the Core machine. Replication is resumed for all protected machines on the remote Core.
`-outgoing`	Host name of the remote target core to which data is replicating. Replication is resumed for all protected machines on the remote core.

Example:

Resume replication for the protected machine with IP 10.10.10.4 for the local Core:

```
>Resume-Replication -protectedserver 10.10.10.4
```

Resume-Snapshot

An administrator is able to resume snapshots, export to virtual machines, and perform replication. See Start-VMExport for more details.

Usage

The usage for the command is as follows:

```
Resume-Snapshot -core [host name] -user [user name] -password [password] -all | -protectedserver [name | IP address]
```

Command Options

The following table describes the options available for the `Resume-Snapshot` command:

Table 187. Resume-Snapshot command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-all`	All protected servers.
`-protectedserver`	Resume snapshot for the specified machine.

Example:

Resume snapshots for the protected machine with IP 10.10.10.4 for the local Core:

```
>Resume-Snapshot -protectedserver 10.10.10.4
```

Resume-VMExport

The `Resume-VMExport` command lets an administrator export to virtual machines. See Suspend-VMExport for more details.

Usage

The usage for the command is as follows:

```
Resume-VMExport -core [host name] -user [user name] -password [password] -all | -protectedserver [name | IP address]
```

Command Options

The following table describes the options available for the `Resume-VMExport` command:

Table 188. Resume-VMExport command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.

Table 188. Resume-VMExport command options

Option	Description
-all	All protected servers.
-protectedserver	Resume snapshot for the specified machine.

Example:

Resume export to a virtual machine for each protected machine on the local Core:

```
>Resume-VMExport -all
```

Start-Archive

Businesses often use long-term storage to archive both compliant and non-compliant data. The archive feature in AppAssure is used to support the extended retention for compliant and non-compliant data. The administrator can save an archive on the local storage or network location by specifying the `/Path` command and credentials.

Usage

The usage for the command is as follows:

```
Start-Archive -path -startdate -enddate [-all] | -protectedserver [machine name] or
[IP]] -core [host name] -user [user name] -password [password]
```

Command Options

The following table describes the options available for the `Start-Archive` command:

Table 189. Start-Archive command options

Option	Description
-?	Display this help message.
-path	Location path. Example path: **'D:\work\archive'** or network path: **'\\servername\sharename'**.
-all	Archive recovery points for all machines on the Core.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
-startdate	Start date of the date range for the created recovery points. Should be in the format specified by the OS on the current PC.
-enddate	End date of the date range. Defaults to the current time.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
-protectedserver	Archive recovery points for the specified machine.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
-archiveusername	Optional. Required for network path only.
-archivepassword	Optional. Required for network path only.
-comment	Optional. Example: `-comment 'Before install new application'`.

Example:

Archive all recovery points for all machines on the Core:

```
>Start-Archive -path D:\work\archive -startdate 'Example 04/30/2012' -all
```

Start-AttachabilityCheck

The `Start-AttachabilityCheck` command forces an attachability check for all SQL Server databases protected by the Core.

Usage

The usage for the command is as follows:

```
Start-AttachabilityCheck -core [host name] -user [username] - password [password]
- protectedserver [machine name | IP address] -rpn [number | numbers] | -time [time
string]
```

Command Options

The following table describes the options available for the `Start-AttachabilityCheck` command:

Table 190. Start-AttachabilityCheck command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	The protected machine on which to perform the SQL attachability check.
`-rpn`	Optional. The sequential number of a recovery point on which to perform the SQL attachability check. You can use the `-GetRecoveryPoints` command to obtain recovery point numbers. You can specify several space-separated numbers to perform the checks against multiple recovery points with a single command. **NOTE:** If neither 'time' nor 'rpn' option is specified in this command, than the most recent recovery point is used for the attachability check.
`-time`	Optional. Determines recovery point to be selected for SQL attachability check. You need to specify exact time in the format "MM/DD/YYYY hh:mm tt" (for example: "04/24/2015 09:00 AM")." Specify date time values of the time zone set on your local machine. **NOTE:** If neither 'time' nor 'rpn' option is specified in this command, than the most recent recovery point will be exported.

Example:

Perform a SQL attachability check on the most recent recovery point for the specified protected SQL server:

```
>Start-AttachabilityCheck - protectedserver 10.10.9.120
```

Start-EsxiExport

The `Start-EsxiExport` command initiates the launch of a virtual export from the selected recovery point to an ESX(i) server virtual machine.

Required parameters include the name of the protected machine containing recovery points to export; the name of the virtual machine you are exporting to; the amount of RAM to be allocated on the virtual machine; the host name and port of the Linux server host, and the path to the local, network, or Linux folder where the resulting virtual machine files will be stored.

Usage

The usage for the command is as follows:

```
Start-EsxiExport -core [host name] -user [user name] -password [password]
-protectedserver [machine name | IP address] -volumes [volume names] -rpn [number |
numbers] | -time [time string] -vmname [virtual machine name] -hostname [virtual
host name] -hostport [virtual host port number] -hostusername [virtual host user
name] hostpassword [virtual host password] [-ram [total megabytes] | -usesourceram]
-diskprovisioning [thin | thick] -diskmapping [automatic | manual | withvm]
```

Command Options

The following table describes the options available for the `Start-EsxiExport` command:

Table 191. Start-EsxiExport command options

Option	Description
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Protected machine with recovery points to be exported.
`-volumes`	Optional. List of volume names to be exported. If not specified, all volumes in the specified recovery points will be exported. Values must be enclosed in double quotes, each separated by a space. do not use trailing slashes in volume names. For example, specify “C:” not “C:/”
`-rpn`	Optional. The sequential number of a recovery point to be exported. (You can use the `-GetRecoveryPoints` command to obtain recovery point numbers. **NOTE:** If neither ‘time’ nor ‘rpn’ option is specified in this command, than the most recent recovery point will be exported.
`-time`	Optional. Determines recovery point to be selected for export. You need to specify exact time in the format “MM/DD/YYYY hh:mm tt” (for example: “04/24/2015 09:00 AM”).” Specify date time values of the time zone set on your local machine. **NOTE:** If neither ‘time’ nor ‘rpn’ option is specified in this command, than the most recent recovery point will be exported.
`-vmname`	Windows name of the virtual machine.
`-hostname`	The virtual server host name.
`-hostport`	The virtual server port number.
`-hostusername`	The user name to the virtual server host.
`-hostpassword`	The password to the virtual server host.

Table 191. Start-EsxiExport command options

Option	Description
-ram	Allocate specific amount of RAM on the virtual server.
-usesourceram	Optional. Allocate the same amount of RAM on the virtual server as the source protected machine.
-diskprovisioning	Optional. The amount of disk space that will be allocated on the virtual machine. Specify 'thick' to make the virtual disk as large as the original drive on the protected server, or 'thin' to allocate the amount of actual disk space occupied on the original drive, plus some extra space in megabytes. By default, 'thin' provisioning is selected.
-diskmapping	Optional. Select either 'auto,' 'manual,' or 'withvm'. By default, auto-mapping is enabled.

Start-HypervExport

The Start-HypervExport command initiates the launch of a virtual export from the selected recovery point to a Hyper-V server virtual machine.

Usage

The usage for the command is as follows:

```
Start-HypervExport -core [host name] -user [user name] -password [password]
-protectedserver [[machine name] or [IP address]] -volumes [volume names] -rpn
[number | numbers] | -time [time string] [-vmname [uselocalmachine] | -hostname
[virtual host name] -hostport [virtual host port number] -hostusername [virtual host
user name] -hostpassword [virtual host password] -vmlocation [location]] [-ram
[total megabytes] | -usesourceram] -diskformat [VHD | VHDX]
```

Command Options

The following table describes the options available for the Start-HypervExport command:

Table 192. Start-HypervExport command options

Option	Description
-?	Display this help message.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
-protectedserver	Protected machine with recovery points to be exported.
-volumes	Optional. List of volume names to be exported. If not specified, all volumes in the specified recovery points will be exported. Values must be enclosed in double quotes, each separated by a space. do not use trailing slashes in volume names. For example, specify "C:" not "C:/"

Table 192. Start-HypervExport command options

Option	Description
`-rpn`	Optional. The sequential number of a recovery point to be exported. (You can use the `-GetRecoveryPoints` command to obtain recovery point numbers. **NOTE:** If neither 'time' nor 'rpn' option is specified in this command, than the most recent recovery point will be exported.
`-time`	Optional. Determines recovery point to be selected for export. You need to specify exact time in the format "MM/DD/YYYY hh:mm tt" (for example: "04/24/2015 09:00 AM")." Specify date time values of the time zone set on your local machine. **NOTE:** If neither 'time' nor 'rpn' option is specified in this command, than the most recent recovery point will be exported.
`-vmname`	Windows name of the virtual machine.
`-uselocalmachine`	Optional. Connect the local Hyper-V server. If this parameter is used, the following options are ignored: hostname, host port, host username, host password.
`-hostname`	The virtual server host name.
`-hostport`	The virtual server port number.
`-hostusername`	The user name to the virtual server host.
`-hostpassword`	The password to the virtual server host.
`-ram`	Allocate specific amount of RAM on the virtual server.
`-usesourceram`	Optional. Allocate the same amount of RAM on the virtual server as the source protected machine.
`-diskprovisioning`	Optional. The amount of disk space that will be allocated on the virtual machine. Specify 'thick' to make the virtual disk as large as the original drive on the protected server, or 'thin' to allocate the amount of actual disk space occupied on the original drive, plus some extra space in megabytes. By default, 'thin' provisioning is selected.
`-diskmapping`	Optional. Select either 'auto,' 'manual,' or 'withvm'. By default, auto-mapping is enabled.
`-diskformat`	Optional. Specify the appropriate disk format from options VHD or VHDX. VHD is otherwise used by default.

Start-LogTruncation

The `Start-LogTruncation` command forces log truncation for the specified protected SQL Server or Microsoft Exchange server.

Usage

The usage for the command is as follows:

```
Start-LogTruncation -core [host name] -user [user name] -password [password]
-protectedserver [[machine name] or [IP address]] -target [sql | exchange]
```

Command Options

The following table describes the options available for the `Start-LogTruncation` command:

Table 193. Start-LogTruncation command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Archive of recovery points for the specified machine.
`-target`	Specify the type of log truncation (either ‘sql’ or ‘exchange’). If not specified, logs are truncated on all databases.

Example:

Truncate SQL logs:

```
>Start-LogTruncation -protectedserver SQL1 -target sql
```

Truncate Exchange server logs: all recovery points for all machines on the Core:

```
> start-LogTruncation -protectedserver ExServer2 -target exchange
```

Start-MountabilityCheck

The `Start-MountabilityCheck` command forces a mountability check for protected Microsoft Exchange mail stores.

Usage

The usage for the command is as follows:

```
Start-MountabilityCheck -core [host name] -user [user name] -password [password]
-protectedserver [[machine name] or [IP address]] -rpn [number | numbers] |
-time [time string]
```

Command Options

The following table describes the options available for the `Start-MountabilityCheck` command:

Table 194. Start-MountabilityCheck command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Archive of recovery points for the specified machine.
`-rpn`	Optional. The sequential number of a recovery point to be exported. (You can use the `-GetRecoveryPoints` command to obtain recovery point numbers. **NOTE:** If neither 'time' nor 'rpn' option is specified in this command, than the most recent recovery point will be exported.
`-time`	Optional. Determines recovery point to be selected for export. You need to specify exact time in the format "MM/DD/YYYY hh:mm tt" (for example: "04/24/2015 09:00 AM")." Specify date time values of the time zone set on your local machine. **NOTE:** If neither 'time' nor 'rpn' option is specified in this command, than the most recent recovery point will be exported.

Example:

Start a mountability check for oall recovery points for all machines on the Core:

```
> Start-MountabilityCheck -protected EX01
```

Start-Protect

The `Start-Protect` command lets an administrator add a server under protection by a Core.

Usage

```
Start-Protect -core [host name] -user [user name] -password [password] -repository
[repository name] -agent [name | IP address] -agentusername [user name]
-agentpassword [password] -agentport [port] -volumes [volume names]
```

Command Options

The following table describes the options available for the `Start-Protect` command:

Table 195. Start-Protect command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.

Table 195. Start-Protect command options

Option	Description
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
-repository	Name of a repository on the Core where the protected machine's data is stored.
-agentname	Protected machine name or IP address.
-agentusername	Log on to the server to be protected.
-agentpassword	Password to the server to be protected.
-agentport	Protected server port number.
-volumes	List of volumes to protect. Values must be enclosed in double quotes and separated by a space. Do not use trailing slashes in volume names. For example, "**c:**" or "**d:**".

Example:

Put volumes of a server under protection:

```
>Start-Protect -repository "Repository 1" -agentname 10.10.9.120 -agentusername
administrator -agentpassword 12345 -agentport 5002 -volumes "c:" "d:"
```

Start-ProtectCluster

The `Start-ProtectCluster` command lets an administrator add a server cluster under protection by a Core.

Usage

Usage for the command is as follows:

```
Start-ProtectCluster -core [host name] -user [user name] -password [password] -
repository [repository name] -clustername [name | IP address] -clusterusername [user
name for cluster] -clusterpassword [password for cluster] -clusterport [port] -
clustervolumes [volume names] -clusternodes [cluster nodes names and volumes]
```

Command Options

The following table describes the options available for the `Start-ProtectCluster` command:

Table 196. Start-ProtectCluster command options

Option	Description
-?	Display this help message.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.

Table 196. Start-ProtectCluster command options

Option	Description
-repository	Name of a repository on the Core where the protected machine's data is stored. The name must be enclosed in double quotes.
-clustername	The name of the cluster to protect.
-clusterusername	User name for the cluster to be protected.
-clusterpassword	Password to the cluster to be protected.
-clusterport	Port number for the cluster to be protected.
-clustervolumes	List of volumes to protect. Values must be in double quotes and separated by a space. Do not use trailing slashes in volume names. For example, “c:”, “d”.
-clusternodes	List of cluster nodes with volumes to protect. First specify label “nodename” and then type the name of the node. Then, specify label “volumes” and then type a list of volumes for the node. For example: “nodename”, “10.10.10.10”, “volumes”, “c:”, “e:”, “nodename”, “10.10.10.11,” “volumes”, “c:”

Example:

Put volumes of a server under protection:

```
>Start-ProtectCluster -repository "Repository 1" -clustername 10.10.9.120
-clusterusername administrator -clusterpassword 12345 -clusterport 5002
-clustervolumes "c:" "d:" -clusternodes nodename 10.10.10.10 volumes “c:” “e:”
```

Start-RestoreArchive

Businesses often use long-term storage to archive both compliant and non-compliant data. The archive feature in AppAssure is used to support the extended retention for compliant and non-compliant data. The administrator can save an archive on the local storage or network location by specifying the /Path command and credentials.

Usage

The usage for the command is as follows:

```
Start-RestoreArchive –path -repository [-all] | -protectedserver [machine name] or
[IP]] -core [host name] -user [user name] -password [password]
```

Command Options

The following table describes the options available for the Start-RestoreArchive command:

Table 197. Start-RestoreArchive command options

Option	Description
-?	Display this help message.
-path	Location path. Example path: **‘D:\work\archive’** or network path: **‘\\servername\sharename’**.
-repository	Repository of the Core for the unprotected machine.
-all	Archive the recovery points for all of the machines on the Core.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.

Table 197. Start-RestoreArchive command options

Option	Description
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
-protectedserver	Archive of recovery points for the specified machine.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
-archiveusername	Optional. Required for network path only.
-archivepassword	Optional. Required for network path only.

Example:

Archive all recovery points for all machines on the Core:

```
>Start-RestoreArchive -path D:\work\archive -startdate 'Example 04/30/2012' -all
```

Start-VBExport

The start-VBExport command initiates the launch of a virtual export from the selected recovery point to an Oracle VirtualBox server virtual machine.

Required parameters include the name of the protected machine containing recovery points to export; the name of the virtual machine you are exporting to; the amount of RAM to be allocated on the virtual machine; the host name and port of the Linux server host, and the path to the local, network, or Linux folder where the resulting virtual machine files will be stored.

Usage

The usage for the command is as follows:

```
Start-VBExport -core -user [user name] -password [password] -protectedserver
[machine name] or [IP address]] -volumes [volume names] -rpn [number | numbers] |
-time [time string] -vmname [virtual machine name] [-ram [total megabytes] |
-usesourceram] -linuxhostname [linux hostname] -hostport [linux port] -targetpath
[location] pathusername [user name] - pathpassword [password]
```

Command Options

The following table describes the options available for the Start-VBExport command:

Table 198. Start-VBExport command options

Option	Description
-?	Display this help message.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
-protectedserver	Protected machine with recovery points to be exported.

Table 198. Start-VBExport command options

Option	Description
-volumes	Optional. List of volume names to be exported. If not specified, all volumes in the specified recovery points will be exported. Values must be enclosed in double quotes, each separated by a space. do not use trailing slashes in volume names. For example, specify “C:” not “C:/”
-rpn	Optional. The sequential number of a recovery point to be exported. (You can use the Get-RecoveryPoints command to obtain recovery point numbers.) **NOTE:** If neither ‘time’ nor ‘rpn’ option is specified in this command, than the most recent recovery point will be exported.
-time	Optional. Determines recovery point to be selected for export. You need to specify exact time in the format “MM/DD/YYYY hh:mm tt” (for example: “04/24/2015 09:00 AM”).” Specify date time values of the time zone set on your local machine. **NOTE:** If neither ‘time’ nor ‘rpn’ option is specified in this command, than the most recent recovery point will be exported.
-vmname	Windows name of the virtual machine.
-ram	Allocate specific amount of RAM on the virtual server.
-usesourceram	Optional. Allocate the same amount of RAM on the virtual server as the source protected machine.
-linuxhostname	Linux VirtualBox server hostname.
-hostport	Linux VirtualBox server port.
-targetpath	Local or network or Linux path to the folder where the virtual machine files are to be stored.
-pathusername	User name for network machine. Only required when you specify network path in parameter -targetpath.
-pathpassword	Password for network machine. Only required when you specify network path in parameter -targetpath.
-accountusername	Optional. Use if you can specify a user account to register the exported virtual machine. For local or network machine only.
-accountpassword	Optional. Use only when you specify a user account to register the exported virtual machine using parameter -accountusername. For local or network machine only.

Example:

Export all volumes from the latest recovery point on machine 10.10.12.97 to a VM called NewVirtualBoxVM:

```
>Start-VBExport -protectedserver 10.10.12.97 -vmname NewVirtualBoxVM -ram
usesourceram -targetpath D:/exports
```

Start-VMExport

The `start-VMExport` command initiates the launch of a virtual export from the selected recovery point to a VMware Workstaation server virtual machine.

Required parameters include the name of the protected machine containing recovery points to export; the name of the virtual machine you are exporting to; the amount of RAM to be allocated on the virtual machine; and the path to the local or network, folder where the resulting virtual machine files will be stored.

Usage

The usage for the command is as follows:

```
Start-VMExport -core -user [user name] -password [password] -protectedserver
[machine name] or [IP address]] -volumes [volume names] -rpn [number | numbers] |
-time [time string] -vmname [virtual machine name] [-ram [total megabytes] |
-usesourceram] -linuxhostnme [linux hostname] -hostport [linux port] -targetpath
[location] pathusername [user name] - pathpassword [password]
```

Command Options

The following table describes the options available for the `Start-VMExport` command:

Table 199. Start-VMExport command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-protectedserver`	Protected machine with recovery points to be exported.
`-volumes`	Optional. List of volume names to be exported. If not specified, all volumes in the specified recovery points will be exported. Values must be enclosed in double quotes, each separated by a space. do not use trailing slashes in volume names. For example, specify “C:” not “C:/”
`-rpn`	Optional. The sequential number of a recovery point to be exported. (You can use the `Get-RecoveryPoints` command to obtain recovery point numbers. **NOTE:** If neither ‘time’ nor ‘rpn’ option is specified in this command, than the most recent recovery point will be exported.
`-time`	Optional. Determines recovery point to be selected for export. You need to specify exact time in the format “MM/DD/YYYY hh:mm tt” (for example: “04/24/2015 09:00 AM”).” Specify date time values of the time zone set on your local machine. **NOTE:** If neither ‘time’ nor ‘rpn’ option is specified in this command, than the most recent recovery point will be exported.
`-vmname`	Windows name of the virtual machine.
`-ram`	Allocate specific amount of RAM on the virtual server.
`-usesourceram`	Optional. Allocate the same amount of RAM on the virtual server as the source protected machine.
`-targetpath`	Local or network or Linux path to the folder where the virtual machine files are to be stored.

Table 199. Start-VMExport command options

Option	Description
-pathusername	User name for network machine. Only required when you specify network path in parameter -targetpath.
-pathpassword	Password for network machine. Only required when you specify network path in parameter -targetpath.
-version	Optional. Use if you can specify a user account to register the exported virtual machine. For local or network machine only.
-version	Version of VMware Tools to use. Valid versions are: 7, 8, 9, and 10.

Example:

Export all volumes from the latest recovery point on machine 10.10.12.97 to a VM called NewVMwareVM:

```
>Start-VBExport -protectedserver 10.10.12.97 -vmname NewVMWareVM -ram usesourceram -
targetpath D:/exports
```

Suspend-Replication

The Suspend-Replication command lets an administrator pause replication.

A user can pause replication in three ways:

- Pause replication on the master Core for all protected machines (-outgoing parameter)

  The administrator must specify the remote machine name with outgoing replication pairing to pause outgoing replication on the master Core.

  ```
  >Suspend-replication -outgoing 10.10.12.10
  ```

- Pause replication on the master Core for a single protected machine (-protectedserver parameter)

  ```
  >Suspend-replication -protectedserver 10.10.12.97
  ```

- Pause replication on the target Core (-incoming parameter)

  If the local Core is a target Core, the administrator can pause replication by specifying the master Core using the -incoming parameter.

Command Options

The following table describes the options available for the Suspend-Replication command:

Table 200. Suspend-Replication command options

Option	Description
-?	Display this help message.
-all	Pauses all protected machines on the selected Core.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
-pause	[snapshots], [replication] or [vmexport].
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
-protectedserver	Pause the current protected server.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.

Table 200. Suspend-Replication command options

Option	Description
-incoming	Host name of the remote Core that replicates to the Core machine. Replication is suspended for all protected machines on the remote Core.
-outgoing	Host name of the remote target core to which data is replicating. Replication is suspended for all protected machines on the remote core.

Example:

Pause outgoing replication on the remote Core with the IP address: 10.10.1.15, for the single protected machine with the IP address: 10.10.1.76:

```
>Suspend-replication -core 10.10.1.15 -protectedserver 10.10.1.76
```

Pause outgoing replication from the local Core to remote target with the IP address: 10.10.1.63 for all protected machines:

```
>Suspend-replication -outgoing 10.10.1.63
```

Pause incoming replication from 10.10.1.82 on the remote Core with the IP address: 10.10.1.15 (Administrator is able to pause incoming replication only for whole machine):

```
>Suspend-replication -core 10.10.1.15 -incoming 10.10.1.82
```

Stop-ActiveJobs

The `Stop-ActiveJobs` cancels active jobs for A specified protected machine.

Usage

The usage for the command is as follows:

```
Stop-ActiveJobs [-protectedserver [machine name | IP address] | -core [host name]]
-user [user name] -password [password] -jobtype [jobtype]
```

Command Options

The following table describes the options available for the `Stop-ActiveJobs` command:

Table 201. Stop-ActiveJobs command options

Option	Description
-?	Display this help message.
-all	Select and cancel events of the specified type for all protected machines.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.

Table 201. Stop-ActiveJobs command options

Option	Description
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
-protectedserver	Determines protected machine on which jobs should be canceled.
-jobtype	Optional. Specifies job type filter. Available values are: 'transfer' (data transfer), 'repository' (repository maintenance), 'replication' (local and remote replications), 'backup' 9backup and restore), 'bootcdbuilder' (create boot CDs), 'diagnostics'(upload logs), 'exchange' (Exchange Server files check), 'export (recovery point export), 'pushinstall' (deploy Agent software to protected machines), 'rollback' (restore data from recovery point), 'rollup' (recovery point rollup's), 'sqlattach' (agent attachability checks), 'mount' (not repository). By default, all jobs of the specified type are canceled.

Example:

Stop transfer job in protected machine:

```
>Stop-ActiveJobs -protectedserver 10.10.1.76 -jobtype transfer
```

Stop all jobs for a specific protected machine:

```
>Stop-ActiveJobs -protectedserver 10.10.1.76 -all
```

Suspend-Snapshot

The `Suspend-Snapshot` command lets an administrator pause snapshots.

Usage

The usage for the command is as follows:

```
Suspend-Snapshot -core [host name] -user [user name] -password [password] -all |
-protectedserver [name | IP address] -time [time string]
```

Command Options

The following table describes the options available for the `Suspend-Snapshot` command:

Table 202. Suspend-Snapshot command options

Option	Description
-?	Display this help message.
-all	Pauses all protected machines on the selected Core.
-core	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
-user	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
-password	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
-time	The time in the format 'Day-Hours-Minutes' when the snapshots will be resumed (only for snapshots pause).

Example:

Pause snapshots for the protected machine with IP 10.10.10.4 for the local Core with a certain time to resume:

```
>Suspend-Snapshot -protectedserver 10.10.10.4 -time 3-20-50
```

Suspend-VMExport

The `Suspend-VMExport` command lets an administrator pause exports to virtual machines.

Usage

```
Suspend-VMExport -core [host name] -user [user name] -password [password] -all | -protectedserver [name | IP address]
```

Command Options

The following table describes the options available for the `Suspend-VMExport` command:

Table 203. Suspend-VMExport command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-all`	Pauses all protected machines on the selected Core.
`-protectedserver`	Pause the current protected server.

Example:

Suspend VM export for the protected machine with IP 10.10.10.4 for the local Core:

```
>Suspend-VMExport -protectedserver 10.10.12.25
```

Update-Repository

The `Update-Repository` command adds an extent to an existing repository. The size specified must be between 250 Mb and 16 Tb.

Usage

```
Update-Repository -name [repository name] -size [size] [[[-datapath [datapath] -metadatapath [metadata path]] | [-uncpath [UNC path] -shareusername [share user name] -sharepassword [share password]]] -core [host name] -user [user name] -password [password]
```

Command Options

The following table describes the options available for the `Update-Repository` command:

Table 204. Update-Repository command options

Option	Description
`-?`	Display this help message.
`-core`	Optional. Remote Core host machine IP address (with an optional port number). By default the connection is made to the Core installed on the local machine.
`-user`	Optional. User name for the remote Core host machine. If you specify a user name, you also have to provide a password. If none are provided, then the logged-on user's credentials will be used.
`-password`	Optional. Password to the remote Core host machine. If you specify a password, you also have to provide a user name. If none are provided, then the logged-on user's credentials will be used.
`-name`	Repository name.
`-size`	Size of repository extent. Available units are: b, Kb, Mb, Gb, Tb, Pb.
`-datapath`	For local location only. Determines data path of repository extent.
`-metadatapath`	For local location only. Determines metadata path of repository extent.
`-uncpath`	For share location only. Determines data and metadata paths of repository extent.
`-shareusername`	For share location only. Determines login to share location.
`-sharepassword`	For share location only. Determines password to share location.

Example:

Add an extent to the repository of the minimum size:

```
>Update-Repository -name Repository1 -size 250Mb -datapath C:\Repository\Data -
metadatapath C:\repository\Metadata
```

Localization

When running on the same machine on which AppAssure Core is installed, the AppAssure PowerShell module bases its display language on the language set for the AppAssure Core. In this release, supported languages include English, Chinese (Simplified), French, Korean, German, Japanese, Portuguese (Brazil), and Spanish.

If the AppAssure PowerShell module is installed on a separate machine, English is the only language supported.

Qualifiers

The following table describes the qualifiers available for AppAssure PowerShell Module.

Table 205.

Qualifier	Usage
`-core <AppAssure Core Name>`	Host name of the Core. Default: `Localhost`
`-ProtectedServer <Protected Server Name>`	Host name/IP address of the AppAssure Agent. Default: `Localhost` if multiple servers protected, otherwise the single server protected.
`-Mode <READ, READWRITE, WRITE>`	Recovery Point Mount mode. Default: `Read`.
`-Volumes <Snapshot Volume Letter>`	Snapshot volume letter from AppAssure Agent. Default: `All`.
`-User <User Name>`	User name used to connect to the AppAssure Core. This is typically the service user.
`-Domain <Domain Name>`	Domain to which the user defined in /User belongs.
`-Password <Password>`	Password of the user defined in `/User`.
`-Path <Target path to mount, dismount recovery points or archive location>`	For example: `C:\AppAssureMount`.

C

Extending AppAssure jobs using scripting

AppAssure enables administrators to automate the administration and management of resources at certain occurrences through the execution of commands and scripts. The AppAssure software supports the use of PowerShell scripting for Windows and Bourne Shell scripting for Linux.

Core jobs are automatically created whenever you initiate operations on the AppAssure Core such as replication, virtual export, or a backup snapshot. You can extend these jobs by running a script before it or after it. These are known as pre and post scripts.

This section describes the scripts that can be used by administrators at designated occurrences in AppAssure for Windows and Linux.

△ **CAUTION: The sample PowerShell and Bourne scripts provided in this document will function when run as designed by qualified administrators. Take precautions when modifying functioning scripts to retain working versions. Any modifications to the script samples included here, or any scripts you create, are considered customization, which is not typically covered by Customer Support.**

This section includes the following topics:

Using PowerShell scripts in AppAssure

Windows PowerShell is a Microsoft .NET Framework-connected environment designed for administrative automation. AppAssure includes comprehensive client software development kits (SDKs) for PowerShell scripting that lets administrative users run user-provided PowerShell scripts at designated occurrences; for example, before or after a snapshot, attachability and mountability checks, and so on. Administrators can run scripts from both the AppAssure Core and the protected machine. Scripts can accept parameters, and the output of a script is written to Core and protected machine log files.

ⓘ **NOTE:** For nightly jobs, preserve one script file and the *JobType* input parameter to distinguish between nightly jobs.

Script files are located in the %ALLUSERSPROFILE%\AppRecovery\Scripts folder.

- In Windows 7, the path to locate the %ALLUSERSPROFILE% folder is: C:\ProgramData.
- In Windows 2003, the path to locate the folder is: Documents and Settings\All Users\Application Data\.

ⓘ **NOTE:** Windows PowerShell is required and must be installed and configured before running AppAssure scripts.

For more information on how using PowerShell scripts see Sample PowerShell scripts, Input Parameters for PowerShell Scripting, Input parameters for Bourne Shell scripting, and Sample Bourne Shell scripts.

Prerequisites for PowerShell scripting

Before running PowerShell scripts for AppAssure, you must have Windows PowerShell 2.0 or later installed. Due to new features introduced in PowerShell 3.0, including easier access to object properties, PowerShell Web access, and support for REST calls, Dell recommends using PowerShell 3.0 or later.

NOTE: Place the powershell.exe.config file in the PowerShell home directory. For example, C:\WindowsPowerShell\powershell.exe.config.

powershell.exe.config

```
<?xml version="1.0"?>
<configuration>
    <startup useLegacyV2RuntimeActivationPolicy="true">
        <supportedRuntime version="v4.0.30319"/>
        <supportedRuntime version="v2.0.50727"/>
    </startup>
</configuration>
```

Testing PowerShell Scripts

If you want to test the scripts you plan to run, you can do so by using the PowerShell graphical editor, *powershell_is*. You also need to add the configuration file, *powershell_ise.exe.config* to the same folder the configuration file, *powershell.exe.config*.

NOTE: The configuration file, *powershell_ise.exe.config* must have the same content as the *powershell.exe.config* file.

CAUTION: **If the pre-PowerShell or post-PowerShell script fails, the job also fails.**

Localization

When running on the same machine on which AppAssure Core is installed, the AppAssure PowerShell module bases its display language on the language set for the AppAssure Core. In this release, supported languages include English, Chinese (Simplified), French, Korean, German, Japanese, Portuguese (Brazil), and Spanish.

If the AppAssure PowerShell module is installed on a separate machine, English is the only language supported.

Qualifiers

The following table describes the qualifiers available for AppAssure PowerShell module.

Table 206. Qualifiers for the AppAssure PowerShell module

Qualifier	Usage
`-core <AppAssure Core Name>`	Host name of the Core. Default: `Localhost`
`-ProtectedServer <Protected Server Name>`	Host name/IP address of the AppAssure Agent. Default: `Localhost` if multiple servers protected, otherwise the single server protected.
`-Mode <READ, READWRITE, WRITE>`	Recovery Point Mount mode. Default: `Read`.
`-Volumes <Snapshot Volume Letter>`	Snapshot volume letter from AppAssure Agent. Default: `All`.
`-User <User Name>`	User name used to connect to the AppAssure Core, which is typically the service user.
`-Domain <Domain Name>`	Domain to which the user defined in /User belongs.
`-Password <Password>`	Password of the user defined in `/User`.
`-Path <Target path to mount, dismount recovery points or archive location>`	For example: `C:\AppAssureMount`.

Input Parameters for PowerShell Scripting

All available input parameters are used in sample scripts. The parameters are described in the following tables.

NOTE: Script files must possess the same name as the sample script files.

AgentProtectionStorageConfiguration (namespace Replay.Common.Contracts.Agents)

The following table presents the available objects for the AgentProtectionStorageConfiguration parameter.

Table 207. Objects for the AgentProtectionStorageConfiguration parameter

Method	Description
public Guid RepositoryId { get; set; }	Gets or sets the ID of the repository where the agent recovery points are stored.
public string EncryptionKeyId { get; set; }	Gets or sets the ID of the encryption key for this agent's recovery points. An empty string means no encryption.

AgentTransferConfiguration (namespace Replay.Common.Contracts.Transfer)

The following table presents the available objects for the AgentTransferConfiguration parameter.

Table 208. Objects for the AgentTransferConfiguration parameter

Method	Description
public uint MaxConcurrentStreams { get; set; }	Gets or sets the maximum number of concurrent TCP connections the Core establishes to the agent for transferring data.
public uint MaxTransferQueueDepth { get; set; }	When a range of blocks are read from a transfer stream, that range is placed on a producer or consumer queue, where a consumer thread reads it and writes it to the epoch object. If the repository writes slower than the network reads, this queue fills up. The point at which the queue is full and reads stop is the maximum transfer queue depth.
public uint MaxConcurrentWrites { get; set; }	Gets or sets the maximum number of block write operations to have outstanding on an epoch at any given time. If additional blocks are received when this many block writes are outstanding, those additional blocks are ignored until one of the outstanding writes finishes.
public ulong MaxSegmentSize { get; set; }	Gets or sets the maximum number of contiguous blocks to transfer in a single request. Depending on testing, higher or lower values may be optimal.
public Priority Priority { get; set; }	Gets or sets the priority for transfer request.
public int MaxRetries { get; set; }	Gets or sets the maximum number of times a failed transfer should be retried before it is presumed failed.
public Guid ProviderId{ get; set; }	Gets or sets the GUID of the VSS provider to use for snapshots on this host. Administrators typically accept the default.
public Collection<ExcludedWriter> ExcludedWriterIds { get; set; }	Gets or sets the collection of VSS writer IDs, which should be excluded from this snapshot. The writer ID is determined by the name of the writer. This name is for documentation purposes only and does not have to exactly match the name of the writer.
public ushort TransferDataServerPort { get; set; }	Gets or sets a value containing the TCP port upon which to accept connections from the Core for the actual transfer of data from the Agent to the Core. The Agent attempts to listen on this port, but if the port is in use, the Agent can use a different port instead. The Core should use the port number specified in the BlockHashesUri and BlockDataUri properties of the VolumeSnapshotInfo object for each snapped volume.
public TimeSpan SnapshotTimeout { get; set; }	Gets or sets the amount of time to wait for a VSS snapshot operation to complete before giving up and timing out.
public TimeSpan TransferTimeout { get; set; }	Gets or sets the amount of time to wait for further contact from the Core before abandoning the snapshot.

Table 208. Objects for the AgentTransferConfiguration parameter

Method	Description
public TimeSpan NetworkReadTimeout { get; set; }	Gets or sets the timeout for network read operations related to this transfer.
public TimeSpan NetworkWriteTimeout { get; set; }	Gets or sets the timeout for network write operations related to this transfer.

BackgroundJobRequest (namespace Replay.Core.Contracts.BackgroundJobs)

The following table presents the available objects for the BackgroundJobRequest parameter.

Table 209. Objects for the BackgroundJobRequest parameter

Method	Description
public Guid AgentId { get; set; }	Gets or sets the ID of the Agent.
public bool IsNightlyJob { get; set; }	Gets or sets the value indicating whether the background job is a nightly job.
public virtual bool InvolvesAgentId(Guid agentId)	Determines the value indicating whether the concrete agent is involved in job.

ChecksumCheckJobRequest (namespace Replay.Core.Contracts.Exchange.ChecksumChecks)

Inherits its values from the parameter, DatabaseCheckJobRequestBase.

DatabaseCheckJobRequestBase (namespace Replay.Core.Contracts.Exchange)

Inherits its values from the parameter, BackgroundJobRequest.

ExportJobRequest (namespace Replay.Core.Contracts.Export)

Inherits its values from the parameter, BackgroundJobRequest.

The following table presents the available objects for the ExportJobRequest parameter.

Table 210. Objects for the ExportJobRequest parameter

Method	Description
public uint RamInMegabytes { get; set; }	Gets or sets the memory size for the exported VM. Set to zero (0) to use the memory size of the source machine.
public VirtualMachineLocation Location { get; set; }	Gets or sets the target location for this export. This is an abstract base class.
public VolumeImageIdsCollection VolumeImageIds { get; private set; }	Gets or sets the volume images to include in the VM export.
public ExportJobPriority Priority { get; set; }	Gets or sets the priority for export request.

NightlyAttachabilityJobRequest (namespace Replay.Core.Contracts.Sql)

Inherits its values from the parameter, BackgroundJobRequest.

RollupJobRequest (namespace Replay.Core.Contracts.Rollup)

Inherits its values from the parameter, BackgroundJobRequest.

TakeSnapshotResponse (namespace Replay.Agent.Contracts.Transfer)

The following table presents the available objects for the TakeSnapshotResponse parameter.

Table 211. Objects for the TakeSnapshotResponse parameter

Method	Description
public Guid SnapshotSetId { get; set; }	Gets or sets the GUID assigned by VSS to this snapshot.
public VolumeSnapshotInfoDictionary VolumeSnapshots { get; set; }	Gets or sets the collection of snapshot info for each volume included in the snap.

TransferJobRequest (namespace Replay.Core.Contracts.Transfer)

Inherits its values from the parameter, BackgroundJobRequest.

The following table presents the available objects for the TransferJobRequest parameter.

Table 212. Objects for the TransferJobRequest parameter

Method	Description
public VolumeNameCollection VolumeNames { get; set; }	Gets or sets the collection of names for transfer. VolumeNames is a data structure that contains the following data: • **GuidName.** The Guid associated with the volume, used as the name if a DisplayName is not set. • **DisplayName.** The displayed name of the volume.
public ShadowCopyType ShadowCopyType { get; set; }	Gets or sets the type of copying for transfer. The available values are: • Unknown • Copy • Full
public AgentTransferConfiguration TransferConfiguration { get; set; }	Gets or sets the transfer configuration.
public AgentProtectionStorageConfiguration StorageConfiguration { get; set; }	Gets or sets the storage configuration.

Table 212. Objects for the TransferJobRequest parameter

Method	Description
public string Key { get; set; }	Generates a pseudorandom (but not cryptographically secure) key, which can be used as a one-time password to authenticate transfer requests.
public bool ForceBaseImage { get; set; }	Gets or sets the value indicating whether the base image was forced or not.
public bool IsLogTruncation { get; set; }	Gets or sets the value indicating whether the job is log truncation or not.

TransferPrescriptParameter (namespace Replay.Common.Contracts.PowerShellExecution)

The following table presents the available objects for the TransferPrescript parameter.

Table 213. Objects for the TransferPrescript parameter

Method	Description
public VolumeNameCollection VolumeNames (get; set;)	Gets or sets the collection of volume names for transfer. VolumeNames is a data structure that contains the following data: • **GuidName**. The Guid associated with the volume, used as the name if a DisplayName is not set. • **DisplayName**. The displayed name of the volume.
public ShadowCopyType ShadowCopyType { get; set; }	Gets or sets the type of copying for transfer.ShadowCopyType is an enumeration with values. The available values are: • Unknown • Copy • Full

Table 213. Objects for the TransferPrescript parameter

Method	Description

public AgentTransferConfiguration TransferConfiguration { get; set; }	Gets or sets the transfer configuration.
	AgentTransferConfiguration is an object which will have the following data:
	• **MaxConcurrentStreams**. The maximum number of concurrent TCP connections the core will establish to the agent for transferring data
	• **MaxTransferQueueDepth**. The maximum number of block extents which can be queued up for writing
	• **MaxConcurrentWrites**. The maximum number of block write operations to have outstanding on an epoch at any given time. If additional blocks are received when this many block writes are outstanding, those additional blocks will be ignored until one of the outstanding blocks gets written.
	• **MaxSegmentSize**. The maximum number of contiguous blocks to transfer in a single request
	• **Priority**. An object which will have the following data:
	  • Undefined
	  • One
	  • Two
	  • Three
	  • Four
	  • Five
	  • Six
	  • Seven
	  • Eight
	  • Nine
	  • Ten
	  • Highest (which is equal to One)
	  • Lowest (which is equal to Ten)
	  • Default (which is equal to Five)
	• MaxRetries. The maximum number of times a failed transfer should be retried before it is presumed failed
	• **UseDefaultMaxRetries**. A value indicating that the maximum number of retries is the default value
	• **ProviderId**. The GUID of the VSS provider to use for snapshots on this host. Users typically use the default setting.

Table 213. Objects for the TransferPrescript parameter

Method	Description
public AgentTransferConfiguration TransferConfiguration { get; set; } (cont.)	• **ExcludedWriterIds**. Collection of VSS writer IDs which should be excluded from this snapshot. The writer ID is keyed by the name of the writer. This name is for documentation purposes only and does not have to exactly match the actual name of the writer. • **TransferDataServerPort**. A value containing the TCP port upon which to accept connections from the core for the actual transfer of data from the agent to the core. • **SnapshotTimeout**. The amount of time to wait for a VSS snapshot operation to complete before giving up and timing out. • **TransferTimeout**. The amount of time to wait for further contact from the core before abandoning the snapshot. • **NetworkReadTimeout**. The timeout for network read operations related to this transfer. • **NetworkWriteTimeout**. The timeout for network write operations related to this transfer. • **InitialQueueSize**. A size of initial queue of requests. • **MinVolumeFreeSpacePercents**. A minimal amount of free space on a volume in percent. • **MaxChangeLogsSizePercents**. A maximum size of driver change logs as part of volume capacity measured in percent. • **EnableVerification**. A value indicating whether diagnostic verification of each block sent to Core should be performed.
public string Key { get; set; }	The Key method generates a pseudorandom (but not cryptographically secure) key, which can be used as a one-time password to authenticate transfer requests.
public bool ForceBaseImage { get; set; }	Gets or sets the value indicating whether the transfer was a forced base image capture.
public bool IsLogTruncation { get; set; }	Gets or sets the value indicating whether logging is being truncated.
public uint LatestEpochSeenByCore { get; set; }	Gets or sets latest epoch value. The LatestEpochSeenByCore method is the ordinal number of the most recent snapshot taken by the Core. This is the 'epoch number' assigned by the filter driver to this particular snapshot at the moment it was taken with VSS.

TransferPostscriptParameter (namespace Replay.Common.Contracts.PowerShellExecution)

The following table presents the available objects for the TransferPostscript parameter.

Table 214. Objects for the TransferPostscript parameter

Method	Description
public VolumeNameCollection VolumeNames (get; set;)	Gets or sets the collection of volume names for transfer. VolumeNames is a data structure that contains the following data: • **GuidName.** The Guid associated with the volume, used as the name if a DisplayName is not set. • **DisplayName**. The displayed name of the volume.
public ShadowCopyType ShadowCopyType { get; set; }	Gets or sets the type of copying for transfer.ShadowCopyType is an enumeration with values. The available values are: • Unknown • Copy • Full

Table 214. Objects for the TransferPostscript parameter

Method	Description

public AgentTransferConfiguration TransferConfiguration { get; set; }	Gets or sets the transfer configuration.
	AgentTransferConfiguration is an object which will have the following data:
	• **MaxConcurrentStreams**. The maximum number of concurrent TCP connections the core will establish to the agent for transferring data
	• **MaxTransferQueueDepth**. The maximum number of block extents which can be queued up for writing
	• **MaxConcurrentWrites**. The maximum number of block write operations to have outstanding on an epoch at any given time. If additional blocks are received when this many block writes are outstanding, those additional blocks will be ignored until one of the outstanding blocks gets written.
	• **MaxSegmentSize**. The maximum number of contiguous blocks to transfer in a single request
	• **Priority**. An object which has the following data:
	  • "Undefined
	  • "One
	  • "Two
	  • "Three
	  • "Four
	  • "Five
	  • "Six
	  • "Seven
	  • "Eight
	  • "Nine
	  • "Ten
	  • "Highest (which is equal to One)
	  • "Lowest (which is equal to Ten)
	  • "Default (which is equal to Five)
	• **MaxRetries**. The maximum number of times a failed transfer should be retried before it is presumed failed
	• **UseDefaultMaxRetries**. A value indicating that the maximum number of retries is the default value
	• **ProviderId**. The GUID of the VSS provider to use for snapshots on this host. Administrators typically accept the default.

Table 214. Objects for the TransferPostscript parameter

Method	Description
public AgentTransferConfiguration TransferConfiguration { get; set; } (cont.)	• **ExcludedWriterIds**. Collection of VSS writer IDs which should be excluded from this snapshot. The writer ID is keyed by the name of the writer. This name is for documentation purposes only and does not have to exactly match the actual name of the writer. • **TransferDataServerPort**. A value containing the TCP port upon which to accept connections from the core for the actual transfer of data from the agent to the core. • **SnapshotTimeout**. The amount of time to wait for a VSS snapshot operation to complete before giving up and timing out. • **TransferTimeout**. The amount of time to wait for further contact from the core before abandoning the snapshot. • **NetworkReadTimeout**. The timeout for network read operations related to this transfer. • **NetworkWriteTimeout**. The timeout for network write operations related to this transfer. • **InitialQueueSize**. A size of initial queue of requests. • **MinVolumeFreeSpacePercents**. A minimal amount of free space on a volume in percent. • **MaxChangeLogsSizePercents**. A maximum size of driver change logs as part of volume capacity measured in percent. • **EnableVerification**. A value indicating whether diagnostic verification of each block sent to Core should be performed.
public AgentProtectionStorageConfiguration StorageConfiguration { get; set; }	Gets or sets the storage configuration The AgentProtectionStorageConfiguration object contains the following data: • **RepositoryId**. The name of the repository where this agent's recovery points will be stored • **EncryptionKeyId**. The ID of the encryption key for this agent's recovery points. An empty string means no encryption
public string Key { get; set; }	The Key method generates a pseudorandom (but not cryptographically secure) key, which can be used as a one-time password to authenticate transfer requests.
public bool ForceBaseImage { get; set; }	Gets or sets the value indicating whether the transfer was a forced base image capture.
public bool IsLogTruncation { get; set; }	Gets or sets the value indicating whether logging is being truncated.

Table 214. Objects for the TransferPostscript parameter

Method	Description
public uint LatestEpochSeenByCore { get; set; }	Gets or sets latest epoch value. The LatestEpochSeenByCore method is the ordinal number of the most recent snapshot taken by the Core. This is the 'epoch number' assigned by the filter driver to this particular snapshot at the moment it was taken with VSS.
public Guid SnapshotSetId { get; set; }	Gets or sets the GUID assigned by VSS to this snapshot.
public VolumeSnapshotInfoDictionary VolumeSnapshots { get; set; }	Gets or sets the collection of snapshot info for each volume included in the snap.

VirtualMachineLocation (namespace Replay.Common.Contracts.Virtualization)

The following table presents the available objects for the VirtualMachineLocation parameter.

Table 215. Objects for the VirtualMachineLocation parameter

Method	Description
public string Description { get; set;}	Gets or sets a human-readable description of this location.
public string Method { get; set;}	Gets or sets the name of the VM.

VolumeImageIdsCollection (namespace Replay.Core.Contracts.RecoveryPoints)

Inherits its values from the parameter, System.Collections.ObjectModel.Collection<string>.

VolumeName (namespace Replay.Common.Contracts.Metadata.Storage)

The following table presents the available objects for the VolumeName parameter.

Table 216. Objects for the VolumeName parameter

Method	Description
public string GuidName { get; set;}	Gets or sets the ID of the volume.
public string DisplayName { get; set;}	Gets or sets the name of the volume.
public string UrlEncode()	Gets a URL-encoded version of the name which can be passed cleanly on a URL. **NOTE:** A known issue exists in .NET 4.0 WCF (https://connect.microsoft.com/VisualStudio/feedback/ViewFeedback.aspx?FeedbackID=413312), which prevents path escape characters from working correctly in a URI template. Because a volume name contains both '\' and '?', you must replace the special characters '\' and '?' with other special characters.
public string GetMountName()	Returns a name for this volume that is valid for mounting volume image to some folder.

VolumeNameCollection (namespace Replay.Common.Contracts.Metadata.Storage)

Inherits its values from the parameter, System.Collections.ObjectModel.Collection<VolumeName>.

The following table presents the available objects for the VolumeNameCollection parameter.

Table 217. Objects for the VolumeNameCollection parameter

Method	Description
public override bool Equals(object obj)	Determines whether this instance and a specified object, which must also be a VolumeNameCollection object, have the same value. (Overrides Object.Equals(Object).)
public override int GetHashCode()	Returns the hash code for this VolumeNameCollection. (Overrides Object.GetHashCode().)

VolumeSnapshotInfo (namesapce Replay.Common.Contracts.Transfer)

The following table presents the available objects for the VolumeSnapshotInfo parameter.

Table 218. Objects for the VolumeSnapshotInfo parameter

Method	Description
public Uri BlockHashesUri { get; set;}	Gets or sets the URI at which the MD5 hashes of volume blocks can be read.
public Uri BlockDataUri { get; set;}	Gets or sets the URI at which the volume data blocks can be read.

VolumeSnapshotInfoDictionary (namespace Replay.Common.Contracts.Transfer)

Inherits its values from the parameter, System.Collections.Generic.Dictionary<VolumeName, VolumeSnapshotInfo>.

Sample PowerShell scripts

The following sample scripts are provided to assist administrative users in executing PowerShell scripts. The sample scripts include:

- PreTransferScript.ps1
- PostTransferScript.ps1
- PreExportScript.ps1
- PostExportScript.ps1
- PreNightlyJobScript.ps1
- PostNightlyJobScript.ps1

PreTransferScript.ps1

The PreTransferScript is run on the protected machine before transferring a snapshot.

Sample PreTransferScript

```
# receiving parameter from transfer job
param([object]$TransferPrescriptParameter)

# building path to Agent's Common.Contracts.dll and loading this assembly
$regLM = [Microsoft.Win32.Registry]::LocalMachine
$regLM =
$regLM.OpenSubKey('SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\AppRecovery
Agent 5')
$regVal = $regLM.GetValue('InstallLocation')
$regVal = $regVal + 'Common.Contracts.dll'
[System.Reflection.Assembly]::LoadFrom($regVal)  |  out-null

# Converting input parameter into specific object
$TransferPrescriptParameterObject = $TransferPrescriptParameter -as
[Replay.Common.Contracts.PowerShellExecution.TransferPrescriptParameter];

# Working with input object. All echo's are logged
if($TransferPrescriptParameterObject -eq $null) {

     echo 'TransferPrescriptParameterObject parameter is null'
}
else {

     echo
     'TransferConfiguration:'$TransferPrescriptParameterObject.TransferConfiguratio
     n

     echo 'StorageConfiguration:'
     $TransferPrescriptParameterObject.StorageConfiguration
}
```

PostTransferScript.ps1

The PostTransferScript is run on the protected machine after transferring a snapshot.

Sample PostTransferScript

```
# receiving parameter from transfer job
param([object] $TransferPostscriptParameter)

# building path to Agent's Common.Contracts.dll and loading this assembly

$regLM = [Microsoft.Win32.Registry]::LocalMachine
$regLM =
$regLM.OpenSubKey('SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\AppRecovery
Agent 5')
$regVal = $regLM.GetValue('InstallLocation')
$regVal = $regVal + 'Common.Contracts.dll'
[System.Reflection.Assembly]::LoadFrom($regVal)  |  out-null

# Converting input parameter into specific object
$TransferPostscriptParameterObject = $TransferPostscriptParameter -as
[Replay.Common.Contracts.PowerShellExecution.TransferPostscriptParameter];

# Working with input object. All echo's are logged
if($TransferPostscriptParameterObject -eq $null) {

     echo 'TransferPostscriptParameterObject parameter is null'
}
else {
echo 'VolumeNames:' $TransferPostscriptParameterObject.VolumeNames

          echo 'ShadowCopyType:' $TransferPostscriptParameterObject.ShadowCopyType

     echo 'ForceBaseImage:' $TransferPostscriptParameterObject.ForceBaseImage

     echo 'IsLogTruncation:' $TransferPostscriptParameterObject.IsLogTruncation

}
```

PreExportScript.ps1

The PreExportScript is run on the Core before any export job.

Sample PreExportScript

```
# receiving parameter from export job
param([object]$ExportJobRequest)

# building path to Core's Common.Contracts.dll and loading this assembly
$regLM = [Microsoft.Win32.Registry]::LocalMachine
$regLM =
$regLM.OpenSubKey('SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\AppRecovery
Core 5')
$regVal = $regLM.GetValue('InstallLocation')
$regVal = $regVal + 'CoreService\Common.Contracts.dll'
[System.Reflection.Assembly]::LoadFrom($regVal)  |  out-null

# Converting input parameter into specific object
$ExportJobRequestObject = $ExportJobRequest -as
[Replay.Core.Contracts.Export.ExportJobRequest]

# Working with input object. All echo's are logged
if($ExportJobRequestObject -eq $null) {

     echo 'ExportJobRequestObject parameter is null'
}
else {

     echo 'Location:' $ExportJobRequestObject.Location

     echo 'Priority:' $ExportJobRequestObject.StorageConfiguration
}
```

PostExportScript.ps1

The PostExportScript is run on the Core after any export job.

NOTE: There are no input parameters for the PostExportScript when used to run once on the exported protected machine after initial startup. The regular protected machine should contain this script in the PowerShell script folder as PostExportScript.ps1.

Sample PostExportScript

```
# receiving parameter from export job
param([object]$ExportJobRequest)

# building path to Core's Common.Contracts.dll and loading this assembly
$regLM = [Microsoft.Win32.Registry]::LocalMachine
$regLM =
$regLM.OpenSubKey('SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\AppRecovery
Core 5')
$regVal = $regLM.GetValue('InstallLocation')
$regVal = $regVal + 'CoreService\Common.Contracts.dll'
[System.Reflection.Assembly]::LoadFrom($regVal)  |  out-null
$regVal2 = $regLM.GetValue('InstallLocation')
$regVal2 = $regVal2 + 'CoreService\Common.Contracts.dll'

# Converting input parameter into specific object
$ExportJobRequestObject = $ExportJobRequest -as
[Replay.Core.Contracts.Export.ExportJobRequest]
```

```
# Working with input object. All echo's are logged
if($ExportJobRequestObject -eq $null) {
     echo 'ExportJobRequestObject parameter is null'
}
else {
     echo 'VolumeImageIds:' $ExportJobRequestObject.VolumeImageIds
     echo 'RamInMegabytes:' $ExportJobRequestObject.RamInMegabytes
}
```

PreNightlyJobScript.ps1

The PreNightlyJobScript is run before every nighty job on Core side. It contains the parameter $JobClassName, which helps to handle those child jobs separately.

Sample PreNightlyJobScript

```
# receiving parameters from Nightlyjob
param([System.String]$JobClassMethod , [object]$NightlyAttachabilityJobRequest,
[object]$RollupJobRequest, [object]$Agents, [object]$ChecksumCheckJobRequest,
[object]$TransferJobRequest, [int]$LatestEpochSeenByCore)

# building path to Core's Common.Contracts.dll and loading this assembly
$regLM = [Microsoft.Win32.Registry]::LocalMachine
$regLM =
$regLM.OpenSubKey('SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\AppRecovery
Core 5')
$regVal = $regLM.GetValue('InstallLocation')
$regVal = $regVal + 'CoreService\Common.Contracts.dll'
[System.Reflection.Assembly]::LoadFrom($regVal)  |  out-null

# Nightlyjob has four child jobs: NightlyAttachability Job, Rollup Job, Checksum
Check Job and Log Truncation Job. All of them are triggering the script, and
$JobClassMethod (contain job name that calls the script) helps to handle those child
jobs separately
switch ($JobClassMethod) {

# working with NightlyAttachability Job
   NightlyAttachabilityJob {
      $NightlyAttachabilityJobRequestObject = $NightlyAttachabilityJobRequest -as
      [Replay.Core.Contracts.Sql.NightlyAttachabilityJobRequest];
      echo 'Nightly Attachability job results:';
            if($NightlyAttachabilityJobRequestObject -eq $null) {
                  echo 'NightlyAttachabilityJobRequestObject parameter is null';
            }
            else {
                  echo 'AgentId:' $NightlyAttachabilityJobRequestObject.AgentId;
                  echo 'IsNightlyJob:'
                  $NightlyAttachabilityJobRequestObject.IsNightlyJob;
            }
            break;
      }
# working with Rollup Job
```

```
   RollupJob {
      $RollupJobRequestObject = $RollupJobRequest -as
      [Replay.Core.Contracts.Rollup.RollupJobRequest];
      echo 'Rollup job results:';
            if($RollupJobRequestObject -eq $null) {
                        echo 'RollupJobRequestObject parameter is null';
            }
            else {
                  echo 'SimultaneousJobsCount:'
                  $RollupJobRequestObject.SimultaneousJobsCount;
                        echo 'AgentId:' $RollupJobRequestObject.AgentId;
                        echo 'IsNightlyJob:' $RollupJobRequestObject.IsNightlyJob;
            }
            $AgentsCollection = $Agents -as
            "System.Collections.Generic.List``1[System.Guid]"
            if($AgentsCollection -eq $null) {
                  echo 'AgentsCollection parameter is null';
            }
            else {
                  echo 'Agents GUIDs:'
                  foreach ($a in $AgentsCollection) {
                        echo $a
            }
      }
      break;
   }
# working with Checksum Check Job
      ChecksumCheckJob {
            $ChecksumCheckJobRequestObject = $ChecksumCheckJobRequest -as
            [Replay.Core.Contracts.Exchange.ChecksumChecks.ChecksumCheckJobRequest];
            echo 'Exchange checksumcheck job results:';
            if($ChecksumCheckJobRequestObject -eq $null) {
                  echo 'ChecksumCheckJobRequestObject parameter is null';
      }
      else {
                  echo 'RecoveryPointId:'
                  $ChecksumCheckJobRequestObject.RecoveryPointId;
                  echo 'AgentId:' $ChecksumCheckJobRequestObject.AgentId;
                  echo 'IsNightlyJob:' $ChecksumCheckJobRequestObject.IsNightlyJob;
            }
            break;
      }
```

```
# working with Log Truncation Job
   TransferJob {

      $TransferJobRequestObject = $TransferJobRequest -as
      [Replay.Core.Contracts.Transfer.TransferJobRequest];

      echo 'Transfer job results:';

      if($TransferJobRequestObject -eq $null) {

                echo 'TransferJobRequestObject parameter is null';

      }

      else {

                echo 'TransferConfiguration:'
                $TransferJobRequestObject.TransferConfiguration;

                echo 'StorageConfiguration:'
                $TransferJobRequestObject.StorageConfiguration;

           }

           echo 'LatestEpochSeenByCore:' $LatestEpochSeenByCore;

           break;

      }
}
```

PostNightlyJobScript.ps1

The PostNightlyJobScript is run after every nighty job on the Core. It contains the parameter $JobClassName, which helps to handle those child jobs separately.

Sample PostNightlyJobScript

```
# receiving parameters from Nightlyjob
param([System.String]$JobClassMethod , [object]$NightlyAttachabilityJobRequest,
[object]$RollupJobRequest, [object]$Agents, [object]$ChecksumCheckJobRequest,
[object]$TransferJobRequest, [int]$LatestEpochSeenByCore,
[object]$TakeSnapshotResponse)

# building path to Core's Common.Contracts.dll and loading this assembly
$regLM = [Microsoft.Win32.Registry]::LocalMachine
$regLM =
$regLM.OpenSubKey('SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\AppRecovery
Core 5')
$regVal = $regLM.GetValue('InstallLocation')
$regVal = $regVal + 'CoreService\Common.Contracts.dll'
[System.Reflection.Assembly]::LoadFrom($regVal)  |  out-null
$regVal2 = $regLM.GetValue('InstallLocation')
$regVal2= $regVal2 + 'CoreService\Core.Contracts.dll'
[System.Reflection.Assembly]::LoadFrom($regVal2) | out-null

# Nightlyjob has four child jobs: NightlyAttachability Job, Rollup Job, Checksum
Check Job and Log Truncation Job. All of them are triggering the script, and
$JobClassMethod (contain job name that calls the script) helps to handle those child
jobs separately
switch ($JobClassMethod) {

# working with NightlyAttachability Job
   NightlyAttachabilityJob {
```

```
      $NightlyAttachabilityJobRequestObject = $NightlyAttachabilityJobRequest -as
      [Replay.Core.Contracts.Sql.NightlyAttachabilityJobRequest];

      echo 'Nightly Attachability job results:';

      if($NightlyAttachabilityJobRequestObject -eq $null) {

            echo 'NightlyAttachabilityJobRequestObject parameter is null';

      }

      else {

            echo 'AgentId:' $NightlyAttachabilityJobRequestObject.AgentId;

            echo 'IsNightlyJob:' $NightlyAttachabilityJobRequestObject.IsNightlyJob;

      }

      break;
   }
# working with Rollup Job
   RollupJob {

      $RollupJobRequestObject = $RollupJobRequest -as
      [Replay.Core.Contracts.Rollup.RollupJobRequest];

      echo 'Rollup job results:';

      if($RollupJobRequestObject -eq $null) {

            echo 'RollupJobRequestObject parameter is null';

      }

      else {

            echo 'SimultaneousJobsCount:'
            $RollupJobRequestObject.SimultaneousJobsCount;

            echo 'AgentId:' $RollupJobRequestObject.AgentId;

            echo 'IsNightlyJob:' $RollupJobRequestObject.IsNightlyJob;

      }

      $AgentsCollection = $Agents -as
      "System.Collections.Generic.List``1[System.Guid]"

      if($AgentsCollection -eq $null) {

            echo 'AgentsCollection parameter is null';

      }

      else {

      echo 'Agents GUIDs:'

            foreach ($a in $AgentsCollection) {

                  echo $a

                  }

            }

            break;

      }
# working with Checksum Check Job
   ChecksumCheckJob {
```

```
      $ChecksumCheckJobRequestObject = $ChecksumCheckJobRequest -as
      [Replay.Core.Contracts.Exchange.ChecksumChecks.ChecksumCheckJobRequest];

      echo 'Exchange checksumcheck job results:';

      if($ChecksumCheckJobRequestObject -eq $null) {

            echo 'ChecksumCheckJobRequestObject parameter is null';

      }

      else {

            echo 'RecoveryPointId:' $ChecksumCheckJobRequestObject.RecoveryPointId;

            echo 'AgentId:' $ChecksumCheckJobRequestObject.AgentId;

            echo 'IsNightlyJob:' $ChecksumCheckJobRequestObject.IsNightlyJob;

      }

      break;
   }
# working with Log Truncation Job
   TransferJob {

      $TransferJobRequestObject = $TransferJobRequest -as
      [Replay.Core.Contracts.Transfer.TransferJobRequest];

      echo 'Transfer job results:';

      if($TransferJobRequestObject -eq $null) {

            echo 'TransferJobRequestObject parameter is null';

      }

      else {

             echo 'TransferConfiguration:'
            $TransferJobRequestObject.TransferConfiguration;

            echo 'StorageConfiguration:'
            $TransferJobRequestObject.StorageConfiguration;

      }

      echo 'LatestEpochSeenByCore:' $LatestEpochSeenByCore;

      $TakeSnapshotResponseObject = $TakeSnapshotResponse -as
      [Replay.Agent.Contracts.Transfer.TakeSnapshotResponse];

      if($TakeSnapshotResponseObject -eq $null) {

            echo 'TakeSnapshotResponseObject parameter is null';

      }

      else {

            echo 'ID of this transfer session:' $TakeSnapshotResponseObject.Id;

            echo 'Volumes:' $TakeSnapshotResponseObject.Volumes;

      }

      break;
   }
}
```

Using Bourne Shell scripting in AppAssure

Bourne shell (sh) is a shell language or command-line interpreter for Unix-based operating systems. Bourne shell is used in AppAssure with Linux to customize environments and specify certain operations to occur in a predetermined sequence. The .sh is the file extension and naming convention for Bourne shell files.

Bourne Again Shell (BASH) is a similar shell language that implements the same grammar, parameter, and variable expansion, redirection and quoting. BASH also uses the same .sh file extension. The information here applies equally to BASH.

Using pre and post transfer and export script hooks, you can perform system operations before and after a transfer or export. For example, you may want to disable a certain cronjob while a transfer is occurring and enable it once the transfer has finished. As another example, you may need to run commands to flush application-specific data to disk. The contents are written to a temporary file and run using exec. The script then runs using the interpreter defined in the first line of the script, for example, `(#!/usr/bin/env bash)`. If the specified interpreter is not available, the script uses the default shell defined in the $SHELL environment variable.

You can substitute and use any interpreter. For example, on the `#!` line of the script, you can replace “bash” with “zsh” (Z shell), “tcsh” (tee shell), and so on, based on your preference.

You can add available objects from the TransferPrescript parameter or add your own commands to the PreTransferScript.sh and PostTransfer.sh scripts to customize them.

This section describes the scripts that can be used by administrators at designated occurrences in AppAssure for Windows and Linux. It includes the following topics:

- Input parameters for Bourne Shell scripting
- Sample Bourne Shell scripts

Prerequisites for Bourne Shell scripting

All scripts must be named PreTransferScript.sh, PostTransfer.sh, and PostExportScript.sh. Additionally, all scripts must reside in the `/opt/appassure/scripts/` directory.

Testing Bourne Shell scripting

You can test the scripts you want to run by using the editor for the script (.sh) files.

NOTE: If the pre-Bourne Shell or post-Bourne Shell scripts fail, the job also fails. Information about the job is available in the /var/log/appassure/appassure.log file.

Successful scripts return the exit code 0.

Input parameters for Bourne Shell scripting

The parameters for Bourne Shell scripting in AppAssure are described in the following tables.

TransferPrescriptParameters_VolumeNames

The following table presents the available objects for the TransferPrescript parameter.

Table 219. TransferPrescript objects

Method	Description
public VolumeNameCollection VolumeNames (get; set;)	Gets or sets the collection of volume names for transfer. VolumeNames is a data structure that contains the following data: • **GuidName**. The Guid associated with the volume, used as the name if a DisplayName is not set. • **DisplayName**. The displayed name of the volume.
public ShadowCopyType ShadowCopyType { get; set; }	Gets or sets the type of copying for transfer. ShadowCopyType is an enumeration with values. The available values are: • Unknown • Copy • Full
public string Key { get; set; }	The Key method generates a pseudorandom (but not cryptographically secure) key, which can be used as a one-time password to authenticate transfer requests.
public bool ForceBaseImage { get; set; }	Gets or sets the value indicating whether the transfer was a forced base image capture.
public bool IsLogTruncation { get; set; }	Gets or sets the value indicating whether logging is being truncated.
public uint LatestEpochSeenByCore { get; set; }	Gets or sets latest epoch value. The LatestEpochSeenByCore method is the ordinal number of the most recent snapshot taken by the Core. This is the 'epoch number' assigned by the filter driver to this particular snapshot at the moment it was taken with VSS.

TransferPostscriptParameter

The following table presents the available objects for the TransferPostscript parameter.

Table 220. TransferPostscript objects

Method	Description
public VolumeNameCollection VolumeNames (get; set;)	Gets or sets the collection of volume names for transfer. VolumeNames is a data structure that contains the following data: • **GuidName**. The Guid associated with the volume, used as the name if a DisplayName is not set. • **DisplayName**. The displayed name of the volume.
public ShadowCopyType ShadowCopyType { get; set; }	Gets or sets the type of copying for transfer.ShadowCopyType is an enumeration with values. The available values are: • Unknown • Copy • Full
public string Key { get; set; }	The Key method generates a pseudorandom (but not cryptographically secure) key, which can be used as a one-time password to authenticate transfer requests.
public bool ForceBaseImage { get; set; }	Gets or sets the value indicating whether the transfer was a forced base image capture.
public bool IsLogTruncation { get; set; }	Gets or sets the value indicating whether logging is being truncated.
public uint LatestEpochSeenByCore { get; set; }	Gets or sets latest epoch value. The LatestEpochSeenByCore method is the ordinal number of the most recent snapshot taken by the Core. This is the 'epoch number' assigned by the filter driver to this particular snapshot at the moment it was taken with VSS.

Sample Bourne Shell scripts

This section describes the sample Bourne Shell scripts available for administrative users to run on protected machines.

> **CAUTION:** **The sample Bourne scripts provided in this document will function when run as designed by qualified administrators. Take precautions when modifying functioning scripts to retain working versions. Any modifications to the script samples included here, or any scripts you create, are considered customization, which is not typically covered by Customer Support.**

The sample scripts for protected machines include:

- PreTransferScript.sh
- PostTransferScript.sh
- PostExportScript.sh

> **NOTE:** Protected machines use the 'exec' shell command to launch the script. You can indicate which interpreter should run the script by defining that information in the first line of the script. If you don't specify the interpreter, the default shell interprets the script. If you choose something other than the default shell, you must ensure that the specified interpreter is available on all protected machines.

PreTransferScript.sh

The PreTransferScript is run on the protected machine before transferring a snapshot.

The following script stores the values from input parameters in the Pre(Post)TransferScriptResult.txt, which is located and stored in the root home directory.

Sample PreTransferScript

```
#!/bin/bash
echo
"TransferPrescriptParameter_VolumeNames=$TransferPrescriptParameter_VolumeNames
TransferPrescriptParameter_ShadowCopyType=$TransferPrescriptParameter_ShadowCopyTyp
e
TransferPrescriptParameter_Key=$TransferPrescriptParameter_Key
TransferPrescriptParameter_ForceBaseImage=$TransferPrescriptParameter_ForceBaseImag
e
TransferPrescriptParameter_IsLogTruncation=$TransferPrescriptParameter_IsLogTruncat
ion
TransferPrescriptParameter_LatestEpochSeenByCore=$TransferPrescriptParameter_Latest
EpochSeenByCore" > ~/PreTransferScriptResult.txt
exit 0
```

PostTransferScript.sh

The PostTransferScript is run on the protected machine after transferring a snapshot.

The following script stores the values from input parameters in the Pre(Post)TransferScriptResult.txt, which is located and stored in the root home directory.

Sample PostTransferScript

```
#!/bin/bash
echo
"TransferPostscriptParameter_VolumeNames=$TransferPostscriptParameter_VolumeNames
TransferPostscriptParameter_ShadowCopyType=$TransferPostscriptParameter_ShadowCopyT
ype
TransferPostscriptParameter_Key=$TransferPostscriptParameter_Key
TransferPostscriptParameter_ForceBaseImage=$TransferPostscriptParameter_ForceBaseIm
age
TransferPostscriptParameter_IsLogTruncation=$TransferPostscriptParameter_IsLogTrunc
ation
TransferPostscriptParameter_LatestEpochSeenByCore=$TransferPostscriptParameter_Late
stEpochSeenByCore" > ~/PostTransferScriptResult.txt
exit 0
```

PostExportScript.sh

The PostExportScript is run on the protected machine after the transfer.

The following script stores the values from input parameters in the Pre(Post)ExportScriptResult.txt, which is located and stored in the root home directory.

Sample PostExportScript

```
#!/bin/bash
echo
"$curr_name-exported" > /etc/hostname
exit 0
```

D

Understanding AppAssure APIs

The purpose of this section is to provide an introduction and overview of the AppAssure REST APIs, their use and function.

The AppAssure Web service APIs are RESTful and allow you to automate and customize certain functions and tasks within the AppAssure software solution to assist you with meeting your business objectives.

This document describes the AppAssure Core and Agent API references and lists the service contracts and URI for each category.

See either of the following sections for information regarding the service contracts that are included in the Core and Agent client assemblies:

- Using AppAssure Core API
- Using AppAssure Agent API

Intended Audience

AppAssure APIs are intended for use by application developers who want to integrate and extend AppAssure in their application as well as administrators who want to script interactions with the AppAssure Core Server.

Recommended additional reading

- *Dell AppAssure Installation and Upgrade Guide*. This guide provides an overview of the AppAssure architecture and features and describes the steps and instruction necessary for installing the AppAssure components, as well as for migrating from Replay to AppAssure.

You can download this guide from https://support.software.dell.com/appassure/5.4.3/release-notes-guides/.

Working with AppAssure REST APIs

The AppAssure APIs are REST-style APIs, which means that they use HTTP requests to provide access to resources (data entities) through URI paths. AppAssure APIs use standard HTTP methods such as GET, PUT, POST, and DELETE. Since REST APIs are based on open standards, you can use any language or tool that supports HTTP calls.

There are two ways that application developers and administrators can work with AppAssure APIs. They are:

- Using C# or other .NET languages to directly use AppAssure .NET client DLL files.
- Communicate directly with the HTTP endpoint to generate your own XML.

The first approach is recommended. The client DLLs are available in the AppAssure Core installation directory. The method for calling AppAssure APIs is consistent with the way you would consume any .NET 4.0 WCF service.

The main DLL is Core.Client.dll. Core.Contracts.dll, Agent.Contracts.dll and any other *.Contract.dll can be used as needed to support what you are trying to accomplish with AppAssure.

Using AppAssure Core API

The AppAssure Core API references are grouped according to the following categories and service interfaces. Click a category to get more information about the service operations available for a particular interface.

The service contracts contained in the Core client assembly are described in the following table:

Table 1. Core service contracts

Service Contract	URI	Description
IAgentDiagnosticsManagement	agentdiag/	Replay.Core.Contracts.Agents.IAgentDiagnosticsManagement
IAgentsGroupsManagement	agentGroups/	Interface implemented by the groups service, which performs actions on the core related to the group.
IAgentsManagement	agents/	Interface implemented by the agents service, which performs actions on the core related to the agent.
IApplianceEventsManagement	events/appliance/	Exposes the appliance events for the Core.
IApplicationIdManagement	id/	Exposes the unique ID for the application.
IAutoUpdateManagement	autoupdate/	Replay.Core.Contracts.AutomaticUpdate.IAutoUpdateManagement
IBackgroundJobManagement	jobmgr/	WCF contract interface that is implemented by a class which implements the management interface on top of the background job manager.
IBackupManagement	backup/	Interface implemented by the Core backup management service, that is used for backing up and restoring a local core.
IBootCdBuilderManagement	bootcdbuilder/	Replay.Core.Contracts.BootCdBuilder.IBootCdBuilderManagement
ICloudManagement	cloud/	Exposes the cloud management API.
IClustersManagement	clusters/	Exposes the cluster management API.
ICoreBackupRestoreManagement	corebackuprestore/	Interface that is implemented by the core configuration backup and restore management service which is used to perform a backup and restore of the Core configuration.
ICoreCallbackManagement	corecallback/	Interface that is implemented by the core callback management service, which is used to by remote agents.
ICoreDiagnosticsManagement	corediag/	Replay.Core.Contracts.Diagnostics.ICoreDiagnosticsManagement
ICoreMetadataManagement	metadata/	Interface that is implemented by the core metadata service, which maintains, caches, and returns core metadata.
ICoreSettingsManagement	settings/	Provides a way to query and set assorted core-wide settings.
IDatabaseStorageManagement	databaseStorage/	Interface for the communication and configuration of persistent database storage that is used for events and reports the storing of information.
IDiagnosticsManagement	diag/	Replay.Common.Contracts.Diagnostics.IDiagnosticsManagement

Table 1. Core service contracts

Service Contract	URI	Description
IEmailsManagement	emailsmgr/	WCF contract interface for the management of email.
IEncryptionKeyManagement	encryption/	Encryption key management that is used with repositories and dependent services.
IEventsManagement	events/	Exposes the events and alerts on the Core.
IExchangeManagement	exchange/	Exposes the MS Exchange management on the Core.
IExportQueueManagement	export/queue/	RESTful API for the export queue.
IExportSchedulerManagement	export/schedule/	RESTful API for the export scheduler.
IHyperVAgentManagement	hypervagent/	Replay.Common.Contracts.Virtualization.IHyperVAgentManagement
IIsoDatabaseManagement	bootcdbuilder/	Replay.Core.Contracts.BootCdBuilder.IIsoDatabaseManagement
ILicenseManagement	license/	Interface that is implemented by the core license management service, which provides license information and functionality.
ILocalizationManagement	localization/	Interface that is implemented by the common localization service with a persistent configuration, which maintains, set and gets product's current culture.
ILocalMountManagement	mounts/	Interface that is implemented by the local mounts management service, used to mount and un-mount volume images on the local core.
ILoggingManagement	logs/	Replay.Core.Contracts.Logging.ILoggingManagement
ILogTruncationManagement	logtruncation/	Interface that is implemented by core log truncation management service which provides log truncation functionality.
INightlyJobsManagement	nightlyJobs/	RESTful API for nightly jobs.
IProtectedItemsManagement	protectedItems/	Exposes protected items such as agents, clusters, cluster nodes and groups as tree.
IPushInstallCommunication	pushinstallcomm/	API used internally to communicate between core and PushInstall agent. NOTE: Do not use this API directly.
IPushInstallManagement	pushinstall/	Interface implements the ability to install an agent on remote machine. Push install is a functionality that allows user to initiate and control agent installation on remote environments in a network.
IRecoveryPointsManagement	recoveryPoints/	Interface that is implemented by the recovery points management service, which exposes information about recovery points to REST clients.
IRemoteMountManagement	remoteMounts/	Interface that is implemented by the remote mounts management service, used to disconnect mounted volume images on the remote machines.
IReplayEngineManagement	replayEngine/	Interface that is implemented by the Replay engine management service, which is used to configure and monitor the Replay engine.

Table 1. Core service contracts

Service Contract	URI	Description
IReplicationCommunication	replicationcomms/	API used internally to communicate between cores for replication. NOTE: Do not use this API directly.
IReplicationManagement	replication/	API for the configuration, management, and monitoring of replication.
IReportingManagement	report/	Interface that is implemented by the core reports management service, which provides reporting functionality.
IRepositoryManagement	reposManagement/	Interface that is implemented by the repository management service, which handles core operations related to the repositories.
IRollbackManagement	rollback/	Interface that is implemented by the core rollback service, which provides functionality from the Replay engine to restore from a recovery point.
IRollupManagement	rollup/	Interface implemented by the core rollup management service, which provides rollup functionality from the Replay engine.
ISeedDriveManagement	seedDrive/	Interface that is implemented by the seed drive management service, which provides copy-consume functionality from the Replay engine.
IServiceHostManagement	servicehost/	Interface that is implemented by a class which provides a management interface to the IServiceHost.
ISqlManagement	sql/	Interface that is implemented by the core SQL management service which provides attachability checks and other core-specific SQL management functionality
IStatusSummaryManagement	status/	Exposes summarized status information about multiple services in the core that are used for displaying status icons in the GUI tabs.
ITransferQueueManagement	xfer/queue/	RESTful API for the transfer queue.
ITransferSchedulerManagement	xfer/schedule/	RESTful API for the transfer scheduler.
IUtilitiesManagement	utilities/	API for a variety of helpers.
IVirtualDiskManagement	vhd/	Replay.Common.Contracts.Virtualization.IVirtualDiskManagement
IWhiteLabelingManagement	whitelabeling/	Exposes the customizable strings.

IAgentDiagnosticsManagement

This section describes the service operations available for IAgentDiagnosticsManagement at agentdiag/. The URI and HTTP method are provided for each service operation.

The service operations include:

- ExecuteRemoteCommand
- GetLog
- RestartService
- UploadLogSessions

ExecuteRemoteCommand

Runs an arbitrary remote command.

URI: agentdiag/{agentId}/command/

HTTP Method: POST

GetLog

Gets the entire contents of the replay.log file.

URI: agentdiag/{agentId}/log/

HTTP Method: GET

RestartService

Stops, forcibly kills (if necessary), and re-starts the service.

URI: agentdiag/{agentId}/service/

HTTP Method: DELETE

UploadLogSessions

Uploads the current log session to Gibraltar (http://www.gibraltarsoftware.com/).

URI: agentdiag/{agentId}/log/

HTTP Method: POST

IAgentsGroupsManagement

This section describes the service operations available for IAgentsGroupsManagement at agentGroups/. The IAgentsGroupsManagement service contract is the interface that is implemented by the groups service, which performs actions on the Core related to the group.

The URI and HTTP method are provided for each service operation.

The service operations include:

- AddAgent
- AddGroup
- ChangeName
- GetGroups
- RemoveAgent
- RemoveGroup

AddAgent

Adds protected agent to group.

URI: agentGroups/{groupId}/{agentId}

HTTP Method: POST

AddGroup

Adds a new group.

URI: agentGroups/

HTTP Method: POST

ChangeName

Changes name of group.

URI: agentGroups/{groupId}/{newName}

HTTP Method: PUT

GetGroups

Gets a list of all groups known to the core.

URI: agentGroups/

HTTP Method: GET

RemoveAgent

Removes agent from group.

URI: agentGroups/{groupId}/{agentId}

HTTP Method: DELETE

RemoveGroup

Removes group from core.

URI: agentId}agentGroups/{groupId}

HTTP Method: DELETE

IAgentsManagement

This section describes the service operations available for IAgentsManagement at agents/. The IAgentsManagement service contract is the interface that is implemented by the agents service, which performs actions on the Core related to the agent.

The URI and HTTP method are provided for each service operation.

The service operations include:

- AddAgent
- AddAgents
- AddAgentWithVolumeGroupsValidation
- ChangeAgentDescriptor
- ChangeDisplayName
- ChangeHostName
- ChangePort

- CheckAgentPairing
- DeleteAgent
- DeleteAgents
- FindAgentById
- ForceAgentMetadataRefresh
- GetAgentDetails
- GetAgentInfo
- GetAgentInfoSummaries
- GetAgentMetadata
- GetAgentMetadataById
- GetAgentMetadataCredentials
- GetAgents
- GetAgentSummaryInfo
- GetAgentSummaryMetadatadata/sum
- GetAgentVolumeGroupsAvailableForProtection
- GetAgentVolumesAvailableForProtection
- GetCachedAgentMetadataById
- GetCachedAgentSummaryMetadataById
- GetPairingSettings
- GetProtectedAgents
- GetProtectedAgentsSummaries
- GetRecoveryPointsOnlyAgents
- GetRecoveryPointsOnlyAgentsSummaries
- GetReplicatedAgents
- GetReplicatedAgentsSummaries
- GetWriters
- IsClusterAddress
- RepairPairing
- SetAgentMetadataCredentials
- SetReplicatedAgentRepository
- ValidateAgentCreationParameters
- ValidateAgentCreationParametersForDeploy
- ValidateAgentProtectionAbility
- VerifyAgentVolumesAreNotInConflictWithRepositoryPaths
- VerifyConnect

AddAgent

Adds an agent that is not currently protected, pairs with the agent and writes the protection configuration information.

URI: agents/new

HTTP Method: POST

AddAgents

Adds the specified agents that are not currently protected. The agents are paired with and their protection configuration is written.

URI: agents/newagents

HTTP Method: POST

AddAgentWithVolumeGroupsValidation

Adds an agent that is not currently protected, pairs with the agent and writes the protection configuration information and then validates protection groups.

URI: agents/newWithValidGroups

HTTP Method: POST

ChangeAgentDescriptor

Changes the descriptor that is used for an existing agent, allowing the core to use a new caller-specified URI and credentials.

URI: agents/{agentId}/descriptor

HTTP Method: POST

ChangeDisplayName

Changes the display name used by the GUI for a given agent.

URI: agents/{agentId}/changeDisplayName/{newDisplayName}

HTTP Method: POST

ChangeHostName

Changes the host name that is used in connecting to a given agent.

URI: agents/{agentId}/changeHostName/{newHostName}

HTTP Method: POST

ChangePort

Changes the port that is used in connecting to a given agent.

URI: agents/{agentId}/port

HTTP Method: POST

CheckAgentPairing

Connects to an agent that is specified in the descriptor (which is probably not currently protected by the core) and checks to determine if the agent is paired to the core or any core.

URI: agents/query/checkpairing

HTTP Method: PUT

DeleteAgent

Deletes an agent from the core, optionally deletes the recovery points for the agent and also disables the volumes.

URI: agents/{agentId}/delete

HTTP Method: POST

DeleteAgents

Deletes agents from the core, optionally deletes the recovery points for the agent and also disables the volumes.

URI: agents/delete

HTTP Method: POST

FindAgentById

Finds the information about a protected agent, if it exists.

URI: agents/find/{agentId}

HTTP Method: GET

ForceAgentMetadataRefresh

Forces a metadata refresh for an agent with a specified ID.

URI: agents/{agentId}/metadata/refresh

HTTP Method: POST

GetAgentDetails

Gets all information that is required to display the Agent Details page in the GUI.

URI: agents/{agentId}/details

HTTP Method: GET

GetAgentInfo

Gets information about the protected agent.

URI: agents/{agentId}/info

HTTP Method: GET

GetAgentInfoSummaries

Gets a list of info summary data contracts for each (protected, replicated and replay only) agent. This is an extremely lightweight method that does not perform any metadata-related REST calls and provides very limited information.

URI: agents/infosummaries

HTTP Method: GET

GetAgentMetadata

Connects to an agent (which is probably not currently being protected by the Core) that is specified in the descriptor and requests the metadata for the agent.

URI: agents/query/metadata

HTTP Method: PUT

GetAgentMetadataById

Gets the latest metadata for a protected agent.

URI: agents/{agentId}/metadata

HTTP Method: GET

GetAgentMetadataCredentials

Gets credentials which are used for retrieving the metadata for the agent with a specific ID.

URI: agents/{agentId}/metadata/credentials

HTTP Method: GET

GetAgents

Gets a list of all agents known to the core.

URI: agents/

HTTP Method: GET

GetAgentSummaryInfo

Gets summary agent information including summary metadata, recent associated alerts, and recent recovery points.

URI: agents/{agentId}/summaryInfo

HTTP Method: GET

GetAgentSummaryMetadataById

Gets latest summary metadata for an agent that is protected by the Core.

URI: agents/{agentId}/metadata/summary

HTTP Method: GET

GetAgentSummaryMetadatadata/sum

Connects to an agent (which is probably not currently being protected by the Core) that is specified in the descriptor and requests the summary metadata for the agent.

URI: agents/query/summarymetadata

HTTP Method: PUT

GetAgentVolumeGroupsAvailableForProtection

Connects to an agent (which is probably not currently being protected by the Core) that is specified in the descriptor and requests the list of volumes which are available for protection, and grouped to reflect protection dependencies. It is possible that an agent and a core are installed on the same machine so a resulting list would not contain volumes which contain core repository data or metadata directories. For regular cases this list would be identical to a list of volumes returned in metadata.

URI: agents/query/availableGroups

HTTP Method: PUT

GetAgentVolumesAvailableForProtection

Connects to an agent specified in the descriptor (probably not currently protected by this core) and requests the list of volumes which are available for protection. It is possible that an agent and a core are installed on the same machine so a resulting list would not contain volumes which contain core repository data or metadata directories. For regular cases this list would be identical to a list of volumes returned in metadata.

URI: agents/query/availablevolumes

HTTP Method: PUT

GetCachedAgentMetadataById

Gets cached metadata for an agent protected by Replay.

URI: agents/{agentId}/cachedmetadata

HTTP Method: GET

GetCachedAgentSummaryMetadataById

Gets cached summary metadata for an agent protected by Replay.

URI: agents/{agentId}/cachedmetadata/summary

HTTP Method: GET

GetPairingSettings

Connects to an agent specified in the descriptor (probably not currently protected by this core) and requests the agent's pairing information.

URI: agents/query/pairing

HTTP Method: PUT

GetProtectedAgents

Gets a list of protected agents known to the core.

URI: agents/protected

HTTP Method: GET

GetProtectedAgentsSummaries

Gets a list of protected agents summaries known to the core.

URI: agents/protectedSummaries

HTTP Method: GET

GetRecoveryPointsOnlyAgents

Gets a list of agent-only recovery points.

URI: agents/recoveryPointsOnly

HTTP Method: GET

GetRecoveryPointsOnlyAgentsSummaries

Gets a summary list of agent-only recover points.

URI: agents/recoveryPointsOnlySummaries

HTTP Method: GET

GetReplicatedAgents

Gets a list of replicated agents known to the core.

URI: agents/replicated

HTTP Method: GET

GetReplicatedAgentsSummaries

Gets a list of replicated agents summaries known to the core.

URI: agents/replicatedSummaries

HTTP Method: GET

GetWriters

Gets detailed information about VSS writers installed on the agent.

URI: agents/{agentId}/writers

HTTP Method: GET

IsClusterAddress

Verify if provided address is cluster.

URI: agents/verifyClusterAddress

HTTP Method: POST

RepairPairing

Attempts to re-establish pairing with an agent already protected by the core, possibly using new credentials.

URI: agents/pairing/repair

HTTP Method: POST

SetAgentMetadataCredentials

Sets credentials which used for metadata retrieval for agent with specified Id.

URI: agents/{agentId}/metadata/credentials

HTTP Method: POST

SetReplicatedAgentRepository

Assigns a repository to an agent which was added to replication.

URI: agents/{agentId}/changeRepository/{repositoryId}

HTTP Method: POST

ValidateAgentCreationParameters

Validate agent creation parameters.

URI: agents/validateAgentCreationParameters

HTTP Method: POST

ValidateAgentCreationParametersForDeploy

Validate agent creation parameters for deploy.

URI: agents/validateAgentCreationParametersForDeploy

HTTP Method: POST

ValidateAgentProtectionAbility

Verify if the Agent can be protected.

URI: agents/validateAgentProtectionAbility

HTTP Method: POST

VerifyAgentVolumesAreNotInConflictWithRepositoryPaths

Verifies whether volumes, selected for protection are not in a conflict state with any of the specified repository paths on agent environment.

URI: agents/verifyVolumesConflictState

HTTP Method: POST

VerifyConnect

Tries to connect to the agent and then return the ID.

URI: agents/verifyConnection/?useNtlm={useNtlm}

HTTP Method: POST

IApplianceEventsManagement

This section describes the service operations available for IApplianceEventsManagement at events/appliance/. The IApplianceEventsManagement service contract exposes the appliance events for the Core.

The URI and HTTP method is provided for the service operation:

- SendApplianceEvent

SendApplianceEvent

Posts the passed in event.

URI: events/appliance/raise

HTTP Method: POST

IApplicationIdManagement

This section describes the service operations available for IApplicationIdManagement at id/. The IApplicationIdManagement service contract exposes the application's unique ID.

The URI and HTTP method is provided for the service operation:

- SendApplianceEvent

GetId

Gets the ID for the Core.

URI: id/

HTTP Method: GET

IAutoUpdateManagement

This section describes the service operations available for IAutoUpdateManagement at autoupdate/.

The URI and HTTP method are provided for each service operation.

The service operations include:

- CheckUpdates
- ForceUpgrade
- GetConfiguration
- GetState
- IsCoreServiceStarting
- SetConfiguration

CheckUpdates

Checks for updates.

URI: autoupdate/checkForUpdates

HTTP Method: GET

ForceUpgrade

Sets AutoUpdate configuration.

URI: autoupdate/forceUpgrade

HTTP Method: POST

GetConfiguration

Sets AutoUpdate configuration.

URI: autoupdate/config

HTTP Method: GET

GetState

Runs an upgrade.

URI: autoupdate/updateState

HTTP Method: GET

IsCoreServiceStarting

Returns a value of 'True' if determined that the Core service has restarted.

URI: autoupdate/isCoreServiceRestarting

HTTP Method: GET

SetConfiguration

Gets current update status.

URI: autoupdate/config

HTTP Method: PUT

IBackgroundJobManagement

This section describes the service operations available for IBackgroundJobManagement at jobmgr/.

The IBackgroundJobManagement service contract is the WCF contract interface implemented by a class which implements the management interface on top of the background job manager.

The URI and HTTP method are provided for each service operation.

The service operations include:

- CancelChildJob
- CancelJob
- GetBackgroundJobsConfiguration
- GetJob
- GetJobs
- GetJobsByPage
- GetJobsCount
- ResetBackgroundJobSettings
- UpdateBackgroundJobSettings

CancelChildJob

Cancels the child job.

URI: jobmgr/jobs/{parentJobId}/childJobs/{childJobId}

HTTP Method: DELETE

CancelJob

Cancels the job.

URI: jobmgr/jobs/{jobId}

HTTP Method: DELETE

GetBackgroundJobsConfiguration

Gets background jobs configuration.

URI: jobmgr/config

HTTP Method: GET

GetJob

Gets the specified job.

URI: jobmgr/jobs/{jobId}

HTTP Method: GET

GetJobs

Gets all jobs that pass through the specified filter in both the memory and the database.

URI: jobmgr/jobs/all

HTTP Method: PUT

GetJobsByPage

Gets the current list of jobs that pass through the specified filter which are then presented for ease of viewing by a paged grid.

URI: jobmgr/jobs/paged/{jobsPerPage}/{page}

HTTP Method: PUT

GetJobsCount

Gets the current number of jobs that pass through the specified filter in both the memory and the database.

URI: jobmgr/jobs/count

HTTP Method: PUT

ResetBackgroundJobSettings

Resets the background job setting to the default setting.

URI: jobmgr/resetSettings

HTTP Method: PUT

UpdateBackgroundJobSettings

Performs an update of the background job settings.

URI: jobmgr/settings

HTTP Method: PUT

IBackupManagement

This section describes the service operations available for IBackupManagement at backup/.

The IBackupManagement service contract is the Interface implemented by the core backup management service, used for backing up and restoring a local core.

The URI and HTTP method are provided for each service operation.

The service operations include:

- DeleteScheduledBackup
- GetBackupAgentRecoveryPointsInfo
- GetBackupAgentRecoveryPointsInfoByCore
- GetBackupManifest
- GetBackupManifests
- GetScheduledBackups
- GetWaitingJobs
- PauseScheduledBackup
- ResumeScheduledBackup
- SaveScheduledBackupSettings
- StartBackup
- StartCheckBackup
- StartRestore
- UpdateBackup
- UpdateRestore
- VerifyBackupLocation
- VerifyRestoreContent
- VerifyRestoreLocation

DeleteScheduledBackup

Deletes scheduled archives.

URI: backup/deleteScheduledBackups

HTTP Method: DELETE

GetBackupAgentRecoveryPointsInfo

Gets the metadata for an existing agent backup by agentId.

URI: backup/metadataByCore/agents/{agentId}

HTTP Method: POST

GetBackupAgentRecoveryPointsInfoByCore

Gets the metadata for an existing agent backup by agentId and coreId.
URI: backup/metadataByCore/{requestedCoreId}/agents/{agentId}
HTTP Method: POST

GetBackupManifest

Gets the metadata for an existing core backup by coreId.
URI: backup/metadataByCore/{requestedCoreId}
HTTP Method: POST

GetBackupManifests

Gets all of the metadata for an existing backup.
URI: backup/metadataAll
HTTP Method: POST

GetScheduledBackups

Retrieves settings for all scheduled backups.
URI: backup/scheduledBackups
HTTP Method: GET

GetWaitingJobs

Gets a list of waiting backup and restore jobs.
URI: backup/waiting
HTTP Method: GET

PauseScheduledBackup

Pauses scheduled archives.
URI: backup/pauseScheduledBackups
HTTP Method: POST

ResumeScheduledBackup

Resumes scheduled archives.
URI: backup/resumeScheduledBackups
HTTP Method: POST

SaveScheduledBackupSettings

Saves scheduled archive.
URI: backup/saveScheduledBackupSettings
HTTP Method: POST

StartBackup

Starts a backup.

URI: backup/backup

HTTP Method: POST

StartCheckBackup

Starts a backup's check.

URI: backup/checkBackup

HTTP Method: POST

StartRestore

Starts a restore.

URI: backup/restore

HTTP Method: POST

UpdateBackup

Updates the job request for an existing backup.

URI: backup/backup/update/{jobId}

HTTP Method: POST

UpdateRestore

Updates the job request for an existing restore.

URI: backup/restore/update/{jobId}

HTTP Method: POST

VerifyBackupLocation

Verifies whether the correct file system path is specified as backup location and that the backup can be placed at this path.

URI: backup/verifyBackupLocation

HTTP Method: POST

VerifyRestoreContent

Verifies whether backup content is not in conflict state with currently protected machines.

URI: backup/verifyRestoreContent

HTTP Method: POST

VerifyRestoreLocation

Verifies whether backup job has been completed for specified location and core.

URI: backup/verifyCanStartRestore

HTTP Method: POST

IBootCdBuilderManagement

This section describes the service operations available for IBootCdBuilderManagement at bootcdbuilder/.

The URI and HTTP method are provided for each service operation.

The service operations include:

- GetDefaultOutputPath
- StartBuildingIso
- ValidateDriverPackages
- VerifyBuildingIsoParameters

GetDefaultOutputPath

Gets the default path were resulting ISO image will be put if output path was not specified explicitly.

URI: bootcdbuilder/defaults

HTTP Method: GET

StartBuildingIso

Starts new ISO image building task.

URI: bootcdbuilder/start

HTTP Method: POST

ValidateDriverPackages

Performs validation of specified driver packages.

URI: bootcdbuilder/validate

HTTP Method: POST

VerifyBuildingIsoParameters

Verifies that ISO parameters are built properly.

URI: bootcdbuilder/verify

HTTP Method: POST

ICloudManagement

This section describes the service operations available for ICloudManagement at cloud/. The ICloudManagement service contract exposes the Cloud management API.

The URI and HTTP method are provided for each service operation.

The service operations include:

- AddConfigurationAccount
- DeleteConfigurationAccount
- GetCloudConfiguration
- GetConfigurationAccount
- IsConfigurationAccountUsed

- ListFiles
- ListFolders
- ListItems
- SetCloudConfiguration
- StartDownload
- StartUpload
- UpdateConfigurationAccount
- VerifyConnect

AddConfigurationAccount

Add account to cloud configuration.

URI: cloud/configurationAccount

HTTP Method: POST

DeleteConfigurationAccount

Delete existing account of cloud configuration.

URI: cloud/configurationAccount/{accountId}

HTTP Method: DELETE

GetCloudConfiguration

Gets cloud configuration.

URI: cloud/config

HTTP Method: GET

GetConfigurationAccount

Get existing account of cloud configuration.

URI: cloud/configurationAccount/{accountId}

HTTP Method: GET

IsConfigurationAccountUsed

Checks whether cloud account is used, for example, scheduled archive.

URI: cloud/isConfigurationAccountUsed/{accountId}

HTTP Method: GET

ListContainers

Gets cloud containers.

URI: cloud/containers?accountId={0}

HTTP Method: GET

ListFiles

Gets cloud files.

URI: cloud/items

HTTP Method: PUT

ListFolders

Gets cloud folders.

URI: cloud/items

HTTP Method: PUT

ListItems

Gets cloud files and folders.

URI: cloud/items

HTTP Method: PUT

SetCloudConfiguration

Sets cloud configuration.

URI: cloud/config

HTTP Method: PUT

StartDownload

Starts a download.

URI: cloud/download

HTTP Method: PUT

StartUpload

Starts an upload.

URI: cloud/upload

HTTP Method: PUT

UpdateConfigurationAccount

Edit existing account of cloud configuration.

URI: cloud/configurationAccount

HTTP Method: PUT

VerifyConnect

Verify cloud availability.

URI: cloud/verifyConnect

HTTP Method: GET

IClustersManagement

This section describes the service operations available for IClustersManagement at clusters/. The IClustersManagement service contract exposes the Cluster management API.

The URI and HTTP method are provided for each service operation.

The service operations include:

- AddClusterAgent
- AddClusterNodeAgent
- ConvertAgentToClusterNode
- ConvertClusterNodeToAgent
- GetClusterNodes
- GetRecommendedClusterRepositoryIds
- GetReplicatedClusterNodes

AddClusterAgent

Add cluster node to existing cluster.

URI: clusters/new

HTTP Method: PUT

AddClusterNodeAgent

Add cluster node to existing cluster.

URI: clusters/{clusterId}/nodes/new

HTTP Method: POST

ConvertAgentToClusterNode

Convert regular agent to cluster node.

URI: clusters/{agentId}/convertToNode

HTTP Method: POST

ConvertClusterNodeToAgent

Convert cluster node to regular agent.

URI: clusters/{clusterNodeId}/convertToAgent

HTTP Method: POST

GetClusterNodes

Gets a list of cluster nodes for specified cluster agent.

URI: clusters/clusterNodes/{clusterAgentId}

HTTP Method: GET

GetRecommendedClusterRepositoryIds

Gets a list of repository ids that are recommended to be used for cluster nodes.

URI: clusters/recommendedRepositories/{clusterAgentId}

HTTP Method: GET

GetReplicatedClusterNodes

Gets a list of replicated cluster nodes for specified cluster agent.

URI: clusters/replicatedClusterNodes/{clusterAgentId}

HTTP Method: GET

ICoreBackupRestoreManagement

This section describes the service operations available for ICoreBackupRestoreManagement at corebackuprestore/.

The ICoreBackupRestoreManagement service contract is the interface implemented by core configuration backup/restore management service used to perform backup and restore of core configuration.

The URI and HTTP method are provided for each service operation.

The service operations include:

- BackupCoreConfiguration
- ReadRepositoriesFromBackupSettings
- RestoreCoreConfiguration
- RestoreCoreConfigurationWithRestart
- ValidateRepositoryDirectory

BackupCoreConfiguration

Performs backup of Core configuration.

URI: corebackuprestore/backup

HTTP Method: POST

ReadRepositoriesFromBackupSettings

Performs validation of repositories paths in backup configuration file. Returns dictionary.

URI: corebackuprestore/validatexml

HTTP Method: POST

RestoreCoreConfiguration

Performs restore of Core configuration.

URI: corebackuprestore/restore

HTTP Method: POST

RestoreCoreConfigurationWithRestart

Performs restore of Core configuration. Core service will be restarted for apply new configurations.

URI: corebackuprestore/restoreWithRestart

HTTP Method: POST

ValidateRepositoryDirectory

Performs validation of repository directory path. Once the path is verified, a value of 'True' is returned.

URI: corebackuprestore/validatedirectory

HTTP Method: POST

ICoreCallbackManagement

This section describes the service operations available for ICoreCallbackManagement at corecallback/.

The ICoreCallbackManagement service contract is the interface implemented by the core callback management service, used to be called by remote agent.

The URI and HTTP method are provided for each service operation.

The service operations include:

- ProcessAgentProtectionRequest
- VerifyConnect

ProcessAgentProtectionRequest

The method is called by failover agent in order to perform remote pairing. This method is for internal usage only.

URI: corecallback/agentprotectionrequest

HTTP Method: POST

VerifyConnect

Called by agent to verify connectivity to this core.

URI: corecallback/connect

HTTP Method: GET

ICoreDiagnosticsManagement

This section describes the service operations available for ICoreDiagnosticsManagement at corediag/.

The URI and HTTP method are provided for each service operation.

The service operations include:

- CollectServerLogs
- GetAggregateStatus
- GetDiskSubsystemStatus
- GetServerLogs
- GetServerLogsInfo
- GetSystemHWStatus

- StartGatheringDiagnosticInfo
- UploadLogSessions

CollectServerLogs

Starts collecting server logs by way of a background job.

URI: corediag/collectLogs

HTTP Method: POST

GetAggregateStatus

Gets aggregate system status from AMC provider.

URI: corediag/systemAggregateStatus

HTTP Method: GET

GetDiskSubsystemStatus

Gets disk subsystem status from AMC provider.

URI: corediag/diskSubsystemStatus

HTTP Method: GET

GetServerLogs

Gets the entire contents of archived server log files.

URI: corediag/serverLogs/

HTTP Method: GET

GetServerLogsInfo

Gets the name and UTC creation timestimp of current archive file with server logs.

URI: corediag/serverLogsArchiveName

HTTP Method: GET

GetSystemHWStatus

Gets system hardware status from AMC provider.

URI: corediag/systemHWStatus

HTTP Method: GET

StartGatheringDiagnosticInfo

Starts gathering diagnostic information by way of a background job and then returns the following response: StartGatheringDiagnosticInformationResponse.

URI: corediag/diagnostic-info/

HTTP Method: POST

UploadLogSessions

Uploads the current logs for the Core and all protected agents that are online to Gibraltar (http://www.gibraltarsoftware.com/) by way of a background job. Nothing is returned.

URI: corediag/logs/

HTTP Method: POST

ICoreMetadataManagement

This section describes the service operations available for ICoreMetadataManagement at metadata/.

The ICoreMetadataManagement service contract is the interface implemented by the core metadata service, which maintains, caches, and returns core metadata.

The URI and HTTP method are provided for each service operation.

The service operations include:

- GetCached
- GetCachedSummary
- GetCurrent
- GetCurrentSummary
- GetStartupGuideFinished
- SetStartupGuideFinished

GetCached

Gets cached Core metadata.

URI: metadata/cached

HTTP Method: GET

GetCachedSummary

Gets cached Core summary metadata.

URI: metadata/cachedsummary

HTTP Method: GET

GetCurrent

Gets the current Core metadata.

URI: metadata/

HTTP Method: GET

GetCurrentSummary

Gets the current Core summary metadata.

URI: metadata/summary

HTTP Method: GET

GetStartupGuideFinished

Gets the value indicating whether to show startup guide.
URI: metadata/getStartupGuideFinished
HTTP Method: GET

SetStartupGuideFinished

Sets the value indicating whether to show setup wizard on startup.
URI: metadata/setStartupGuideFinished/{finished}
HTTP Method: POST

ICoreSettingsManagement

This section describes the service operations available for ICoreSettingsManagement at settings/.

The ICoreSettingsManagement service contract provides a way to query and set assorted core-wide settings.

The URI and HTTP method are provided for each service operation.

The service operations include:

- GetClientTimeout
- GetCoreInstallPath
- GetCoreSettings
- SetClientTimeout
- SetCoreSettings

GetClientTimeout

Gets timeout settings which are used in REST and socket communication.

URI: settings/clientTimeout

HTTP Method: GET

GetCoreInstallPath

Get path to a directory where Agent and Local Mount Utility installers are located.
URI: settings/installersPath
HTTP Method: GET

GetCoreSettings

Gets general global Core settings such as timeout settings and display name.
URI: settings/core
HTTP Method: GET

SetClientTimeout

Applies timeout settings which are used in REST and socket communication.
URI: settings/clientTimeout
HTTP Method: PUT

SetCoreSettings

Applies general global Core settings such as timeout settings and display name.

URI: settings/core

HTTP Method: PUT

IDatabaseStorageManagement

This section describes the service operations available for IDatabaseStorageManagement at databaseStorage/.

The IDatabaseStorageManagement service contract is the interface for communication and configuration of persistant database storage used for events and reports info storing.

The URI and HTTP method are provided for each service operation.

The service operations include:

- GetDatabaseStorageConfiguration
- GetDefaultDatabaseStorageConfiguration
- SetDatabaseStorageConfiguration
- TestConnection

GetDatabaseStorageConfiguration

Gets database storage configuration (like database server connection settings, database data retention period, connection timeout).

URI: databaseStorage/configuration

HTTP Method: GET

GetDefaultDatabaseStorageConfiguration

Gets default database storage configuration.

URI: databaseStorage/defaults

HTTP Method: GET

SetDatabaseStorageConfiguration

Sets database storage configuration (like database server connection settings, database data retention period, connection timeout).

URI: databaseStorage/configuration

HTTP Method: PUT

TestConnection

Tests connection with database storage instance using specified connection settings.

URI: databaseStorage/connection

HTTP Method: POST

IDiagnosticsManagement

This section describes the service operations available for IDiagnosticsManagement at diag/.

The URI and HTTP method are provided for each service operation.

The service operations include:

- ExecuteRemoteCommand
- GetLog
- GetLogSession
- ReadFile
- RestartService
- UploadLogSessions

ExecuteRemoteCommand

Runs an arbitrary remote command.

URI: diag/command/

HTTP Method: POST

GetLog

Gets the entire contents of the replay.log file.

URI: diag/log/

HTTP Method: GET

GetLogSession

Packages the current log session and returns it as a byte stream. The contents of the stream is a Gibraltar (.glp file).

URI: diag/logSession/

HTTP Method: GET

ReadFile

Reads a file from the local file system and streams it back to the client.

URI: diag/files/?q={path}

HTTP Method: GET

RestartService

Stops, forcibly kills (if necessary), and re-starts the service.

URI: diag/service/

HTTP Method: DELETE

UploadLogSessions

Uploads the current log session to the Gibraltar (http://www.gibraltarsoftware.com/) logging framework.

URI: diag/logSession/

HTTP Method: POST

IEmailsManagement

This section describes the service operations available for IEmailsManagement at emailsmgr/.

The IEmailsManagementservice contract is the WCF contract interface for the Emails management.

The URI and HTTP method are provided for each service operation.

The service operations include:

- GetSmtpServerSettings
- IsConfigured
- SendTestEmail
- SetSmtpServerSettings

GetSmtpServerSettings

Returns SMTP server settings from emails service configuration.
URI: emailsmgr/settings
HTTP Method: GET

IsConfigured

Indicates whether SMTP server settings have been set by user.
URI: emailsmgr/settings/check
HTTP Method: GET

SendTestEmail

Generates and sends a test email using the specified email configuration.
URI: emailsmgr/config/test
HTTP Method: PUT

SetSmtpServerSettings

Sets SMTP server settings within emails service configuration.
URI: emailsmgr/settings
HTTP Method: PUT

IEncryptionKeyManagement

This section describes the service operations available for IEncryptionKeyManagement at encryption/.

The IEncryptionKeyManagement service contract is the encryption key management used in conjunction with the repositories and dependent services.

The URI and HTTP method are provided for each service operation.

The service operations include:

- ChangeKeyStatusToReplication
- ChangeKeyStatusToUniversal
- ChangePassphrase
- Create

ChangeKeyStatusToReplication

Changes key status to 'Replication Key'. Returns updated key info.
URI: encryption/changeKeyStatusToReplication/{keyId}
HTTP Method: POST

ChangeKeyStatusToUniversal

Changes key status to 'Universal Key'. Returns updated key info.
URI: encryption/changeKeyStatusToUniversal
HTTP Method: POST

ChangePassphrase

Changes the passphrase for the specified key.
URI: encryption/passphrase
HTTP Method: POST

Create

Creates a new key with the specified name and passphrase.
URI: encryption/
HTTP Method: POST

DeleteKey

Deletes an encryption key or throws exception if selected key is used.
URI: encryption/delete/{keyId}
HTTP Method: DELETE

Export

Exports a key.
URI: encryption/{keyId}
HTTP Method: GET

GetFilteredKeys

Gets the filtered list of encryption keys. Pass the type of the keys as parameter.
URI: encryption/
HTTP Method: GET

GetKeys

Gets the list of encryption keys.
URI: encryption/
HTTP Method: GET

Import

Imports an exported key. Returns imported key info.
URI: encryption/import
HTTP Method: POST

LockKey

Locks the specified key. Returns updated key info.
URI: encryption/lock/{keyId}
HTTP Method: POST

SerializedExport

Returns serialized binary representation of encryption key by given key ID.
URI: encryption/{keyId}/serialized
HTTP Method: GET

SerializedImport

Imports an encryption key by given serialized binary representation. Returns imported key info.
URI: encryption/serimport
HTTP Method: POST

UnlockKey

Unlocks the specified key. Returns updated key info.
URI: encryption/unlock
HTTP Method: POST

UpdateDescription

Updates existing key name and comment. Returns updated key info.
URI: encryption/{keyId}
HTTP Method: PUT

IEventsManagement

This section describes the service operations available for IEventsManagement at events/. The IEventsManagement service contract exposes the events and alerts on the Core.

The URI and HTTP method are provided for each service operation.

The service operations include:

- DismissAllAgentAlerts
- DismissAllCoreAlerts
- DismissEvent
- GetAgentAlertsByPage
- GetAgentAlertsCount
- GetAgentAlertsSettings
- GetAgentEventsByPage
- GetAgentEventsCount
- GetAllAgentAlerts
- GetAllAgentEvents
- GetAllCoreAlerts
- GetAllCoreEvents
- GetCachedEventsByDate
- GetConfiguration
- GetCoreAlertsByPage
- GetCoreAlertsCount
- GetCoreEventsByPage
- GetCoreEventsCount
- GetDetailsForEvent
- GetEventsByDate
- GetEventTypes
- SendTestEmail
- SetAgentAlertsSettings
- SetConfiguration

DismissAllAgentAlerts

Gets the summary information for all alerts associated with the specified agent. Note that this list can potentially be very large.

URI: events/agents/{agentId}/all

HTTP Method: DELETE

DismissAllCoreAlerts

Marks all alerts for the core as read, thereby dismissing them from the list of alerts.

URI: events/core/all

HTTP Method: DELETE

DismissEvent

Marks an event as read, thereby dismissing it from the list of events.

URI: events/event/{eventId}

HTTP Method: DELETE

GetAgentAlertsByPage

Gets summary info for alerts for a given agent ID, such that the results are paged for easy viewing in a paged grid.

URI: events/agents/{agentId}/alerts/paged?max={max}&page={page}

HTTP Method: GET

GetAgentAlertsCount

Gets non-dismissed alerts count for specified agent.

URI: events/agents/{agentId}/alertsCount

HTTP Method: GET

GetAgentAlertsSettings

Returns the alert settings for the specified agent such as email content and notification settings.

URI: events/config/agents/{agentId}

HTTP Method: GET

GetAgentEventsByPage

Gets summary info for events for a given agent ID, such that the results are paged for easy viewing in a paged grid.

URI: events/agents/{agentId}/paged?max={max}&page={page}

HTTP Method: GET

GetAgentEventsCount

Gets events count for specified agent.

URI: events/agents/{agentId}/eventsCount

HTTP Method: GET

GetAllAgentAlerts

Gets the summary information for all alerts associated with the specified agent. Note that this list can potentially be very large.

URI: events/agents/{agentId}/alerts/all

HTTP Method: GET

GetAllAgentEvents

Gets the summary information for all events associated with the specified agent. Note that this list can potentially be very large.

URI: events/agents/{agentId}/all

HTTP Method: GET

GetAllCoreAlerts

Gets the summary information for all alerts associated with the core. Note that this list can potentially be very large.

URI: events/core/alerts/all

HTTP Method: GET

GetAllCoreEvents

Gets the summary information for all events associated with the core. Note that this list can potentially be very large.

URI: events/core/all

HTTP Method: GET

GetCachedEventsByDate

Gets the summary information for cached events associated with the specified core ordering by date.

URI: events/cachedEventsDateParam

HTTP Method: PUT

GetConfiguration

Returns configuration information for events such as email content and notification settings.

URI: events/config

HTTP Method: GET

GetCoreAlertsByPage

Gets summary info for events for the core, such that the results are paged for easy viewing in a paged grid.

URI: events/core/alerts/paged?max={max}&page={page}

HTTP Method: GET

GetCoreAlertsCount

Gets non-dismissed alerts count for core.

URI: events/core/alertsCount

HTTP Method: GET

GetCoreEventsByPage

Gets summary info for events for the core, such that the results are paged for easy viewing in a paged grid.

URI: events/core/paged?max={max}&page={page}

HTTP Method: GET

GetCoreEventsCount

Gets events count for core.

URI: events/core/eventsCount

HTTP Method: GET

GetDetailsForEvent

Gets the details for a single event.
URI: events/event/{eventId}
HTTP Method: GET

GetEventsByDate

Gets the summary information for all events associated with the specified core ordering by date.
URI: events/EventsDateRange
HTTP Method: PUT

GetEventTypes

Gets the list of all possible event types, organized into groups.
URI: events/types
HTTP Method: GET

SendTestEmail

Generates and sends a test email notification based on the specified email configuration.
URI: events/config/email/test
HTTP Method: PUT

SetAgentAlertsSettings

Sets alert settings for the specified agent such as email content and notification settings.
URI: events/config/agents/{agentId}
HTTP Method: PUT

SetConfiguration

Sets configuration information for events such as email content and notification settings.
URI: events/config
HTTP Method: PUT

IExchangeManagement

This section describes the service operations available for IExchangeManagement at exchange/. The IExchangeManagement service contract exposes the MS Exchange management on the Core.

The URI and HTTP method are provided for each service operation.

The service operations include:

- ForceChecksumCheck
- ForceMountabilityCheck
- GetAgentExchangeServerSettings
- GetChecksumCheckQueueConfiguration
- GetChecksumCheckQueueContents
- GetChecksumCheckQueueEntry

- GetMountabilityQueueConfiguration
- GetMountabilityQueueContents
- GetMountabilityQueueEntry
- SetAgentExchangeServerSettings
- SetChecksumCheckConfiguration
- SetMountabilityConfiguration
- VerifyCredentials

ForceChecksumCheck

Forces checksum verification for the specified recovery point.
URI: exchange/checksumcheck/{recoveryPointId}/force
HTTP Method: POST

ForceMountabilityCheck

Forces mountability verification for the specified recovery point.
URI: exchange/mountabilitycheck/{recoveryPointId}/force
HTTP Method: POST

GetAgentExchangeServerSettings

Gets the exchange server settings for the agent.
URI: exchange/agents/{agentId}/exchangeSettings
HTTP Method: GET

GetChecksumCheckQueueConfiguration

Gets the configuration of the checksum check queue.
URI: exchange/checksumcheckconfig
HTTP Method: GET

GetChecksumCheckQueueContents

Gets the contents of the checksum check queue.
URI: exchange/checksumcheckentries
HTTP Method: GET

GetChecksumCheckQueueEntry

Gets the info for a specific checksum check queue entry.
URI: exchange/checksumcheckentries/{entryid}
HTTP Method: GET

GetMountabilityQueueConfiguration

Gets the configuration of the mountability queue.
URI: exchange/mountabilityConfig
HTTP Method: GET

GetMountabilityQueueContents

Gets the contents of the mountability queue.
URI: exchange/mountabilityentries
HTTP Method: GET

GetMountabilityQueueEntry

Gets the info for a specific moutability queue entry.
URI: exchange/mountabilityentries/{entryid}
HTTP Method: GET

SetAgentExchangeServerSettings

Sets the exchange server settings for the agent.
URI: exchange/agents/{agentId}/exchangeSettings
HTTP Method: PUT

SetChecksumCheckConfiguration

Sets the configuration of the checksum check queue.
URI: exchange/checksumcheckconfig
HTTP Method: POST

SetMountabilityConfiguration

Sets the configuration of the mountability queue.
URI: exchange/mountabilityConfig
HTTP Method: POST

VerifyCredentials

Verifies credentials to Exchange instance. Throws exception on validation failure.
URI: exchange/agent/{agentId}/verifyExchangeCredentials
HTTP Method: PUT

IExportQueueManagement

This section describes the service operations available for IExportQueueManagement at export/queue/. The IExportQueueManagement service contract is a RESTful API for the export queue.

The URI and HTTP method are provided for each service operation.

The service operations include:

- CancelAllExports
- CancelExports
- GetConfiguration
- GetEntryInfo
- GetQueueContents
- SetConfiguration

CancelAllExports

Cancels every export in the queue.
URI: export/queue/entries
HTTP Method: DELETE

CancelExports

Cancels the export queue entries identified by the export IDs.
URI: export/queue/entries/cancel
HTTP Method: POST

GetConfiguration

Gets the configuration of the VM export queue.
URI: export/queue/config
HTTP Method: GET

GetEntryInfo

Gets the info for a specific export queue entry.
URI: export/queue/entries/{exportid}
HTTP Method: GET

GetQueueContents

Gets the contents of the export queue.
URI: export/queue/entries
HTTP Method: GET

SetConfiguration

Sets the configuration of the VM export queue.
URI: export/queue/config
HTTP Method: POST

IExportSchedulerManagement

This section describes the service operations available for IExportSchedulerManagement at export/scheduler/. The IExportSchedulerManagement service contract is a RESTful API for the export schduler.

The URI and HTTP method are provided for each service operation.

The service operations include:

- DeleteAgentExportConfiguration
- DeleteEC2Credentials
- ForceExport
- GetAgentExportConfiguration
- GetAllAgentsExportConfiguration
- GetAllAgentsExportStatus

- GetAllAgentsExportStatusWithAdHoc
- GetAllEC2Credentials
- GetEC2Credentials
- GetEc2Environment
- GetFreeSpaceOnNetworkShare
- GetGroupAgentsExportStatus
- GetHyperVCapabilities
- GetVSphereServerInformation
- SetAgentExportConfiguration
- UpdateEC2Credentials
- ValidateEC2Credentials
- ValidateExportLocation
- ValidateHyperVCredentials
- ValidateHyperVDiskExists
- ValidateHyperVRoleIsInstalled
- ValidateHyperVUefiPartitionByAgentId
- ValidateHyperVUefiPartitionByRecoveryPointId
- ValidateRemoteLinuxMachineCredentials
- ValidateStoragePoolSettings
- ValidateSystemVolumeDiskLocation
- ValidateVirtualBoxInstalled
- ValidateVirtualBoxUefiPartition
- ValidateWindowsMachineCredentials

DeleteAgentExportConfiguration

Deletes the export configuration for the agent.

URI: export/schedule/{agentId}/pgs

HTTP Method: DELETE

DeleteEC2Credentials

Deletes EC2 credential.

URI: export/schedule/ec2/credentials/{key}

HTTP Method: DELETE

ForceExport

Immediately starts VM export of a recovery point which corresponds to specified agent. The recovery point and resulting VM type (Workstation, EC2 or ESXi) should be specified in the request instance.

URI: export/schedule/{agentId}/export

HTTP Method: POST

GetAgentExportConfiguration

Gets the export configuration for the agent.

URI: export/schedule/{agentId}/pgs

HTTP Method: GET

GetAllAgentsExportConfiguration

Gets the export configuration for all agents.

URI: export/schedule/agents/allExportConfig

HTTP Method: GET

GetAllAgentsExportStatus

Gets a summary of the export status for every agent on the core for which Virtual Standby is enabled.

URI: export/schedule/agents/all

HTTP Method: GET

GetAllAgentsExportStatusWithAdHoc

Gets a summary of the export status for every agent on the core. Includes AdHoc exports and Virtual Standby exports.

URI: export/schedule/agents/alladhoc

HTTP Method: GET

GetAllEC2Credentials

Gets all EC2 credentials.

URI: export/schedule/ec2/credentials

HTTP Method: GET

GetEC2Credentials

Gets specific EC2 credentials.

URI: export/schedule/ec2/credentials/{key}

HTTP Method: GET

GetEc2Environment

Gets EC2 security information specific to an environment whose method is being called.

URI: export/schedule/ec2/environment/{credentialKey}

HTTP Method: GET

GetFreeSpaceOnNetworkShare

Validates a UNC share path.

URI: export/schedule/uncshare/validate

HTTP Method: POST

GetGroupAgentsExportStatus

Gets a summary of the export status for every agent in the given group for which Virtual Standby is enabled.
URI: export/schedule/groups/{groupId}/all
HTTP Method: GET

GetHyperVCapabilities

Gets Hyper-V capabilities.
URI: export/schedule/hyperv/capabilities
HTTP Method: POST

GetVSphereServerInformation

Retrieves the details about a VSphere server for UI functionality.
URI: export/schedule/vsphere/info
HTTP Method: POST

SetAgentExportConfiguration

Sets the export configuration for the agent.
URI: export/schedule/{agentId}/pgs
HTTP Method: PUT

UpdateEC2Credentials

Creates or updates EC2 credential.
URI: export/schedule/ec2/credentials
HTTP Method: POST

ValidateEC2Credentials

Validates an EC2 credential.
URI: export/schedule/ec2/validate
HTTP Method: POST

ValidateExportLocation

Validates location for virtual machine.
URI: export/schedule/vm/location/validate
HTTP Method: POST

ValidateHyperVCredentials

Validates an Hyper-V credentials.
URI: export/schedule/hyperv/credentials
HTTP Method: POST

ValidateHyperVDiskExists

Validates Hyper-V disk exists.
URI: export/schedule/hyperv/location/{path}
HTTP Method: POST

ValidateHyperVRoleIsInstalled

Validates Hyper-V credentials.
URI: export/schedule/hyperv/role
HTTP Method: POST

ValidateHyperVUefiPartitionByAgentId

Validates if this machine has UEFI partition by agent identifier.
URI: export/schedule/hyperv/validateUefiByAgent/{agentId}
HTTP Method: POST

ValidateHyperVUefiPartitionByRecoveryPointId

Validates if this machine has UEFI partition by recovery point identifier.
URI: export/schedule/hyperv/validateUefiByRecoveryPoints/{recoveryPointId}
HTTP Method: POST

ValidateRemoteLinuxMachineCredentials

Validates VirtualBox credentials.
URI: export/schedule/vbox/credentials
HTTP Method: POST

ValidateStoragePoolSettings

Validates whether Storage Pool feature was enabled on the protected machine.
URI: export/schedule/storagepool/{agentId}
HTTP Method: POST

ValidateSystemVolumeDiskLocation

Validates location for virtual machine.
URI: export/schedule/systemvolume/location/validate
HTTP Method: POST

ValidateVirtualBoxInstalled

Validates VirtualBox is installed on local/remote machine.
URI: export/schedule/vbox/installed?isVirtualMachineUpdating={isVirtualMachineUpdating}
HTTP Method: POST

ValidateVirtualBoxUefiPartition

Validates whether target VM configuration has UEFI support.
URI: export/schedule/vbox/uefi/validate/{agentId}/{recoveryPointId}
HTTP Method: PUT

ValidateWindowsMachineCredentials

Validates VirtualBox Windows credentials.
URI: export/schedule/vbox/wincredentials/{path}/{userName}
HTTP Method: POST

IHyperVAgentManagement

This section describes the service operations available for IHyperVAgentManagement at hypervagent/.

The URI and HTTP method are provided for each service operation.

The service operations include:

- AddDvdDrive
- AddIsoImage
- AddNetworkAdapter
- AttachVirtualDisk
- DeleteVirtualMachine
- DetachVirtualDisk
- EndSession
- GetAvailableVirtualNetworks
- GetMaximumProcessorCount
- GetOrCreateVirtualMachineAndAttach
- GetSnapshotsCount
- GetVirtualDisks
- GetVirtualMachineName
- InsertIntegrationServices
- PingSession
- RenameVirtualMachine
- SetProcessorCount
- SetRamMegabytes
- VerifyConnection

AddDvdDrive

Adds a DVD drive to current virtual machine.
URI: hypervagent/{virtualMachineId}/dvd
PUT

AddIsoImage

Adds an ISO image to a DVD drive. If DVD drive doesn't exist - creates it.
URI: hypervagent/{virtualMachineId}/iso/{isoPath}
HTTP Method: PUT

AddNetworkAdapter

Adds new network adapter to current virtual machine.
URI: hypervagent/{virtualMachineId}/nic/{networkAdapterName}
HTTP Method: PUT

AttachVirtualDisk

Attaches a virtual disk to current virtual machine.
URI: hypervagent/{virtualMachineId}/disks/{diskPath}/{storageController}
HTTP Method: PUT

DeleteVirtualMachine

Deletes current virtual machine and detaches from it.
URI: hypervagent/existent/{virtualMachineId}
HTTP Method: DELETE

DetachVirtualDisk

Detaches a virtual disk from current virtual machine.
hypervagent/{virtualMachineId}/disks/{diskPath}
HTTP Method: DELETE

EndSession

Tells Hyper-V Agent to finish session with the virtual machine.
URI: hypervagent/{virtualMachineId}/endsession
HTTP Method: DELETE

GetAvailableVirtualNetworks

Gets a list of virtual network adapters on Hyper-V server available for a virtual machine.
URI: hypervagent/{virtualMachineId}/availablevirtualnetworks
HTTP Method: GET

GetMaximumProcessorCount

Gets the maximum number of virtual CPUs that could be attached to virtual machine.
URI: hypervagent/cpu/{operatingSystemFamily}
HTTP Method: GET

GetOrCreateVirtualMachineAndAttach

Gets or creates virtual machine specified in request parameter.
URI: hypervagent/getOrCreate
HTTP Method: POST

GetSnapshotsCount

Get the count of snapshots for virtual machine.
URI: hypervagent/{virtualMachineId}/getSnapshotsCount
HTTP Method: GET

GetVirtualDisks

Gets a list of virtual disks currently attached to current virtual machine.
URI: hypervagent/{virtualMachineId}/disks
HTTP Method: GET

GetVirtualMachineName

Gets name of current virtual machine.
URI: hypervagent/{virtualMachineId}/name
HTTP Method: GET

InsertIntegrationServices

Mounts the integration services setup disk.
URI: hypervagent/{virtualMachineId}/integrationservices
HTTP Method: PUT

PingSession

Indicates that session is still being used.
URI: hypervagent/{virtualMachineId}/pingSession
HTTP Method: PUT

RenameVirtualMachine

Renames current virtual machine.
URI: hypervagent/{virtualMachineId}/newname/{newVirtualMachineName}
HTTP Method: PUT

SetProcessorCount

Changes number of virtual CPUs in current virtual machine.
URI: hypervagent/{virtualMachineId}/cpu/{processorCount}
HTTP Method: PUT

SetRamMegabytes

Changes amount of RAM in current virtual machine.

URI: hypervagent/{virtualMachineId}/ram/{ramValue}

HTTP Method: PUT

VerifyConnection

Verifies connection to the running HyperV Agent.

URI: hypervagent/connect

HTTP Method: GET

IIsoDatabaseManagement

This section describes the service operations available for IIsoDatabaseManagement at bootcdbuilder/. The URI and HTTP method are provided for each service operation.

The service operations include:

- DeleteIsoEntry
- GetAllIsoEntries

DeleteIsoEntry

Asks service to delete particular entry.

URI: bootcdbuilder/isos/{entryId}

HTTP Method: DELETE

GetAllIsoEntries

Returns full list of ISO files which have been created previously.

URI: bootcdbuilder/isos

HTTP Method: GET

ILicenseManagement

This section describes the service operations available for ILicenseManagement at license/. The ILicenseManagement service contract is the interface implemented by the core licene management service, which provides license information and functionality.

The URI and HTTP method are provided for each service operation.

The service operations include:

- ChangeGroupKey
- ForcePhoneHome
- GetAgentLicenseInfo
- GetCoreLicenseInfo
- GetLicenseConstraintsInfo
- GetLicenseInfo
- GetLicenseStatesNotifications

- IsKeySpecifiedAndValid
- IsPhoneHomeEnable
- IsPhoneHomeInProgress

ChangeGroupKey

Gets new group key from the UI, validates it, and then returns the validation results.
URI: license/changeGroupKey/{groupKey}
HTTP Method: POST

ForcePhoneHome

Forces connection with License Portal immediately.
URI: license/phoneHome/force
HTTP Method: POST

GetAgentLicenseInfo

Gets licensing information for the given agent.
URI: license/agent/{agentId}
HTTP Method: GET

GetCoreLicenseInfo

Gets core licensing information.
URI: license/core
HTTP Method: GET

GetLicenseConstraintsInfo

Gets description for all enabled constraints.
URI: license/constraintsInfo
HTTP Method: GET

GetLicenseInfo

Gets licensing information for Core and all the agents.
URI: license/licenseInfo
HTTP Method: GET

GetLicenseStatesNotifications

Gets license states notifications for Core and all the agents.
URI: license/licenseStatesNotifications
HTTP Method: GET

IsKeySpecifiedAndValid

Gets state of the key.

URI: license/key

HTTP Method: GET

IsPhoneHomeEnable

Determines if the phone home operation is enabled.

URI: license/phoneHome/isEnable

HTTP Method: GET

IsPhoneHomeInProgress

Determines if the phone home operation is in progress.

URI: license/phoneHome/isInProgress

HTTP Method: GET

ILocalizationManagement

This section describes the service operations available for ILocalizationManagement at localization/.

The ILocalizationManagement service contract is the interface implemented by the common localization service with a persistent configuration, which maintains, set and gets product's current culture.

The URI and HTTP method are provided for each service operation.

The service operations include:

- GetAvailableCultures
- GetCurrentCulture
- SetCurrentCulture

GetAvailableCultures

Gets a list of all available resource cultures.

URI: localization/availableCultures

HTTP Method: GET

GetCurrentCulture

Gets the current resource culture.

URI: localization/currentCulture

HTTP Method: GET

SetCurrentCulture

Sets new culture to all resources.

URI: localization/newCulture/{lcid}

HTTP Method: POST

ILocalMountManagement

This section describes the service operations available for ILocalMountManagement at mounts/.

The ILocalMountManagement service contract is the interface implemented by the local mounts management service, used to mount and unmount volume images on the local core.

The URI and HTTP method are provided for each service operation.

The service operations include:

- Dismount
- DismountAll
- DismountAllAgent
- GetAgentMounts
- GetMountOptions
- GetMounts
- IsVolumeFileSystemCompatible
- StartMount
- VerifyVolumeImagesMountability

Dismount

Dismounts one mounted volume.

URI: mounts/volume/{mountedVolumeName}

HTTP Method: DELETE

DismountAll

Dismounts all mounted volumes.

URI: mounts/allvolumes

HTTP Method: DELETE

DismountAllAgent

Dismounts all mounted volumes which came from a specific agent.

URI: mounts/agents/{agentId}/allvolumes

HTTP Method: DELETE

GetAgentMounts

Gets the list of currently mounted volumes which came from a specific agent.

URI: mounts/agents/{agentId}

HTTP Method: GET

GetMountOptions

Gets options for where to mount recovery points.

URI: mounts/options

HTTP Method: GET

GetMounts

Gets the list of currently mounted volumes.
URI: mounts/
HTTP Method: GET

IsVolumeFileSystemCompatible

Verifies whether volume file system is compatible with current operation system.
URI: mounts/supportos/{volumeImageId}
HTTP Method: GET

StartMount

Starts mounting a specified recovery point.
URI: mounts/
HTTP Method: POST

VerifyVolumeImagesMountability

Validates mountability of volume images.
URI: mounts/verifyVolumeImagesMountability
HTTP Method: POST

ILoggingManagement

This section describes the service operations available for ILoggingManagement at logs/. The ILoggingManagement service contract represents Replay.Core.Contracts.Logging.ILoggingManagement.

The URI and HTTP method are provided for each service operation.

The service operations include:

- AllRegisteredTraceLogs
- EnableTraceLog
- GetEnabledTraceLogs
- SetEnabledTraceLogs
- TraceLogSuggestions
- UnregisterTraceLogs

AllRegisteredTraceLogs

Gets all registered trace logs entries.
URI: logs/allRegisteredTraceLogs
HTTP Method: GET

EnableTraceLog

Enables specified trace log.
URI: logs/enableTraceLog
HTTP Method: PUT

GetEnabledTraceLogs

Gets enabled trace logs entries.

URI: logs/enabledTraceLogs

HTTP Method: GET

SetEnabledTraceLogs

Applies new set of enabled trace logs.

URI: logs/enableTraceLogs

HTTP Method: POST

TraceLogSuggestions

Gets suggestions to enable trace logs.

URI: logs/traceLogSuggestions

HTTP Method: GET

UnregisterTraceLogs

Unregisteres trace log by filter name.

URI: logs/unregisterTraceLogs

HTTP Method: POST

ILogTruncationManagement

This section describes the service operations available for ILogTruncationManagement at logtruncation/.

The ILogTruncationManagement service contract is the interface implemented by core log truncation management service which provides log truncation functionality.

The URI and HTTP method are provided for each service operation.

The service operations include:

- ForceLogTruncation

ForceLogTruncation

Forces a log truncation job for the specified agent.

URI: logtruncation/agents/{agentId}/force

HTTP Method: POST

INightlyJobsManagement

This section describes the service operations available for INightlyJobsManagement at nightlyJobs/. The INightlyJobsManagement service contract is the RESTful API for nightly jobs.

The URI and HTTP method are provided for each service operation.

The service operations include:

- CancelNightlyJobs
- GetAgentJobConfiguration

- GetAgentNightlyJobs
- GetJobConfiguration
- GetNightlyJobsSettings
- GetNightlyJobsStatus
- SetAgentJobConfiguration
- SetAgentNightlyJobs
- SetJobConfiguration
- SetNightlyJobsSettings

CancelNightlyJobs

Cancels all nightly jobs execution.
URI: nightlyJobs/
HTTP Method: DELETE

GetAgentJobConfiguration

Gets job configuration for the specified job of the agent.
URI: nightlyJobs/jobConfiguration/{jobId}/{agentId}
HTTP Method: GET

GetAgentNightlyJobs

Gets jobs for the specified agent.
URI: nightlyJobs/config/{agentId}
HTTP Method: GET

GetJobConfiguration

Gets job configuration for the specified job.
URI: nightlyJobs/jobConfiguration/{jobId}
HTTP Method: GET

GetNightlyJobsSettings

Gets nightly jobs settings for the Core.
URI: nightlyJobs/config
HTTP Method: GET

GetNightlyJobsStatus

Determines whether nightly jobs are in progress and then gets the transaction id of currently running jobs.
URI: nightlyJobs/status
HTTP Method: GET

SetAgentJobConfiguration

Sets job configuration for specified agent.

URI: nightlyJobs/jobConfiguration/{jobId}/{agentId}

HTTP Method: PUT

SetAgentNightlyJobs

Sets enabled nightly jobs for the agent.

URI: nightlyJobs/config/{agentId}

HTTP Method: PUT

SetJobConfiguration

Sets job configuration.

URI: nightlyJobs/jobConfiguration/{jobId}

HTTP Method: PUT

SetNightlyJobsSettings

Sets nightly jobs settings for the Core.

URI: nightlyJobs/config

HTTP Method: PUT

IProtectedItemsManagement

This section describes the service operations available for IProtectedItemsManagement at protectedItems/.

The IProtectedItemsManagement service contract exposes protected items such as agents, clusters, cluster nodes and groups as tree.

The URI and HTTP method are provided for each service operation.

The service operations include:

- GetProtectedItems

GetProtectedItems

Gets the protected items tree.

URI: protectedItems/allItems

HTTP Method: GET

IPushInstallCommunication

This section describes the service operations available for IPushInstallCommunication at pushinstallcomm/.

The IPushInstallCommunication service contract is used by Replay internally to communicate between core and PushInstall agent. Do not use this API directly.

The URI and HTTP method are provided for each service operation.

The service operations include:

- GetInstallationStatus
- SetInstallationProgress

GetInstallationStatus

Gets installation progress of child push install job.

URI: pushinstallcomm/status/jobs/{jobId}

HTTP Method: GET

SetInstallationProgress

This method is called by push install functionality which is used to track the progress of remote agent installation.

> **CAUTION: This method is for internal usage only.**

URI: pushinstallcomm/setProgress/jobs/{jobId}/{messageType}/{progress}/{message}

HTTP Method: POST

IPushInstallManagement

This section describes the service operations available for IPushInstallManagement at pushinstall/.

The IPushInstallManagement service contract is the interface implement ability to install agent on remote machine. Push install is a functionality that allows user to initiate and control agent installation on remote environments in a network.

The URI and HTTP method are provided for each service operation.

The service operations include:

- DeployAgents
- GetAgentServiceState
- GetConfiguration
- GetDeployAgentVersion
- GetInstalledProducts
- IsRemoveAgentProductRequired
- SetConfiguration
- ValidateMachine

DeployAgents

Starts deploying agent(s) to remote machines.

URI: pushinstall/deployAgents

HTTP Method: PUT

GetAgentServiceState

Checks if agent is already installed on remote machine.

URI: pushinstall/isPushInstallNeed

HTTP Method: POST

GetConfiguration

Gets PushInstall configuration.

URI: pushinstall/config

HTTP Method: GET

GetDeployAgentVersion

Get the version of the Agent to be deployed.

URI: pushinstall/deployAgentVersion

HTTP Method: GET

GetInstalledProducts

Checks for already installed products.

URI: pushinstall/getInstalledProducts

HTTP Method: POST

IsRemoveAgentProductRequired

Checks whether removing of the Agent product is required.

URI: pushinstall/isRemoveAgentProductRequired/{removeProductId}/{installedProductVersion}

HTTP Method: GET

SetConfiguration

Sets PushInstall configuration.

URI: pushinstall/config

HTTP Method: PUT

ValidateMachine

Checks machine availability and permissions for install agent.

URI: pushinstall/validateMachine

HTTP Method: PUT

IRecoveryPointsManagement

This section describes the service operations available for IRecoveryPointsManagement at recoveryPoints/.

The IRecoveryPointsManagement service contract is the interface implemented by the recovery points management service, which exposes information about recovery points to REST clients.

The URI and HTTP method are provided for each service operation.

The service operations include:

- CheckAgentVolumeImages
- CheckAgentVolumeImagesSupport
- DeleteOrphanVolumeImagesChains
- DeleteRecoveryPointsChain
- DeleteRecoveryPointsForAgent
- DeleteRecoveryPointsRange
- DeleteVolumeImages
- DeleteVolumeImagesSupport
- ExportVolumeImagesTree
- GetAgentRecoveryPointCheckSettings
- GetAgentsRecoveryPointsInfo
- GetAllRecoveryPoints
- GetFilteredRecoveryPointsByPage
- GetImageRawData
- GetImageRawKeys
- GetImageRawKeysText
- GetMostRecentRecoveryPoints
- GetRecoveryPointDetails
- GetRecoveryPointLockedKeys
- GetRecoveryPoints
- GetRecoveryPointsByDateRange
- GetRecoveryPointsByPage
- GetRecoveryPointSummary
- GetVolumeImageDetails
- IsOrphanRecoveryPoint
- SetAgentRecoveryPointCheckSettings

CheckAgentVolumeImages

Starts a task that performs integrity check of all agent's recovery points and tries to fix errors being found.

URI: recoveryPoints/agents/checkVolumes

HTTP Method: POST

CheckAgentVolumeImagesSupport

Starts a task that performs integrity check of all agent's recovery points and tries to fix errors being found. For support purpose. Use '*' symbol as mask for RP and volumes.

URI:
recoveryPoints/agents/{agentId}/rps/{recoveryPointId}/volumes/{volumeImageId}/integritycheck/target={target}

HTTP Method: GET

DeleteOrphanVolumeImagesChains

Deletes orphan volume images chains for a specified recovery point.
HTTP Method: URI: recoveryPoints/rps/deleteOrphanVolumeImagesChains/{recoveryPointId}
POST

DeleteRecoveryPointsChain

Deletes all volume image chains that contain volume images for a given recovery point.
URI: recoveryPoints/rps/{recoveryPointId}
HTTP Method: DELETE

DeleteRecoveryPointsForAgent

Deletes all recovery points for specified agent.
URI: recoveryPoints/agents/{agentId}
HTTP Method: DELETE

DeleteRecoveryPointsRange

Deletes all recovery points in a specified time period for the specified agent.
URI: recoveryPoints/rps/deleteRecoveryPointsRange
HTTP Method: POST

DeleteVolumeImages

Deletes volume images without rollup for the specified agent. For debug purpose.
URI: recoveryPoints/agents/deleteVolumes
HTTP Method: DELETE

DeleteVolumeImagesSupport

Deletes volume images without rollup for the specified agent.
URI: recoveryPoints/agents/{agentId}/rps/{recoveryPointId}/volumes/{volumeImageId}
HTTP Method: GET

ExportVolumeImagesTree

Exports agent volume images tree to text file.
URI: recoveryPoints/agents/exportVolumeImagesTree
HTTP Method: PUT

GetAgentRecoveryPointCheckSettings

Gets agent recovery point check settings included nightly job and manually started checks.
URI: recoveryPoints/agents/{agentId}/integritycheck/settings
HTTP Method: GET

GetAgentsRecoveryPointsInfo

Gets a list of all agents with recovery points.

URI: recoveryPoints/agents/all

HTTP Method: GET

GetAllRecoveryPoints

Gets the summary information for all recovery points associated with the specified agent. Note that this list can potentially be very large.

URI: recoveryPoints/agents/{agentId}/all

HTTP Method: GET

GetFilteredRecoveryPointsByPage

Gets summary info for recovery points for a given agent ID and filter, such that the results are paged for easy viewing in a paged grid.

URI: recoveryPoints/agents/{agentId}/paged?max={max}&page={page}

HTTP Method: PUT

GetImageRawData

Gets a stream of data consisting of the data in the image at the specified offset and length. Useful only for diagnostic purposes.

URI: recoveryPoints/images/rawdata/{imageId}.rawdata?blockOffset={blockOffset}&blockLength={blockLength}

HTTP Method: GET

GetImageRawKeys

Gets a stream of offset/key pairs, containing the block offsets in the image and the DVM keys of the record at each block offset. Useful only for diagnostic purposes.

URI: recoveryPoints/images/rawkeys/{imageId}.rawkeys

HTTP Method: GET

GetImageRawKeysText

Gets a stream of offset/key pairs, containing the block offsets in the image and the DVM keys of the record at each block offset. Useful only for diagnostic purposes.

URI: recoveryPoints/images/rawkeys/{imageId}.textkeys

HTTP Method: GET

GetMostRecentRecoveryPoints

Gets summary info for the most recent recovery point of every agent specified in the request.

URI: recoveryPoints/recent

HTTP Method: PUT

GetRecoveryPointDetails

Gets detailed info for a single recovery point.

URI: recoveryPoints/rps/{recoveryPointId}/details

HTTP Method: GET

GetRecoveryPointLockedKeys

Gets a list of encryption keys used by a recovery point that are locked and require a passphrase in order to mount.

URI: recoveryPoints/rps/{recoveryPointId}/lockedkeys

HTTP Method: GET

GetRecoveryPoints

Gets the summary information for the recovery points associated with a specified agent that fall outside of a last modified date/time range. You also specify a maximum number of recovery points to return with the specified range.

URI: recoveryPoints/agents/{agentId}/where?max={max}&olderThan={olderThan}&newerThan={newerThan}

HTTP Method: GET

GetRecoveryPointsByDateRange

Gets the summary information for the recovery points associated with a specified agent that fall inside of a last modified date/time range. You also specify a maximum number of recovery points to return with the specified range.

URI: recoveryPoints/agents/{agentId}/daterange?max={max}&startDate={startDate}&endDate={endDate}

HTTP Method: GET

GetRecoveryPointsByPage

Gets summary info for recovery points for a given agent ID, such that the results are paged for easy viewing in a paged grid.

URI: recoveryPoints/agents/{agentId}/paged?max={max}&page={page}

HTTP Method: GET

GetRecoveryPointSummary

Gets detailed info for a single recovery point.

URI: recoveryPoints/rps/{recoveryPointId}/summary

HTTP Method: GET

GetVolumeImageDetails

Gets information for a single volume image specified by unique identifier.

URI: recoveryPoints/images/{imageId}/'

HTTP Method: GET

IsOrphanRecoveryPoint

Gets whether recovery point is orphan or not.

URI: recoveryPoints/rps/{recoveryPointId}/isOrphan

HTTP Method: GET

SetAgentRecoveryPointCheckSettings

Sets agent recovery point check settings included nightly job and manually started checks.

URI: recoveryPoints/agents/{agentId}/integritycheck/settings

HTTP Method: PUT

IRemoteMountManagement

This section describes the service operations available for IRemoteMountManagement at remoteMounts/.

The IRemoteMountManagement service contract is the Interface implemented by the remote mounts management service, used to diconnect mounted volume images on the remote machines.

The URI and HTTP method are provided for each service operation.

The service operations include:

- AddRemoteMount
- Disconnect
- DisconnectAll
- DisconnectAllForAgent
- GetAgentRemoteMounts
- GetRemoteMounts

AddRemoteMount

Add information about remote mount point.

URI: remoteMounts/addRemoteMount

HTTP Method: POST

Disconnect

Disconnect remote mount point.

URI: remoteMounts/disconnect/{remoteMountItemId}/mountType/{mountType}

HTTP Method: POST

DisconnectAll

Disconnect all remote mount points.

URI: remoteMounts/disconnectAll

HTTP Method: POST

DisconnectAllForAgent

Disconnect all remote mount points for particular agent.

URI: remoteMounts/disconnectAllForAgent/{agentId}

HTTP Method: POST

GetAgentRemoteMounts

Gets information about remote mount points for particular agent.

URI: remoteMounts/getAgentRemoteMoutns/{agentId}

HTTP Method: POST

GetRemoteMounts

Gets information about remote mount points.

URI: remoteMounts/remoteMountInfos

HTTP Method: POST

IReplayEngineManagement

This section describes the service operations available for IReplayEngineManagement at replayEngine/.

The IReplayEngineManagement service contract is the interface implemented by the Replay Engine management service, used to configure and monitor the Replay Engine.

The URI and HTTP method are provided for each service operation.

The service operations include:

- CloseConnection
- GetConfiguration
- GetConnections
- SetConfiguration

CloseConnection

Forcibly closes an existing connection to the AppAssure Engine.

URI: replayEngine/connections/{id}

HTTP Method: DELETE

GetConfiguration

Retrieves the current configuration of the Replay Engine service.

URI: replayEngine/config

HTTP Method: GET

GetConnections

Gets a list of the active connections to the Replay Engine service.

URI: replayEngine/connections

HTTP Method: GET

SetConfiguration

Sets the configuration of the service.

URI: replayEngine/config

HTTP Method: POST

IReplicationCommunication

This section describes the service operations available for IReplicationCommunication at replicationcomms/.

The IReplicationCommunication service contract is the used by Replay internally to communicate between cores for replication. Do not use this API directly.

The URI and HTTP method are provided for each service operation.

The service operations include:

- AddAgentsByDemand
- AddAgentsByRequest
- CancelRemoteMetadataUpdate
- CancelRemoteRollup
- CancelRemoteVolumeImagesDeletion
- CancelUploadFile
- ContinueUploadFile
- DeleteAgent
- DeletePairing
- DemandPairing
- EndCheck
- EndUploadFile
- EndVolumeImageReplicationSession
- EndVolumeImageReplicationSessionOld
- EndVolumeImageVerificationSession
- GetAgentRecoveryPointDetails
- GetAgentRecoveryPoints
- GetAgentRecoveryPointsCount
- GetAgentRepositoryRelationships
- GetAgents
- GetBasicReplicatedVolumeImagesInfo
- GetBasicReplicatedVolumeImagesInfoOld
- GetConsumedSeedDrives
- GetCoreId
- GetExchangeVersions
- GetFileInfoForExchangeDll
- GetMetadataUpdateProgress
- GetNextBlockMD5Digest
- GetNextBlockMD5Digest
- GetPairingStatus
- GetRemoteMasterCoresForDemand
- GetReplicatedAgents
- GetReplicatedAgentsRecoveryPointsInfo

- GetReplicatedAgentsStorageUsage
- GetRepositories
- GetRepositoryFreeSpaceForAgent
- GetRollupProgress
- GetVolumeImageBlockDigest
- GetVolumeImageBlockIndex
- GetVolumeImageDigest
- GetVolumeImageMD5CheckResult
- GetVolumeImagesDeletionProgress
- NegotiateMissingRecords
- NegotiateMissingRecordsOld
- NewVolumeImageMD5BlockCheck
- RequestPairing
- SelectRangeAgentRecoveryPoints
- StartHashVerificationJob
- StartMetadataUpdate
- StartMetadataUpdate
- StartMetadataUpdateJob
- StartNewUploadSession
- StartNewVolumeImageReplicationSession
- StartRemoteReplicationJob
- StartRollup
- StartRollupJob
- StartTransferJob
- StartVolumeImagesDeletion
- StartVolumeImagesDeletionJob
- StartVolumeImagesDeletionJobOld
- StartVolumeImagesDeletionOld
- StartVolumeImageVerificationSession
- SyncPairingStatus
- SyncRemoteReplicationJob
- TransferMissingRecords
- TransferMissingRecordsOld
- TransferUndiscoveredRecords
- TransferVolumeBlock
- UpdateIndex
- UpdateMasterStorageUsage
- UpdateReplicationStatus
- VerifyAddAgentsByDemand
- VerifyAddAgentsByRequest

- VerifyReplicationAbility
- VerifyReplicationCorePairingAbility

AddAgentsByDemand

Add agents by demand to a remote slave core.
URI: replicationcomms/slave/agents/demand
HTTP Method: POST

AddAgentsByRequest

Add agents by request to a remote slave.
URI: replicationcomms/slave/agents/request
HTTP Method: POST

CancelRemoteMetadataUpdate

Cancels metadata update phase of replication job for replicated agent.
URI: replicationcomms/slave/agents/{agentId}/metadataUpdate
HTTP Method: DELETE

CancelRemoteRollup

Cancels rollup phase of replication job for replicated agent.
URI: replicationcomms/slave/agents/{agentId}/rollup
HTTP Method: DELETE

CancelRemoteVolumeImagesDeletion

Cancels volume images deletion phase of replication job for replicated agent.
URI: replicationcomms/slave/agents/{agentId}/volumeImagesDeletion
HTTP Method: DELETE

CancelUploadFile

Cancels current upload session.
URI: replicationcomms/slave/sessions/{uploadSessionId}/
HTTP Method: DELETE

ContinueUploadFile

Reads data from slave core in current upload session.
URI: replicationcomms/slave/sessions/{uploadSessionId}/data/{dataSize}
HTTP Method: POST

DeleteAgent

Deletes a replicated agent from the slave core, including all of its recovery points.
URI: replicationcomms/slave/agents/{agentId}
HTTP Method: DELETE

DeletePairing

Removes replication relationship with Master Core on Slave's Core side. Actual replicated and protected agent on Master and Slave Cores stay available.

URI: replicationcomms/slave/pairing?deleteRecoveryPoints={deleteRecoveryPoints}

HTTP Method: DELETE

DemandPairing

Demands the establishment of a pairing relationship with a remote core. Demands are only accepted if the caller performs NTLM authentication as a member of the administrators group. This method will reset connection for establish new secured connection.

URI: replicationcomms/slave/pairing/demand

HTTP Method: POST

EndCheck

Cancels current MD5 check context.

URI: replicationcomms/slave/contexts/{contextId}/

HTTP Method: POST

EndUploadFile

Ends current upload session and cheks MD5 hash of received file.

URI: replicationcomms/slave/sessions/{uploadSessionId}/

HTTP Method: POST

EndVolumeImageReplicationSession

Ends the volume image replication session, optionally committing the transferred volume image.

URI: replicationcomms/slave/{agentId}/{jobId}/{sessionId}?commit={commit}

HTTP Method: DELETE

EndVolumeImageReplicationSessionOld

Ends the volume image replication session, optionally committing the transferred volume image.

URI: replicationcomms/slave/sessions/{sessionId}?commit={commit}

HTTP Method: DELETE

EndVolumeImageVerificationSession

Ends integrity check session with a slave core.

URI: replicationcomms/slave/check/session/{sessionId}/end?commit={commit}

HTTP Method: DELETE

GetAgentRecoveryPointDetails

Gets the details for a single replicated recovery point.

URI: replicationcomms/slave/agents/{agentId}/rps/{recoveryPointId}

HTTP Method: GET

GetAgentRecoveryPoints

Gets the recovery points replicated for the given agent.

URI: replicationcomms/slave/agents/{agentId}/rps

HTTP Method: GET

GetAgentRecoveryPointsCount

Gets count of the recovery points replicated for the given agent.

URI: replicationcomms/slave/agents/{agentId}/rpsCount

HTTP Method: GET

GetAgentRepositoryRelationships

Gets the repositories for replicated agents.

URI: replicationcomms/slave/cores/agentRepositoryRelationships

HTTP Method: GET

GetAgents

Gets the list of all agents. A pairing must be in place, and this request must be authenticated by the master core's client certificate.

URI: replicationcomms/slave/agents

HTTP Method: GET

GetBasicReplicatedVolumeImagesInfo

Gets the details for all recovery points replicated for the given agent.

URI: replicationcomms/slave/agents/{agentId}/replicatedVolumeImagesNew

HTTP Method: GET

GetBasicReplicatedVolumeImagesInfoOld

Gets the details for all recovery points replicated for the given agent.

URI: replicationcomms/slave/agents/{agentId}/replicatedVolumeImages

HTTP Method: GET

GetConsumedSeedDrives

Gets identifiers of seed drives consumed on the Core for specified agent.

URI: replicationcomms/consumedSeedDrives/{agentId}

HTTP Method: GET

GetCoreId

Tests connection to remote core and returns core ID. If useCredentials in true then NTLM authentication used, otherwise Anonymous authentication.

URI: replicationcomms/slave/validate/?useCredentials={useCredentials}

HTTP Method: GET

GetExchangeVersions

Gets versions of Exchange dlls, with present on remote slave core.

URI: replicationcomms/slave/exchange

HTTP Method: GET

GetFileInfoForExchangeDll

Gets information for given Exchange DLL file.

URI: replicationcomms/slave/exchange/dllinfo/{fileName}

HTTP Method: POST

GetMetadataUpdateProgress

Gets status of the metadata update job initiated from master core.

URI: replicationcomms/slave/agents/{agentId}/metadataUpdate/status

HTTP Method: GET

GetNextBlockMD5Digest

Get MD5 Digest for specified block.

URI: replicationcomms/slave/getBlockMD5/contextId/{contextId}/count/{blockCount}/md5/{blockMD5}

HTTP Method: POST

GetNextBlockMD5Digest

Gets Volume Image MD5 and size. It does not invoke a new check.

URI: replicationcomms/slave/getVolumeImageMD5/?volumeId={volumeId}

HTTP Method: GET

GetPairingStatus

Gets the status of the pairing between the calling core and the remote slave core. The caller is identified by its SSL client certificate. This method is available to a remote core regardless of whether it was paired by way of a request or the initiation of the pairing itself.

URI: replicationcomms/slave/pairing/status

HTTP Method: GET

GetRemoteMasterCoresForDemand

Getting remote masers cores info for current slave core. Using NTLM authentication.

URI: replicationcomms/slave/cores/masters

HTTP Method: GET

GetReplicatedAgents

Gets the list of agents the caller is replicating to this slave core. A pairing must be in place, and this request must be authenticated by the master core's client certificate.

URI: replicationcomms/slave/agents

HTTP Method: GET

GetReplicatedAgentsRecoveryPointsInfo

Gets the list of agents which have recovery points on a remote slave core.
URI: replicationcomms/slave/agents/rpsinfo
HTTP Method: GET

GetReplicatedAgentsStorageUsage

Gets a summary of storage usage of the replicated agents.
replicationcomms/slave/agents
HTTP Method: GET

GetRepositories

Gets all repositories. With certificate authentication for already paired cores.
URI: replicationcomms/slave/repositories
HTTP Method: GET

GetRepositoryFreeSpaceForAgent

Get free space for agent's remote repository.
URI: replicationcomms/slave/agents/{agentId}
HTTP Method: GET

GetRollupProgress

Gets status of the rollup job initiated from master core.
replicationcomms/slave/agents/{agentId}/rollup/progress
HTTP Method: GET

GetVolumeImageBlockDigest

Get volume image block digest for specified block.
URI: replicationcomms/slave/check/session/{sessionId}/blockDigest
HTTP Method: POST

GetVolumeImageBlockIndex

Get volume image index values for specified block.
URI: replicationcomms/slave/check/session/{sessionId}/index?block={blockNumber}
HTTP Method: GET

GetVolumeImageDigest

Gets volume image digest for a specified integrity check session.
URI: replicationcomms/slave/check/session/{sessionId}/digest
HTTP Method: GET

GetVolumeImageMD5CheckResult

Gets current volume MD5 check result for the context.

URI: replicationcomms/slave/checkResult/{contextId}/

HTTP Method: POST

GetVolumeImagesDeletionProgress

Gets status of the deletion job initiated from master core

URI: replicationcomms/slave/agents/{agentId}/volumeImagesDeletion/progress

HTTP Method: GET

NegotiateMissingRecords

Sends a stream of record metadata for the image being replicated, and receives back a stream of records which are missing from the remote core.

URI: replicationcomms/slave/{agentId}/{jobId}/{sessionId}/records/keys

HTTP Method: POST

NegotiateMissingRecordsOld

Sends a stream of record metadata for the image being replicated, and receives back a stream of records which are missing from the remote core.

URI: replicationcomms/slave/sessions/{sessionId}/records/keys

HTTP Method: POST

NewVolumeImageMD5BlockCheck

Starts new MD5 check and master/slave volume record index compare session. Returns contextId.

URI: replicationcomms/slave/volumeImageMD5BlockCheck/{volumeId}

HTTP Method: POST

RequestPairing

Sends a request to a remote slave for authorization to replicate one or more agents. The request is adjudicated by a human operator and will be approved or denied at a later date. This method will reset connection for establish new secured connection.

URI: replicationcomms/slave/pairing/request

HTTP Method: POST

SelectRangeAgentRecoveryPoints

Select range of the recovery points replicated for the given agent.

URI: replicationcomms/slave/agents/{agentId}/skipCount/{skipCount}/maxCount/{maxCount}/rps

HTTP Method: GET

StartHashVerificationJob

Starts integrity check job for specified agent.

URI: replicationcomms/slave/agents/{agentId}/verification/{jobId}/start

HTTP Method: POST

StartMetadataUpdate

Starts metadata update for specified agent.
URI: replicationcomms/slave/agents/{agentId}/metadataUpdate
HTTP Method: POST

StartMetadataUpdateJob

Starts metadata update job for specified agent.
URI: replicationcomms/slave/agents/{agentId}/metadataUpdateJob
HTTP Method: POST

StartNewUploadSession

Starts new file upload session.
URI: replicationcomms/slave/newsession/
HTTP Method: POST

StartNewVolumeImageReplicationSession

Starts a replication session with a slave core.
URI: replicationcomms/slave/sessions/new
HTTP Method: POST

StartRemoteReplicationJob

Starts remote mirrored replication job on slave core.
URI: replicationcomms/slave/agents/replicationJob/start
HTTP Method: POST

StartRollup

Starts rollup for specified slave agent for specified granularity cells (time intervals).
URI: replicationcomms/slave/agents/{agentId}/rollup
HTTP Method: POST

StartRollupJob

Starts rollup job for specified slave agent for specified granularity cells (time intervals).
URI: replicationcomms/slave/agents/{agentId}/rollupJob
HTTP Method: POST

StartTransferJob

Starts remote mirrored transfer job on slave core.
URI: replicationcomms/slave/agents/{agentId}/transferJob/{jobId}
HTTP Method: POST

StartVolumeImagesDeletion

Starts deletion of volume images with specified identifiers for specified agent.

URI: replicationcomms/slave/agents/{agentId}/volumeImagesDeletionNew

HTTP Method: POST

StartVolumeImagesDeletionJob

Starts deletion of volume images job with specified identifiers for specified agent.

URI: replicationcomms/slave/agents/{agentId}/volumeImagesDeletionJobNew

HTTP Method: POST

StartVolumeImagesDeletionJobOld

Starts deletion of volume images job with specified identifiers for specified agent.

URI: replicationcomms/slave/agents/{agentId}/volumeImagesDeletionJob

HTTP Method: POST

StartVolumeImagesDeletionOld

Starts deletion of volume images with specified identifiers for specified agent.

URI: replicationcomms/slave/agents/{agentId}/volumeImagesDeletion

HTTP Method: POST

StartVolumeImageVerificationSession

Starts a integrity check session with a slave core.

URI: replicationcomms/slave/check/{agentId}/{jobId}/session/start

HTTP Method: POST

SyncPairingStatus

Gets the status of the pairing between the calling core and the remote slave core. The caller is identified by its SSL client certificate. This method is available to a remote core regardless of whether it was paired by way of a request or initiation of the pairing itself.

URI: replicationcomms/slave/pairing/sync

HTTP Method: POST

SyncRemoteReplicationJob

Sync with remote mirrored replication job on slave core.

URI: replicationcomms/slave/agents/replicationJob/sync

HTTP Method: POST

TransferMissingRecords

Sends a stream of raw records to the slave core, the list of which is determined by NegotiateMissingRecords.

URI: replicationcomms/slave/{agentId}/{jobId}/{sessionId}/records/rawdata

HTTP Method: POST

TransferMissingRecordsOld

Sends a stream of raw records to the slave core, the list of which is determined by NegotiateMissingRecords.

URI: replicationcomms/slave/sessions/{sessionId}/records/rawdata

HTTP Method: POST

TransferUndiscoveredRecords

Sends a stream of record data for the image being checked, and receives back a stream of record metadata which are detected as missing from the remote core.

URI: replicationcomms/slave/check/session/{sessionId}/transferRecords

HTTP Method: POST

TransferVolumeBlock

Reads data from agent in current transmit session.

URI:
replicationcomms/slave/blockTransfer/contextId/{contextId}/blockCount/{blockCount}/offset/{offset}/size/{size}

HTTP Method: POST

UpdateIndex

Updates volume image index on remote core.

URI: replicationcomms/slave/check/session/{sessionId}/updateIndex

HTTP Method: POST

UpdateMasterStorageUsage

Reports a summary of storage usage on the master core to the slave core. This primarily exists to support MSP billing needs.

URI: replicationcomms/slave/storage

HTTP Method: PUT

UpdateReplicationStatus

Set replication status on the slave core.

URI: replicationcomms/slave/agents/{agentId}/status

HTTP Method: PUT

VerifyAddAgentsByDemand

Verifies whether agents can be safely replicated by demand

URI: replicationcomms/slave/demand/agents/verify

HTTP Method: POST

VerifyAddAgentsByRequest

Verifies whether agents can be safely replicated by request

URI: replicationcomms/slave/request/agents/verify

HTTP Method: POST

VerifyReplicationAbility

Verifies replication ability.

URI: replicationcomms/slave/agents/{agentId}/replication/verifyStart

HTTP Method: GET

VerifyReplicationCorePairingAbility

Verifies pairing ability.

URI: replicationcomms/slave/replication/verifyStart/?useCredentials={useCredentials}

HTTP Method: POST

IReplicationManagement

This section describes the service operations available for IReplicationManagement at replication/.

The IReplicationManagement service contract is used the for the configuration, management, and monitoring of replication.

The URI and HTTP method are provided for each service operation.

The service operations include:

- AddAgentsByDemand
- AddAgentsByRequest
- DeleteAgentFromMaster
- DeleteAgentFromSlave
- DeleteMasterCore
- DeletePairing
- DeleteSlaveCore
- DemandPairing
- ForceReplication
- GetAgentReplicationSettings
- GetAgentRepositoryRelationships
- GetCoreIdByDescriptor
- GetCoreIdByUrl
- GetCountRemoteAgentsRecoveryPoints
- GetPendingPairingRequest
- GetPendingPairingRequests
- GetCountRemoteAgentsRecoveryPoints
- GetPendingPairingRequest
- GetPendingPairingRequests
- GetRemoteAgentsRecoveryPoints
- GetRemoteCoreRepositories
- GetRemoteCoreRepositoriesForDemand
- GetRemoteCores
- GetRemoteMasterCoresForDemand

- GetRemoteSlaveCoreReplicationPolicy
- GetRemoteSlaveRecoveryPoint
- GetReplicatedAgentsRecoveryPointsInfo
- GetReplicationConfiguration
- IgnorePairingRequest
- RequestForceReplication
- RequestPairing
- RespondToAddAgentsByRequest
- RespondToPairingRequest
- SelectRangeRemoteAgentsRecoveryPoints
- SetAgentReplicationPauseConfiguration
- SetAgentReplicationPauseConfigurationForMasterCores
- SetAgentReplicationPauseConfigurationForSlaveCores
- SetAgentReplicationSettings
- SetRemoteSlaveCoreReplicationPolicy
- SetReplicationConfiguration
- SwitchFailoverAgentToReplicated
- SwitchReplicatedAgentToFailover
- UpdateSlaveCoreSettings
- VerifyAddAgentsByDemand
- VerifyAddAgentsByDemandForExistingCore
- VerifyAddAgentsByRequest
- VerifyAddAgentsByRequestForExistingCore
- VerifyCorePairingAbilityByDemand
- VerifyCorePairingAbilityByRequest

AddAgentsByDemand

Add agents to existing pairing by demand.

URI: replication/cores/{coreId}/agents/demand

HTTP Method: POST

AddAgentsByRequest

Add agents to existing pairing by request.

URI: replication/cores/{coreId}/agents/request

HTTP Method: POST

DeleteAgentFromMaster

Delete agent from replication relationship from slave's side only. Actual replicated and protected agent on master and slave cores stay available.

URI: replication/masters/{coreId}/replicatedagents/{agentId}/?deleteRecoveryPoints={deleteRecoveryPoints}

HTTP Method: DELETE

DeleteAgentFromSlave

Delete agent from replication relationship from master's side only. Actual replicated and protected agent on master and slave cores stay available.

URI: replication/slaves/{coreId}/replicatedagents/{agentId}

HTTP Method: DELETE

DeleteMasterCore

Delete remote master core from replication.

URI: replication/masters/{coreId}?deleteRecoveryPoints={deleteRecoveryPoints}

HTTP Method: DELETE

DeletePairing

Delete pairing between master and slave cores.

URI: replication/cores/{coreId}/pairing?deleteRecoveryPoints={deleteRecoveryPoints}

HTTP Method: DELETE

DeleteSlaveCore

Delete remote slave core from replication.

URI: replication/slaves/{coreId}

HTTP Method: DELETE

DemandPairing

Instructs this core to send a replication demand to a remote core. This operation will require admin credentials on the remote core, but if successful will take effect right away. Returns slave core Id.

URI: replication/cores/pairing/demand

HTTP Method: POST

ForceReplication

Force replication for agents.

URI: replication/force

HTTP Method: PUT

GetAgentReplicationSettings

Get replication settings for agent with given ID.

URI: replication/agents/{agentId}

HTTP Method: GET

GetAgentRepositoryRelationships

Gets the repositories on a remote core for agents.

URI: replication/cores/slaves/{slaveCoreId}/agentRepositoryRelationships

HTTP Method: GET

GetCoreIdByDescriptor

Tests a core descriptor to validate the ability to connect to it. Returns CoreId. Using NTLM authentication.

URI: replication/cores/descriptor

HTTP Method: PUT

GetCoreIdByUrl

Tests connection to a remote core. Returns CoreId. Using Anonymous authentication.

URI: replication/cores/{hostUri}

HTTP Method: PUT

GetCountRemoteAgentsRecoveryPoints

Gets count of replicated recovery points on a remote slave core for the specific agent. This won't work with a master core.

URI: replication/cores/{coreId}/agents/{agentid}/recoveryPointsCount

HTTP Method: GET

GetPendingPairingRequest

Gets a the pending request for a specific request ID.

URI: replication/requests/pairing/pending/{requestId}

HTTP Method: GET

GetPendingPairingRequests

Gets a list of all pending replication pairing requests received by this core from remote master cores.

URI: replication/requests/pending

HTTP Method: GET

GetRemoteAgentsRecoveryPoints

Gets the replicated recovery points on a remote slave core for the specific agent. This won't work with a master core.

URI: replication/cores/{coreId}/agents/{agentid}/recoveryPoints

HTTP Method: GET

GetRemoteCoreRepositories

Gets the repositories on a remote core. Admin credentials on the remote core are required.

URI: replication/cores/slave/repositories

HTTP Method: PUT

GetRemoteCoreRepositoriesForDemand

Gets the repositories on a remote core for. Uses certificate authentication and works only for demanded core.

URI: replication/cores/slaves/{slaveCoreId}/pairingdemand/repositories

HTTP Method: PUT

GetRemoteCores

Gets a list of all of the remote cores this core knows about, both master and slave
URI: replication/cores/?forceRefresh={forceRefresh}
HTTP Method: GET

GetRemoteMasterCoresForDemand

Getting remote masers cores info for current slave core.
URI: replication/cores/slave/masters
HTTP Method: PUT

GetRemoteSlaveCoreReplicationPolicy

Gets remote slave core replication policy. This work with a master core side only.
URI: replication/cores/slaves/{slaveCoreId}/settings/policy
HTTP Method: GET

GetRemoteSlaveRecoveryPoint

Gets the details for a replicated recovery point on a remote slave core. This won't work with a master core.
URI: replication/cores/{coreId}/agents/{agentId}/rps/{recoveryPointId}
HTTP Method: GET

GetReplicatedAgentsRecoveryPointsInfo

Gets the list of agents which have recovery points on a remote slave core.
URI: replication/cores/{coreId}/agents/rpsinfo
HTTP Method: GET

GetReplicationConfiguration

Get replication configuration.
URI: replication/config
HTTP Method: GET

IgnorePairingRequest

Deletes a pending replication request without responding to it.
URI: replication/requests/{requestId}
HTTP Method: DELETE

RequestForceReplication

Request force replication.
URI: replication/requestForceReplication
HTTP Method: PUT

RequestPairing

Instructs this core to send a replication request to a remote core. Replication will start once the remote core approves the request. Returns slave core ID.

URI: replication/cores/pairing/request

HTTP Method: POST

RespondToAddAgentsByRequest

Responds to a pending agents from replication requests.

URI: replication/requests/pairing/{requestId}/agents

POST

RespondToPairingRequest

Responds to a pending replication requests.

URI: replication/requests/pairing/{requestId}

HTTP Method: POST

SelectRangeRemoteAgentsRecoveryPoints

Gets the range of replicated recovery points on a remote slave core for the specific agent. This won't work with a master core.

URI:
replication/cores/{coreId}/agents/{agentid}/skipCount/{skipCount}/maxCount/{maxCount}/recoveryPoints

HTTP Method: GET

SetAgentReplicationPauseConfiguration

Pauses replication for agent.

URI: replication/slaves/{slaveCoreId}/agents/{agentId}/pauseConfiguration

HTTP Method: POST

SetAgentReplicationPauseConfigurationForMasterCores

Pauses replication for agent.

URI: replication/masters/{masterCoreId}/agents/{agentId}/pauseConfiguration

HTTP Method: POST

SetAgentReplicationPauseConfigurationForSlaveCores

Pauses replication for agent.

URI: replication/slaves/agents/{agentId}/pauseConfiguration

HTTP Method: POST

SetAgentReplicationSettings

Set replication settings into an agent with given ID.

URI: replication/agents/{agentId}

HTTP Method: PUT

SetRemoteSlaveCoreReplicationPolicy

Sets remote slave core replication policy. This work with a master core side only.
URI: replication/cores/slaves/{slaveCoreId}/settings/policy
HTTP Method: PUT

SetReplicationConfiguration

Sets replication configuration.
URI: replication/config
HTTP Method: PUT

SwitchFailoverAgentToReplicated

Converts replicated agent to failover.
URI: replication/cores/{coreId}/agents/{agentId}/failover
HTTP Method: POST

SwitchReplicatedAgentToFailover

Converts failover agent to replicated agent.
URI: replication/cores/{coreId}/agents/{agentId}/failback?ignoreRunningReplicationJobs={ignoreReplication}
HTTP Method: POST

UpdateSlaveCoreSettings

Sets remote slave core configuration. This work with a master core side only.
URI: replication/cores/slaves/{slaveCoreId}/settings
HTTP Method: PUT

VerifyAddAgentsByDemand

Verifies whether agents can be safely replicated by demand.
URI: replication/cores/slave/agents/demand/verify
HTTP Method: POST

VerifyAddAgentsByDemandForExistingCore

Verifies whether agents can be safely replicated by demand.
URI: replication/cores/slave/{coreId}/agents/demand/verify
HTTP Method: POST

VerifyAddAgentsByRequest

Verifies whether agents can be safely replicated by request.
URI: replication/cores/slave/agents/request/verify
HTTP Method: POST

VerifyAddAgentsByRequestForExistingCore

Verifies whether agents can be safely replicated by request.

URI: replication/cores/slave/{coreId}/agents/request/verify

HTTP Method: POST

VerifyCorePairingAbilityByDemand

Tests a core descriptor to validate the ability to create pairing to remote core. Returns CoreId. Using NTLM authentication.

URI: replication/cores/verify/demand

HTTP Method: PUT

VerifyCorePairingAbilityByRequest

Tests a core descriptor to validate the ability to create pairing to remote core. Returns CoreId. Using anonymous authentication.

URI: replication/cores/verify/request/{hostUri}

HTTP Method: PUT

IReportingManagement

This section describes the service operations available for IReportingManagement at report/.

The IReportingManagement service contract is the interface implemented by core reports management service, which provides reporting functionality.

The URI and HTTP method are provided for each service operation.

The service operations include:

- GetAgentFailureReport
- GetAgentReport
- GetCoreFailureReport
- GetCoreReport
- GetDatabaseCoreIdentities
- GetSelfFailureReport
- GetSelfFailureReport
- GetSelfReport
- GetSummaryReport

GetAgentFailureReport

Gets a failure report for a particular agent

URI: report/agentFailureReport

HTTP Method: PUT

GetAgentReport

Gets a report for a particular agent

URI: report/agentReport

HTTP Method: PUT

GetCoreFailureReport

Gets a failure report for a particular set of cores

URI: report/coreFailureReport

HTTP Method: PUT

GetCoreReport

Gets a report for a particular set of cores.

URI: report/coreReport

HTTP Method: PUT

GetDatabaseCoreIdentities

Gets identity (id-name pairs) of Cores having reporting data in database currently used by the Core.

URI: report/databaseCoreIdentities

HTTP Method: GET

GetSelfFailureReport

Gets a failure report for this core.

URI: report/selfFailureReport

HTTP Method: PUT

GetSelfReport

Gets a report for this core.

URI: report/selfReport

HTTP Method: PUT

GetSummaryReport

Gets a summary report for this core.

URI: report/summaryReport

HTTP Method: PUT

IRepositoryManagement

This section describes the service operations available for IRepositoryManagement at reposManagement/.

The IRepositoryManagement service contract is the interface implemented by the repository management service, which handles core operations related to the repositories.

The URI and HTTP method are provided for each service operation.

The service operations include:

- AddExistentByConfigurations
- AddExistentWithIds
- AppendRepositoryFiles
- ApplyDefaultDeduplicationCacheConfiguration
- CheckRepository
- CheckRepositoryIntegrity
- CheckRepositoryIntegrityEstimate
- Create
- DeleteRepository
- GetConfiguration
- GetExistent
- GetFailedDirectories
- GetFreeDiskSpace
- GetRepositories
- GetRepositoriesCacheInfo
- GetRepositoryById
- GetRepositoryCacheInfo
- GetRepositoryFile
- GetRepositorySummaries
- GetRepositoryUsage
- IsRepoErrors
- IsRepositoryMounted
- MoveRepositoryFile
- ReformatRepository
- RenameRepository
- SetConfiguration
- SetRepositoryCacheParameters
- SetRepositoryFileSpecification
- UpdateRepositoriesDirectories
- UpdateRepositorySpecification
- UpdateWriteCachePolicy
- ValidateDirectoryPath
- VerifyFileSpecifications
- VerifyNetworkCredentials
- VerifyPaths

AddExistentByConfigurations

Loads one or several existing repositories.
URI: reposManagement/repositories/addexistentbyconfigurations
HTTP Method: POST

AddExistentWithIds

Load existent repositories by guid list.
URI: reposManagement/repositories/addexistentwithids
HTTP Method: POST

AppendRepositoryFiles

Appends and mounts additional files to a live repository.
URI: reposManagement/repositories/{repositoryId}/append
HTTP Method: POST

ApplyDefaultDeduplicationCacheConfiguration

Create default deduplication cache configuration.
URI: reposManagement/defaults
HTTP Method: PUT

CheckRepository

Checks a repository.
URI: reposManagement/repositories/check
HTTP Method: POST

CheckRepositoryIntegrity

Checks the repository integrity.
URI: reposManagement/repositories/checkRepositoryIntegrity
HTTP Method: POST

CheckRepositoryIntegrityEstimate

Estimates repository check job duration.
URI: reposManagement/repositories/{repositoryId}/checkRepositoryIntegrityEstimate
HTTP Method: GET

Create

Creates a new repository.
URI: reposManagement/repositories
HTTP Method: PUT

DeleteRepository

Deletes the specified repository and all data therein.
URI: reposManagement/repositories/{repositoryId}
HTTP Method: DELETE

GetConfiguration

Sets the configuration of the repository service.
URI: reposManagement/config
HTTP Method: GET

GetExistent

Shows the list of the existent repositories described by the configuration file at the specified directory.
URI: reposManagement/repositories/getexistent
HTTP Method: POST

GetFailedDirectories

Gets repositories directories' paths that do not pass validation.
URI: reposManagement/repositories/failedDirectories
HTTP Method: GET

GetFreeDiskSpace

Gets the free space in the directory or UNC share.
URI: reposManagement/getFreeDiskSpace
HTTP Method: POST

GetRepositories

Gets a list of all repositories, including configuration and status.
URI: reposManagement/repositories
HTTP Method: GET

GetRepositoriesCacheInfo

Gets all repositories cache info.
URI: reposManagement/repositories/getRepositoriesCacheInfo
HTTP Method: POST

GetRepositoryById

Gets the configuration and status information for a repository.
URI: reposManagement/repositories/{id}/id
HTTP Method: GET

GetRepositoryCacheInfo

Gets the repository cache info.
URI: reposManagement/repositories/{repositoryId}/getRepositoryCacheInfo
HTTP Method: POST

GetRepositoryFile

Gets the configuration and status information for a repository file.
URI: reposManagement/repositories/{repositoryId}/files/{fileId}
HTTP Method: GET

GetRepositorySummaries

Gets a list of all repositories, including configuration and status.
URI: reposManagement/repositorySummaries
HTTP Method: GET

GetRepositoryUsage

Gets the current uses of a specified repository.
URI: reposManagement/repositories/{id}/usage
HTTP Method: GET

IsRepoErrors

Checks if there any repository with errors. Returns true if at least one repository contains error(s).
URI: reposManagement/repositories/isRepoErrors
HTTP Method: GET

IsRepositoryMounted

Verify paths are reachable for a given repository.
URI: reposManagement/repositories/{repositoryId}/isRepositoryMounted
HTTP Method: POST

MoveRepositoryFile

Moves a repository file.
URI: reposManagement/repositories/move
HTTP Method: POST

ReformatRepository

Reformats the repository, deleting any existing recovery points.
URI: reposManagement/repositories/{repositoryId}/format
HTTP Method: POST

RenameRepository

Renames a repository.
URI: reposManagement/repositories/{repositoryId}/{newReposName}/rename
HTTP Method: POST

SetConfiguration

URI: Sets the configuration of the repository service.
reposManagement/config
HTTP Method: POST

SetRepositoryCacheParameters

Updates a repository cache parameters.
URI: reposManagement/repositories/{repositoryId}/setRepositoryCacheParameters
PUT

SetRepositoryFileSpecification

Updates a repository file specification.
URI: reposManagement/repositories/{repositoryId}/file/{repositoryFileId}
PUT

UpdateRepositoriesDirectories

Updates directories for repositories.
URI: reposManagement/repositories/updateDirectories
PUT

UpdateRepositorySpecification

Applies a new specification to a repository.
URI: reposManagement/repositories/updateRepositorySpecification
PUT

UpdateWriteCachePolicy

Updates files write caching policies.
URI: reposManagement/repositories/{repositoryId}/writePolicy
PUT

ValidateDirectoryPath

Perform validation of specified path of the repository directory.
URI: reposManagement/repositories/validateDirectoryPath
POST

VerifyFileSpecifications

Verifies paths and free space on the specified devices.

URI: reposManagement/repositories/verifyFileSpecifications

POST

VerifyNetworkCredentials

Verifies network credentials for specified paths.

URI: reposManagement/repositories/verifyNetworkCredentials

POST

VerifyPaths

Verify paths are reachable for a given repository.

URI: reposManagement/repositories/{repositoryId}/verifyPaths

POST

IRollbackManagement

This section describes the service operations available for IRollbackManagement at rollback/.

The IRollbackManagement service contract is the interface implemented by the core rollback service, which provides functionality from the Replay engine to restore data from recovery points.

NOTE: The restore functionality was formerly referred to as *rollback*. This term is deprecated.

The URI and HTTP method are provided for each service operation.

The service operations include:

- CheckJobByAgentId
- CheckJobByRrcIp
- GeneratePartitionPlan
- GetDatabasesForRemount
- GetSummaryMetadata
- GetTargetDisks
- GetTargetInfo
- GetTargetVolumes
- StartRollback
- ValidateStoragePoolSettings
- ValidateSystemVolumeExistence
- VerifyRollbackRequest

CheckJobByAgentId

Check state of roll-back job using agent id.

URI: rollback/checkJobByAgentId/{agentId}

HTTP Method: POST

CheckJobByRrcIp

Check state of roll-back job using ip address of RRC-agent.
URI: rollback/checkJobByRrcIp/{agentIp}
HTTP Method: POST

GeneratePartitionPlan

Generates a partition plan for a given recovery point, volume list, and target disks.
URI: rollback/autoPartition
HTTP Method: POST

GetDatabasesForRemount

Gets list of databases for remount along with warnings.
URI: rollback/databasesForRemount
HTTP Method: POST

GetSummaryMetadata

Gets the disks which are available on a given rollback (restore) target.
URI: rollback/summarymetadata
HTTP Method: PUT

GetTargetDisks

Gets the disks which are available on a given rollback (restore) target.
URI: rollback/targetDisks
HTTP Method: POST

GetTargetInfo

Gets the disks and volumes which are available on a given rollback (restore) target.
URI: rollback/targetInfo
HTTP Method: POST

GetTargetVolumes

Gets the volumes which are available on a given rollback (restore) target.
URI: rollback/targetVolumes
HTTP Method: POST

StartRollback

Starts a rollback (restore) job.
URI: rollback/target
HTTP Method: POST

ValidateStoragePoolSettings

Validates whether Storage Pool feature was enabled on the protected machine.

URI: rollback/storagepool

HTTP Method: POST

ValidateSystemVolumeExistence

Validate system volume existence.

URI: rollback/validateSystemVolumeExistence

HTTP Method: POST

VerifyRollbackRequest

Verify rollback (restore) request for volumes and disks.

URI: rollback/verifyRollbackRequest

HTTP Method: POST

IRollupManagement

This section describes the service operations available for IRollupManagement at rollup/.

The IRollupManagement service contract is the interface implemented by the core rollup management service, which provides rollup functionality from the Replay engine.

The URI and HTTP method are provided for each service operation.

The service operations include:

- ClearAgentRetentionPolicy
- ForceRollup
- GetAgentRetentionPolicy
- GetCoreRollupConfiguration
- GetGranularityIntervals
- SetAgentRetentionPolicy
- SetCoreRollupConfiguration

ClearAgentRetentionPolicy

Clears an agent's retention policy, and instead use the core-wide default one.

URI: rollup/agents/{agentId}

HTTP Method: DELETE

ForceRollup

Force rollup job for agent with given ID.

URI: rollup/agents/{agentId}/force

HTTP Method: POST

GetAgentRetentionPolicy

Gets retention policy for a given agent ID.

URI: rollup/agents/{agentId}

HTTP Method: GET

GetCoreRollupConfiguration

Gets core rollup configuration.

URI: rollup/config

HTTP Method: GET

GetGranularityIntervals

Gets granularity intervals for retention timeline.

URI: rollup/granularityIntervals

HTTP Method: PUT

SetAgentRetentionPolicy

Sets retention policy onto an agent with given ID.

URI: rollup/agents/{agentId}

HTTP Method: PUT

SetCoreRollupConfiguration

Sets core rollup configuration.

URI: rollup/config

HTTP Method: PUT

ISeedDriveManagement

This section describes the service operations available for ISeedDriveManagement at seedDrive/.

The ISeedDriveManagement service contract is the interface implemented by the seed drive management service, which provides copy-consume functionality from the Replay engine.

The URI and HTTP method are provided for each service operation.

The service operations include:

- AbandonOutstandingSeedDrive
- GetConsumedSeedDrives
- GetOutstandingSeedDrives
- GetSeedDriveManifest
- StartConsumeSeedDrive
- StartCopySeedDrive

AbandonOutstandingSeedDrive

Deletes seed drive from the list of seed drives which are waiting to be consumed.

URI: seedDrive/outstandingSeedDrives/{seedDriveId}

HTTP Method: DELETE

GetConsumedSeedDrives

Gets info about seed drives consumed by specified replicated agent on the slave core.

URI: seedDrive/consumedSeedDrives/{agentId}

HTTP Method: GET

GetOutstandingSeedDrives

Gets outstanding seed drives on master core which are waiting to be consumed on slave cores.

URI: seedDrive/outstandingSeedDrives

HTTP Method: GET

GetSeedDriveManifest

Gets the metadata for an existing seed drive by the core ID.

URI: seedDrive/metadataByCore/{coreId}

HTTP Method: POST

StartConsumeSeedDrive

Starts consuming recovery points data from user-specified seed drive.

URI: seedDrive/consume

HTTP Method: POST

StartCopySeedDrive

Starts copying recovery points data to user-specified seed drive.

URI: seedDrive/copy

HTTP Method: POST

IServiceHostManagement

This section describes the service operations available for IServiceHostManagement at servicehost/.

The IServiceHostManagement service contract is the interface implemented by a class which provides a management interface to the IServiceHost.

The URI and HTTP method are provided for each service operation.

The service operations include:

- GetApiVersionInfo
- GetConfiguration
- Restart
- SetConfiguration
- VerifyConnection

GetApiVersionInfo

Obtains the release version information for the API.

URI: servicehost/apiVersion

HTTP Method: GET

GetConfiguration

Gets current configuration of a server that listens for incoming REST calls.

URI: servicehost/config

HTTP Method: GET

Restart

Immediately restarts a server that listens for incoming REST calls.

URI: servicehost/restart

HTTP Method: GET

SetConfiguration

Sets current configuration of a server that listens for incoming REST calls.

URI: servicehost/config

HTTP Method: POST

VerifyConnection

Allows to verify if listening server is configured properly and able to receive incoming REST calls.

URI: servicehost/verify

HTTP Method: GET

ISqlManagement

This section describes the service operations available for ISqlManagement at sql/.

The ISqlManagement service contract is the interface implemented by core sql management service which provides attachability checks and other core-specific SQL management functionality.

The URI and HTTP method are provided for each service operation.

The service operations include:

- ForceAttachability
- GetAttachabilitySettings
- GetCoreSqlInstances
- SetAttachabilitySettings
- TestSqlConnection

ForceAttachability

Force attachability job for agent with given ID.

URI: sql/recoveryPoints/{recoveryPointId}/force

HTTP Method: POST

GetAttachabilitySettings

Gets core-level attachability settings.

URI: sql/attachabilitySettings

HTTP Method: GET

GetCoreSqlInstances

Gets core-level attachability settings.

URI: sql/coreSqlInstances

HTTP Method: GET

SetAttachabilitySettings

Gets core-level attachability settings.

URI: sql/attachabilitySettings

HTTP Method: PUT

TestSqlConnection

Sets core-level attachability settings.

URI: sql/connection

HTTP Method: PUT

IStatusSummaryManagement

This section describes the service operations available for IStatusSummaryManagement at status/.

The IStatusSummaryManagement service contract exposes summarized status info about multiple services in the core for use displaying status icons in the GUI tabs.

The URI and HTTP method are provided for each service operation.

The service operations include:

- Crash
- GetAgentDashboardInfo
- GetCoreEventsInfo
- GetCoreHomeInfo
- GetCoreSystemInfo
- GetDashboardInfo
- GetFailedServicesInfo
- GetLatestEvents
- GetStatusSummary
- RetryStartFailedServices

Crash

Crashes the process.

URI: status/crash

HTTP Method: GET

GetAgentDashboardInfo

Get the status info for a specific Agent.
URI: status/dashboard/{agentId}
HTTP Method: GET

GetCoreEventsInfo

Gets latest core events info such as events accumulated since last call to itself or start of core if call is first.
URI: status/coreEventsInfo
HTTP Method: POST

GetCoreHomeInfo

Gets core metadata and recent 10 alerts.
URI: status/coreHomeInfo
HTTP Method: GET

GetCoreSystemInfo

Gets core system info such as mounts, metadata, settings, repository configuration, and so on.
URI: status/coreSystemInfo
HTTP Method: GET

GetDashboardInfo

Get the status info necessary to display the dashboard on the main screen.
URI: status/dashboard
HTTP Method: GET

GetFailedServicesInfo

Gets names of services failed to initialize.
URI: status/failedServicesInfo
HTTP Method: GET

GetLatestEvents

Gets latest core events info such as events accumulated since last call to itself or start of core if call is first.
URI: status/latestEventsInfo
HTTP Method: GET

GetStatusSummary

Get a summary of the status of all services including in the status summary API.
URI: status/statusSummary
HTTP Method: GET

RetryStartFailedServices

Tries to restart failed services.
URI: status/restart/
HTTP Method: POST

ITransferQueueManagement

This section describes the service operations available for ITransferQueueManagement at xfer/queue/.

The ITransferQueueManagement service contract is the RESTful API for the transfer queue.

The URI and HTTP method are provided for each service operation.

The service operations include:

- AdjustTransferPriority
- CancelAllTransfers
- CancelTransfer
- GetActiveQueueEntries
- GetEntryInfo
- GetPendingQueueEntries
- GetQueueConfiguration
- GetQueueContents
- SetQueueConfiguration

AdjustTransferPriority

Adjusts the priority of the transfer in the queue.
URI: xfer/queue/entries/{transferid}
HTTP Method: POST

CancelAllTransfers

Cancels every transfer in the queue.
URI: xfer/queue/entries
HTTP Method: DELETE

CancelTransfer

Cancels the transfer queue entry identified by the transfer ID.
URI: xfer/queue/entries/{transferid}
HTTP Method: DELETE

GetActiveQueueEntries

Gets the active entries in the transfer queue.
URI: xfer/queue/activeEntries
HTTP Method: GET

GetEntryInfo

Gets the info for a specific transfer queue entry.
URI: xfer/queue/entries/{transferid}
HTTP Method: GET

GetPendingQueueEntries

Gets the pending entries in the transfer queue.
URI: xfer/queue/pendingEntries
HTTP Method: GET

GetQueueConfiguration

Gets the configuration of the transfer queue.
URI: xfer/queue/config
HTTP Method: GET

GetQueueContents

Gets the contents of the transfer queue.
URI: xfer/queue/entries
HTTP Method: GET

SetQueueConfiguration

Sets the configuration of the transfer queue.
URI: xfer/queue/config
HTTP Method: POST

ITransferSchedulerManagement

This section describes the service operations available for ITransferSchedulerManagement at xfer/schedule/. The ITransferSchedulerManagement service contract is the RESTful API for the transfer scheduler.

The URI and HTTP method are provided for each service operation.

The service operations include:

- AddScheduleTemplate
- ApplyRepositoryToUnprotectedAgents
- BatchAgentsProtectionPaused
- BatchAgentsProtectionResumed
- DeleteScheduleTemplate
- ForceAllTransfer
- ForceTransfer
- GetActualAgentProtectionConfiguration
- GetAgentConfiguration
- GetCachedAgentProtectionConfiguration
- GetScheduleTemplate

- GetScheduleTemplates
- RemoveRepositoryFromProtection
- SetAgentConfiguration
- SetAgentProtectionConfiguration
- UpdateScheduleTemplate

AddScheduleTemplate

Gets the transfer schedule templates collection.
URI: xfer/schedule/transferScheduleTemplates
HTTP Method: POST

ApplyRepositoryToUnprotectedAgents

Applies a new repository to agents that do not have a repository.
URI: xfer/schedule/applyrepository/{repositoryName}
HTTP Method: PUT

BatchAgentsProtectionPaused

Pauses protection of several agents.
URI: xfer/schedule/agentsPaused
HTTP Method: POST

BatchAgentsProtectionResumed

Resumes protection of several agents.
URI: xfer/schedule/agentsResumed
HTTP Method: POST

DeleteScheduleTemplate

Removes transfer schedule template.
URI: xfer/schedule/transferScheduleTemplates/{templateId}
HTTP Method: DELETE

ForceAllTransfer

Forces an immediate transfer for all available agents.
URI: xfer/schedule/forcealltransfer
HTTP Method: POST

ForceTransfer

Forces an immediate transfer of the specified protection group.
URI: xfer/schedule/{agentId}/transfer
HTTP Method: POST

GetActualAgentProtectionConfiguration

Gets the actual protection groups configured for the agent.
URI: xfer/schedule/{agentId}/actualProtectionConfiguration
HTTP Method: GET

GetAgentConfiguration

Gets the transfer configuration settings for the agent.
URI: xfer/schedule/{agentId}/config
HTTP Method: GET

GetCachedAgentProtectionConfiguration

Sets the protection groups configured for the agent.
URI: xfer/schedule/{agentId}/pgs
HTTP Method: GET

GetScheduleTemplate

Gets the transfer schedule template by id.
URI: xfer/schedule/transferScheduleTemplates/{templateId}
HTTP Method: GET

GetScheduleTemplates

Gets the transfer schedule templates collection.
URI: xfer/schedule/transferScheduleTemplates
HTTP Method: GET

RemoveRepositoryFromProtection

Disassociates agents from a named repository (for example, one that has been deleted).
URI: xfer/schedule/removerepository/{repositoryName}
HTTP Method: DELETE

SetAgentConfiguration

Sets the transfer configuration settings for the agent.
URI: xfer/schedule/{agentId}/config
HTTP Method: PUT

SetAgentProtectionConfiguration

Sets the protection groups configured for the agent.
URI: xfer/schedule/{agentId}/pgs
HTTP Method: PUT

UpdateScheduleTemplate

Updates existing transfer schedule template.

URI: xfer/schedule/transferScheduleTemplates/{templateId}

HTTP Method: PUT

IUtilitiesManagement

This section describes the service operations available for IUtilitiesManagement at utilities/. The IUtilitiesManagement service contract is the API for variety helpers.

The URI and HTTP method are provided for each service operation.

The service operations include:

- GetDomainInfo
- GetMachinesFromActiveDirectory
- GetMachinesFromActiveDirectoryByPage

GetDomainInfo

Gets domain information.

URI: utilities/domainInformation

HTTP Method: POST

GetMachinesFromActiveDirectory

Gets all machines from Active Directory.

URI: utilities/activeDirectoryMachines

HTTP Method: POST

GetMachinesFromActiveDirectoryByPage

Gets machines by specific page from Active Directory.

URI: utilities/activeDirectoryMachinesByPage

HTTP Method: POST

IVirtualDiskManagement

This section describes the service operations available for IVirtualDiskManagement at vhd/. The URI and HTTP method are provided for each service operation.

The service operations include:

- BaseFileName
- BeginBatch
- Close
- Create
- Delete
- EndBatch
- HasSnapshot

- Open
- Read
- ReadCustomMetadata
- SectorSize
- SnapshotFileName
- TakeSnapshot
- TotalSectorCapacity
- TranslateSectorOffsetToChsTuple
- Write
- WriteCustomMetadata

BaseFileName

Gets VHD base file name.

URI: vhd/{id}/baseFileName

HTTP Method: GET

BeginBatch

Begins batch.

URI: vhd/{id}/beginBatch/{target}

HTTP Method: POST

Close

Closes VHD.

URI: vhd/{id}/close

HTTP Method: GET

Create

Creates VHD.

URI:
vhd/createVhd?path={path}&bytesCapacity={bytesCapacity}&bytesPerSector={bytesPerSector}&containsBootSystemVolume={containsBootSystemVolume}&preallocate={preallocate}

HTTP Method: PUT

Delete

Deletes VHD snapshot or base file.

URI: vhd/{id}/delete/{target}

HTTP Method: POST

EndBatch

Ends batch.

URI: vhd/{id}/endBatch

HTTP Method: POST

HasSnapshot

Verifies VHD has snapshot.
URI: vhd/{id}/hasSnapshot
HTTP Method: GET

Open

Opens VHD.
URI: vhd/openVhd?path={path}
HTTP Method: PUT

Read

Writes raw data to VHD.
URI: vhd/{id}/read/{target}/{sectorOffset}/{sectorLength}
HTTP Method: POST

ReadCustomMetadata

Reads a user-defined custom metadata string.
URI: vhd/{id}/readCustomMetadata/{target}/{key}
HTTP Method: POST

SectorSize

Gets sector size of the VHD.
URI: vhd/{id}/sectorSize
HTTP Method: GET

SnapshotFileName

Gets VHD snapshot file name.
URI: vhd/{id}/snapshotFileName
HTTP Method: GET

TakeSnapshot

Takes VHD snapshot.
URI: vhd/{id}/takeSnapshot
HTTP Method: GET

TotalSectorCapacity

Gets VHD capacity.
URI: vhd/{id}/totalSectorCapacity
HTTP Method: GET

TranslateSectorOffsetToChsTuple

Translates sector offset to chs tuple.

URI: vhd/{id}/translateSectorOffsetToChsTuple?sectorOffset={sectorOffset}

HTTP Method: GET

Write

Writes raw data to VHD.

URI: vhd/{id}/write/{target}/{sectorOffset}/{sectorLength}

HTTP Method: POST

WriteCustomMetadata

Reads a user-defined custom metadata string.

URI: vhd/{id}/writeCustomMetadata/{target}/{key}?value={value}

HTTP Method: POST

IWhiteLabelingManagement

This section describes the service operations available for IWhiteLabelingManagement at whitelabeling/. The IWhiteLabelingManagement service contract exposes the customizable strings.

The URI and HTTP method are provided for the service operation is as follows:

- GetWhiteLabelingInfo

GetWhiteLabelingInfo

Gets customizable strings.

URI: whitelabeling/whiteLabelingInfo

HTTP Method: GET

Using AppAssure Agent API

The AppAssure Agent API is grouped according to the following categories/service interfaces. Click a category below to get more information about the service operations available for a particular interface.

The service contracts contained in the Agent client assembly are described in the following table:

Table 2. Agent service contracts

Service Contract	Uri	Description
IAgentMetadataManagement	metadata/	Interface implemented by the agent metadata service, which maintains, caches, and returns agent metadata.
IAgentPairManagement	pair/	Interface implemented by the agent pairing service, which provides pairing functionality on the agents.
IAgentServiceHostManagement	agentServiceHost/	Interface implemented by the Agent service, which performs actions related to the Agent's ServiceHost.

Table 2. Agent service contracts

Service Contract	Uri	Description
IAgentSettingsManagement	settings/	Provides a way to query and set assorted agent-wide settings.
IAgentUpdateManagement	update/	Interface implemented by the agent update service, which downloads and installs updates.
IApplicationIdManagement	id/	Exposes the application's unique ID.
IDiagnosticsManagement	diag/	Replay.Common.Contracts.Diagnostics.IDiagnosticsManagement
IDriverChangeLogsManagement	driverchangelogs/	Provides an ability to manage AAFsFlt change logs: enumerate, delete, control volume enablement.
IExchangeManagement	exchangedll/	Provides possibility to upload MS Exchange DLLs to Core.
IExchangeServerManagement	exchangeServer/	Provides possibility to run the command over the Exchange Server installed on the physical machine.
IHyperVAgentManagement	hypervagent/	Replay.Common.Contracts.Virtualization.IHyperVAgentManagement.
IPowerShellManagement	powerShell/	Interface allows to run the script with parameters from file.
IRollbackManagement	rollback/	REST service which exposes methods to initiate restoring data from a recovery point on an agent.
IRrcRollbackManagement	rollback/	REST service which exposes methods to initiate restoring data from a recovery point on an agent.
IServiceHostManagement	servicehost/	Interface implemented by a class which provides a management interface to the IServiceHost.
IShadowCopyManagement	shadowcopy/	Exposes some diagnostic information about the Volume Shadow Copy service (VSS).
ITransferManagement	transfer/	WCF contract interface implemented by a class which implements the management interface on top of the transfer service.
IVirtualDiskManagement	vhd/	Replay.Common.Contracts.Virtualization.IVirtualDiskManagement
IWhiteLabelingManagement	whitelabeling/	Exposes the customizable strings.

IAgentMetadataManagement

This section describes the service operations available for IAgentMetadataManagement at metadata/.

The IAgentMetadataManagement interface is implemented by the agent metadata service, which maintains, caches, and returns agent metadata.

The URI and HTTP method are provided for each service operation.

The service operations include:

- GetCurrent
- GetCurrentCluster
- GetCurrentClusterSummary

- GetCurrentDeduplicationJobs
- GetCurrentSummary
- VerifyCredentials

GetCurrent

Gets cached full metadata for the agent.
URI: metadata/fullMetadata
HTTP Method: PUT

GetCurrentCluster

Gets cached full metadata for the agent.
URI: metadata/fullClusterMetadata
HTTP Method: PUT

GetCurrentClusterSummary

Gets cached full metadata for the agent.
URI: metadata/fullClusterSummaryMetadata
HTTP Method: PUT

GetCurrentDeduplicationJobs

Gets the list of currently active deduplication jobs.
URI: metadata/currentDeduplicationJobs
HTTP Method: GET

GetCurrentSummary

Gets cached summary metadata for the agent with MS Exchange and SQL servers metadata.
URI: metadata/summaryMetadata
HTTP Method: PUT

VerifyCredentials

Verifies specified credentials for MS Exchange and SQL servers.
URI: metadata/credentials
HTTP Method: PUT

IAgentPairManagement

This section describes the service operations available for IAgentPairManagement at pair/.

The IAgentPairManagement interface is implemented by the agent pairing service, which provides pairing functionality on the agents.

The URI and HTTP method are provided for each service operation. The service operations include:

- GetPairingSettings
- Pair

- RemovePairing
- VerifyConnect
- VerifyConnectWithOptionalAuthentication

GetPairingSettings

Gets pairing settings of the Agent which includes agent ID, paired Core name and security certificate thumbprints.

URI: pair/

HTTP Method: GET

Pair

Sets up relationship between a Core and the Agent including security certificates exchange.

URI: pair/

HTTP Method: PUT

RemovePairing

Removes relationship between the Agent and a Core.

URI: pair/
HTTP Method: DELETE

VerifyConnect

Allows to verify connection to specified agent and returns actual agent ID.

URI: pair/connect

HTTP Method: GET

VerifyConnectWithOptionalAuthentication

Allows to verify connection to specified agent with optional authentication scheme and returns actual agent ID.

URI: pair/connect/?useNtlmOnly={useNtlmOnly}

HTTP Method: GET

IAgentServiceHostManagement

This section describes the service operations available for IAgentServiceHostManagement at agentServiceHost/.

The IAgentServiceHostManagement interface is implemented by the Agent service, which performs actions related to the Agent's ServiceHost.

The URI and HTTP method are provided for each service operation is as follows:

- ChangePort

ChangePort

Changes port number used by server which listens for incoming REST calls.

URI: agentServiceHost/port

HTTP Method: POST

IAgentSettingsManagement

This section describes the service operations available for IAgentSettingsManagement at settings/.

The IAgentSettingsManagement service contract provides a way to query and set assorted agent-wide settings.

The URI and HTTP method are provided for each service operation. The service operations include:

- GetAgentSettings
- SetAgentSettings

GetAgentSettings

Gets general global Agent settings such as timeout settings and display name.

URI: settings/agent

HTTP Method: GET

SetAgentSettings

Applies general global Core settings such as timeout settings and display name.

URI: settings/agent

HTTP Method: PUT

IAgentUpdateManagement

This section describes the service operations available for IAgentUpdateManagement at update/.

The IAgentUpdateManagement interface is implemented by the agent update service, which downloads and installs updates.

The URI and HTTP method are provided for each service operation is as follows:

- ApplyUpdate

ApplyUpdate

Performs automatic update of Agent service binaries. This method is for internal usage only and should not be called by users.

URI: update/

HTTP Method: POST

IApplicationIdManagement

This section describes the service operations available for IApplicationIdManagement at id/.

The IApplicationIdManagement exposes the application's unique ID.

The URI and HTTP method are provided for each service operation is as follows:

- GetId

GetId

Gets the core's ID.

URI: id/

HTTP Method: GET

IDiagnosticsManagement

This section describes the service operations available for IDiagnosticsManagement at diag/.

The IDiagnosticsManagement service contract is represented as Replay.Common.Contracts.Diagnostics.IDiagnosticsManagement.

The URI and HTTP method are provided for each service operation. The service operations include:

- ExecuteRemoteCommand
- GetLog
- GetLogSession
- ReadFile
- RestartService
- UploadLogSessions

ExecuteRemoteCommand

Executes an arbitrary remote command.

URI: diag/command/

HTTP Method: POST

GetLog

Gets the entire contents of the replay.log file.

URI: diag/log/

HTTP Method: GET

GetLogSession

Packages the current log session and returns it as a byte stream. The contents of the stream is a Gibraltar (.glp file).

URI: diag/logSession/

HTTP Method: GET

ReadFile

Reads a file from the local file system and streams it back to the client.

URI: diag/files/?q={path}

HTTP Method: GET

RestartService

Stops, forcibly kills (if necessary), and re-starts the service.

URI: diag/service/

HTTP Method: DELETE

UploadLogSessions

Uploads the current log session to the Gibraltar (http://www.gibraltarsoftware.com/) logging framework.

URI: diag/logSession/

HTTP Method: POST

IDriverChangeLogsManagement

This section describes the service operations available for IDriverChangeLogsManagement at driverchangelogs/.

The IDriverChangeLogsManagement provides an ability to manage AAFsFlt change logs: enumerate, delete, control volume enablement.

The URI and HTTP method are provided for each service operation. The service operations include:

- DeleteDriverChangeLogFile
- DisableVolumeLogging
- EnableVolumeLogging
- EnumerateDriverChangeLogs

DeleteDriverChangeLogFile

Deletes a specified change log file for a volume.

URI: driverchangelogs/{volumeName}/{changeLogFile}

HTTP Method: DELETE

DisableVolumeLogging

Disables logging FS changes for a volume.

URI: driverchangelogs/{volumeName}/disablelogging

HTTP Method: POST

EnableVolumeLogging

Enables logging FS changes for a volume.

URI: driverchangelogs/{volumeName}/enablelogging

HTTP Method: POST

EnumerateDriverChangeLogs

Gets a collection of change log files for a volume.

URI: driverchangelogs/{volumeName}/

HTTP Method: GET

IExchangeManagement

This section describes the service operations available for IExchangeManagement at exchangedll/.

The IExchangeManagement provides possibility to upload MS Exchange DLLs to Core.

The URI and HTTP method are provided for each service operation. The service operations include:

- CancelTransmitFile
- ContinueTransmitFile
- EndTransmitFile
- GetExchangeDllInfo
- StartNewFileTransmitSession

CancelTransmitFile

Cancels current transmit session.
URI: exchangedll/sessions/{fileTransmitSessionId}/
HTTP Method: DELETE

ContinueTransmitFile

Reads data from agent in current transmit session.
URI: exchangedll/sessions/{fileTransmitSessionId}/data/{bytesToRead}
HTTP Method: POST

EndTransmitFile

Ends current transmit session.
URI: exchangedll/sessions/{fileTransmitSessionId}/
HTTP Method: POST

GetExchangeDllInfo

Gets information about MS Exchange library.
URI: exchangedll/{fileName}/
HTTP Method: GET

StartNewFileTransmitSession

Starts new file transmit session.
URI: exchangedll/newsession/
HTTP Method: POST

IExchangeServerManagement

This section describes the service operations available for IExchangeServerManagement at exchangeServer/.

The IExchangeServerManagement provides possibility to execute command over the Exchange Server installed on the physical machine.

The URI and HTTP method are provided for each service operation. The service operations include:

- CancelExchangeServerCommand
- ExecuteExchangeServerCommand
- GetExchangeServerCommandStatus

CancelExchangeServerCommand

Cancels the running command.
URI: exchangeServer/cancelCommand
HTTP Method: DELETE

ExecuteExchangeServerCommand

Gracefully performs start/stop of the Exchange Server.
URI: exchangeServer/executeCommand
HTTP Method: PUT

GetExchangeServerCommandStatus

Queries the status of the current start/stop operation of the Exchange Server.
URI: exchangeServer/getCommandStatus
HTTP Method: GET

IHyperVAgentManagement

This section describes the service operations available for IHyperVAgentManagement at hypervagent/.

The IHyperVAgentManagement service contract is represented as Replay.Common.Contracts.Virtualization.IHyperVAgentManagement.

The URI and HTTP method are provided for each service operation. The service operations include:

- AddDvdDrive
- AddIsoImage
- AddNetworkAdapter
- AttachVirtualDisk
- DeleteVirtualMachine
- DetachVirtualDisk
- EndSession
- GetAvailableVirtualNetworks
- GetMaximumProcessorCount
- GetOrCreateVirtualMachineAndAttach
- GetSnapshotsCount
- GetVirtualDisks
- GetVirtualMachineName
- InsertIntegrationServices
- PingSession

- RenameVirtualMachine
- SetProcessorCount
- SetRamMegabytes
- VerifyConnection

AddDvdDrive

Adds a DVD drive to current virtual machine.
URI: hypervagent/{virtualMachineId}/dvd
HTTP Method: PUT

AddIsoImage

Adds an ISO image to a DVD drive. If DVD drive doesn't exist - creates it.
URI: hypervagent/{virtualMachineId}/iso/{isoPath}
HTTP Method: PUT

AddNetworkAdapter

Adds new network adapter to current virtual machine.
URI: hypervagent/{virtualMachineId}/nic/{networkAdapterName}
HTTP Method: PUT

AttachVirtualDisk

Attaches a virtual disk to current virtual machine.
URI: hypervagent/{virtualMachineId}/disks/{diskPath}/{storageController}
HTTP Method: PUT

DeleteVirtualMachine

Deletes current virtual machine and detaches from it.
URI: hypervagent/existent/{virtualMachineId}
HTTP Method: DELETE

DetachVirtualDisk

Detaches a virtual disk from current virtual machine.
URI: hypervagent/{virtualMachineId}/disks/{diskPath}
HTTP Method: DELETE

EndSession

Tells Hyper-V Agent to finish session with the virtual machine.
URI: hypervagent/{virtualMachineId}/endsession
HTTP Method: DELETE

GetAvailableVirtualNetworks

Gets a list of virtual network adapters on Hyper-V server available for a virtual machine.
URI: hypervagent/{virtualMachineId}/availablevirtualnetworks
HTTP Method: GET

GetMaximumProcessorCount

Gets the maximum number of virtual CPUs that could be attached to virtual machine.
URI: hypervagent/cpu/{operatingSystemFamily}
HTTP Method: GET

GetOrCreateVirtualMachineAndAttach

Gets or creates virtual machine specified in request parameter.
URI: hypervagent/getOrCreate
HTTP Method: POST

GetSnapshotsCount

Get the count of snapshots for virtual machine.
URI: hypervagent/{virtualMachineId}/getSnapshotsCount
HTTP Method: GET

GetVirtualDisks

Gets a list of virtual disks currently attached to current virtual machine.
URI: hypervagent/{virtualMachineId}/disks
HTTP Method: GET

GetVirtualMachineName

Gets name of current virtual machine.
URI: hypervagent/{virtualMachineId}/name
HTTP Method: GET

InsertIntegrationServices

Mounts the integration services setup disk.
URI: hypervagent/{virtualMachineId}/integrationservices
HTTP Method: PUT

PingSession

Indicates that session is still being used.
URI: hypervagent/{virtualMachineId}/pingSession
HTTP Method: PUT

RenameVirtualMachine

Renames current virtual machine.

URI: hypervagent/{virtualMachineId}/newname/{newVirtualMachineName}

HTTP Method: PUT

SetProcessorCount

Changes number of virtual CPUs in current virtual machine.

URI: hypervagent/{virtualMachineId}/cpu/{processorCount}

HTTP Method: PUT

SetRamMegabytes

Changes amount of RAM in current virtual machine.

URI: hypervagent/{virtualMachineId}/ram/{ramValue}

HTTP Method: PUT

VerifyConnection

Verifies connection to the running HyperV agent.

URI: hypervagent/connect

HTTP Method: GET

IPowerShellManagement

This section describes the service operations available for IPowerShellManagement at powerShell/.

The IPowerShellManagement interface allows to execute script with parameters from file.

The URI and HTTP method are provided for each service operation is as follows:

- RunPowerShellScriptFromFile

RunPowerShellScriptFromFile

Runs PowerShell script on the agent.

URI: powerShell/

HTTP Method: POST

IRollbackManagement

This section describes the service operations available for IRollbackManagement at rollback/.

The IRollbackManagement REST service which exposes methods to initiate restoring data from a recovery point on an agent.

The URI and HTTP method are provided for each service operation. The service operations include:

- CancelRollback
- DeterminateAutomaticallyAcquiringReplayEngineAddress
- GetRollbackMountDismountStatus
- PartitionDisks

- StartDismounting
- StartMounting
- StartRollback

CancelRollback

Cancels the running rollback (restore) operation.

URI: rollback/

HTTP Method: DELETE

DeterminateAutomaticallyAcquiringReplayEngineAddress

Determinate if agent can automatically determine acquiring ReplayEngine Address.

URI: rollback/determinateAutomaticallyAcquiringReplayEngineAddress

HTTP Method: GET

GetRollbackMountDismountStatus

Queries the status of the current rollback (restore) operation.

URI: rollback/

HTTP Method: GET

PartitionDisks

Performs automatic disks partitioning basing on partition information in a request object. All existing partitions are removed.

URI: rollback/partition

HTTP Method: POST

StartDismounting

Dismount list of databases.

URI: rollback/dismount

HTTP Method: PUT

StartMounting

Mounts a list of databases.

URI: rollback/mount

HTTP Method: PUT

StartRollback

Initiates a rollback (restore) operation on the agent.

URI: rollback/

HTTP Method: POST

IRrcRollbackManagement

This section describes the service operations available for IRrcRollbackManagement at rollback/.

The IRrcRollbackManagement REST service which exposes methods to initiate rollbacks (restoring data from a recovery point) on an agent.

The URI and HTTP method are provided for each service operation is as follows:

- FixBootable

FixBootable

Performs post processing of boot volume in order to correct boot manager and make volume bootable after bare metal restore. This method is for internal usage only and should not be called by users.

URI: rollback/fixBoot

HTTP Method: PUT

IServiceHostManagement

This section describes the service operations available for IServiceHostManagement at servicehost/.

The IServiceHostManagement interface is implemented by a class which provides a management interface to the IServiceHost.

The URI and HTTP method are provided for each service operation. The service operations include:

- GetApiVersionInfo
- GetConfiguration
- Restart
- SetConfiguration
- VerifyConnection

GetApiVersionInfo

Obtains the release version information for the API.

URI: servicehost/apiVersion

HTTP Method: GET

GetConfiguration

Gets current configuration of a server that listens for incoming REST calls.

URI: servicehost/config

HTTP Method: GET

Restart

Immediately restarts a server that listens for incoming REST calls.

URI: servicehost/restart

HTTP Method: GET

SetConfiguration

Sets current configuration of a server that listens for incoming REST calls.

URI: servicehost/config

HTTP Method: POST

VerifyConnection

Allows to verify if listening server is configured properly and able to receive incoming REST calls.

URI: servicehost/verify

HTTP Method: GET

IShadowCopyManagement

This section describes the service operations available for IShadowCopyManagement at shadowcopy/.

The IShadowCopyManagement service contract exposes some diagnostic information about the Volume Shadow Copy service (VSS).

The URI and HTTP method are provided for each service operation is as follows:

- GetWriters

GetWriters

Gets detailed information about Volume Shadow Copy Service (VSS) writers installed on the Agent.

URI: shadowcopy/writers/

HTTP Method: GET

ITransferManagement

This section describes the service operations available for ITransferManagement at transfer/.

The ITransferManagement is a WCF contract interface that is implemented by a class which in turn, implements the management interface on top of the transfer service.

The URI and HTTP method are provided for each service operation. The service operations include:

- DeleteChangeLogsAndMarkClean
- DeleteSnapshot
- FindIncompatibleProducts
- GetDriverMetadata
- GetShadowCopyFile
- GetVolumeAllocatedBlocks
- GetVolumeChangedBlocks
- GetVolumePhysicalDisks
- KeepSessionAlive
- SetDriverMetadata
- SetGlobalVolumeEnablement
- SetVolumeEnablement
- TakeSnapshot

DeleteChangeLogsAndMarkClean

Deletes the Replay change logs for the volume.
URI: transfer/snapshots/{snapshotSetId}/volumes/{volumeName}/logs
HTTP Method: DELETE

DeleteSnapshot

Deletes the snapshot.
URI: transfer/snapshots/{snapshotSetId}
HTTP Method: POST

FindIncompatibleProducts

Finds and returns the installed incompatible products.
URI: transfer/incompatibleProducts
HTTP Method: POST

GetDriverMetadata

Gets the driver metadata for a volume in the snapshot.
URI: transfer/snapshots/{snapshotSetId}/volumes/{volumeName}/driver
HTTP Method: GET

GetShadowCopyFile

Gets information about shadow copy files that were created while taking vss snapshot.
URI: transfer/snapshots/{snapshotSetId}/volumes/{volumeName}/shadowCopyFile
HTTP Method: GET

GetVolumeAllocatedBlocks

Gets a binary representation of the list of allocated blocks on the volume.
URI: transfer/snapshots/{snapshotSetId}/volumes/{volumeName}/blocks/allocated
HTTP Method: GET

GetVolumeChangedBlocks

Gets a binary representation of the list of changed blocks on the volume.
URI: transfer/snapshots/{snapshotSetId}/volumes/{volumeName}/blocks/changed
HTTP Method: GET

GetVolumePhysicalDisks

Gets information about the volume's physical layout on physical disk(s).
URI: transfer/snapshots/{snapshotSetId}/volumes/{volumeName}/disks
HTTP Method: GET

KeepSessionAlive

Keep the session alive.

URI: transfer/session/retention

HTTP Method: POST

SetDriverMetadata

Sets the driver metadata for a volume in the snapshot. Only sets the metadata values which are not set to their default values.

URI: transfer/snapshots/{snapshotSetId}/volumes/{volumeName}/driver

HTTP Method: POST

SetGlobalVolumeEnablement

Enables or disables a volume. This method works globally since no snapshot ids are needed for it.

URI: transfer/volumes/{volumeName}/enablement

HTTP Method: POST

SetVolumeEnablement

Enables or disables a volume.

URI: transfer/snapshots/{snapshotSetId}/volumes/{volumeName}/enablement

HTTP Method: POST

TakeSnapshot

Takes a snapshot of one or more volumes on the agent.

URI: transfer/snapshots

HTTP Method: POST

IVirtualDiskManagement

This section describes the service operations available for IVirtualDiskManagement at vhd/.

The IVirtualDiskManagement is represented as Replay.Common.Contracts.Virtualization.IVirtualDiskManagement.

The URI and HTTP method are provided for each service operation. The service operations include:

- BaseFileName
- BeginBatch
- Close
- Create
- Delete
- EndBatch
- HasSnapshot
- Open
- Read
- ReadCustomMetadata

- SectorSize
- SnapshotFileName
- TakeSnapshot
- TotalSectorCapacity
- TranslateSectorOffsetToChsTuple
- Write
- WriteCustomMetadata

BaseFileName

Gets VHD base file name.

URI: vhd/{id}/baseFileName

HTTP Method: GET

BeginBatch

Begins batch.

URI: vhd/{id}/beginBatch/{target}

HTTP Method: POST

Close

Closes VHD.

URI: vhd/{id}/close

HTTP Method: GET

Create

Creates VHD.

URI:
vhd/createVhd?path={path}&bytesCapacity={bytesCapacity}&bytesPerSector={bytesPerSector}&containsBootSystemVolume={containsBootSystemVolume}&preallocate={preallocate}

HTTP Method: PUT

Delete

Deletes VHD snapshot or base file.

URI: vhd/{id}/delete/{target}

HTTP Method: POST

EndBatch

Ends batch.

URI: vhd/{id}/endBatch

HTTP Method: POST

HasSnapshot

Verifies VHD has snapshot.
URI: vhd/{id}/hasSnapshot
HTTP Method: GET

Open

Opens VHD.
URI: vhd/openVhd?path={path}
HTTP Method: PUT

Read

Reads raw data from VHD.
URI: vhd/{id}/read/{target}/{sectorOffset}/{sectorLength}
HTTP Method: POST

ReadCustomMetadata

Reads a user-defined custom metadata string.
URI: vhd/{id}/readCustomMetadata/{target}/{key}
HTTP Method: POST

SectorSize

Gets sector size of the VHD.
URI: vhd/{id}/sectorSize
HTTP Method: GET

SnapshotFileName

Gets VHD snapshot file name.
URI: vhd/{id}/snapshotFileName
HTTP Method: GET

TakeSnapshot

Takes VHD snapshot.
URI: vhd/{id}/takeSnapshot
HTTP Method: GET

TotalSectorCapacity

Gets VHD capacity.
URI: vhd/{id}/totalSectorCapacity
HTTP Method: GET

TranslateSectorOffsetToChsTuple

Translates sector offset to chs tuple.
URI: vhd/{id}/translateSectorOffsetToChsTuple?sectorOffset={sectorOffset}
HTTP Method: GET

Write

Writes raw data to VHD.
URI: vhd/{id}/write/{target}/{sectorOffset}/{sectorLength}
HTTP Method: POST

WriteCustomMetadata

Reads a user-defined custom metadata string.
vhd/{id}/writeCustomMetadata/{target}/{key}?value={value}
HTTP Method: POST

IWhiteLabelingManagement

This section describes the service operations available for IWhiteLabelingManagement at whitelabeling/.

The IWhiteLabelingManagement service contract exposes the customizable strings.

The URI and HTTP method are provided for each service operation is as follows:

- GetWhiteLabelingInfo

GetWhiteLabelingInfo

Gets customizable strings.
URI: whitelabeling/whiteLabelingInfo
HTTP Method: GET

Glossary

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

A

Agent

The AppAssure Agent is software installed on a physical or virtual machine that lets it be added to protection in the AppAssure Core.

AppAssure

AppAssure sets a new standard for unified data protection by combining backup, replication, and recovery in a single solution that is engineered to be the fastest and most reliable backup for protecting virtual machines (VM), as well as physical and cloud environments.

B

base image

The first backup transfer saved to the Core is called a base image snapshot. All data on all specified volumes (including the operating system, applications, and settings), are saved to the Core. For more information, see snapshot.

C

Central Management Console

The AppAssure Central Management Console is a multi-core management portal. It simplifies the process of managing multiple deployments of the AppAssure Core. Using the Central Management Console, you can group and manage the deployments through a single, Web-based interface.

checksum

A checksum is a function that creates blocks of data that are used for the purpose of detecting accidental errors that are created during transmission or storage.

cluster

See Windows failover cluster.

cluster continuous replication (CCR)

A non-shared storage failover cluster solution, that uses built-in asynchronous log shipping technology to create and maintain a copy of each storage group on a second server in a failover cluster. CCR is designed to be either a one or two data center solution, providing both high availability and site resilience. It is one of two types of clustered mailbox server (CMS) deployments available in Exchange 2007.

cluster node

An individual machine that is part of a Windows Failover cluster.

compression

The Storage Networking Industry Association (SNIA) defines compression as the process of encoding data to reduce its size.

Core

The AppAssure Core is the central component of the AppAssure architecture. The Core provides the essential services for backup, recovery, retention, replication, archiving, and management. In the context of replication, the Core is also called a *source core*. The source core is the originating core, while the target core is the destination (another AppAssure Core on its own dedicated server, where protected machines or clusters are replicated).

D

database availability group (DAG)

A set of up to 16 Microsoft Exchange Server 2010 Mailbox servers that provide automatic, database-level recovery from a database, server, or network failure. DAGs use continuous replication and a subset of Windows failover clustering technologies to provide high availability and site resilience. Mailbox servers in a DAG monitor each other for failures. When a Mailbox server is added to a DAG, it works with the other servers in the DAG to provide automatic, database-level recovery from database failures.

E

encryption

Data is encrypted with the intent that it is only accessible to authorized users who have the appropriate decryption key. Data is encrypted using 256-bit AES in Cipher Block Chaining (CBC) mode. In CBC, each block of data is XORed with the previous ciphertext block before being encrypted, this way each new ciphertext block depends on all preceding plaintext blocks. A passphrase is used as an initialization vector.

G

global deduplication

The Storage Networking Industry Association (SNIA) defines data deduplication as the replacement of multiple copies of data—at variable levels of granularity—with references to a shared copy to save storage space or bandwidth. The AppAssure Volume Manager performs global data deduplication within a logical volume. The granularity level of deduplication is 8 KB. The scope of deduplication in AppAssure is limited to protected machines using the same repository and encryption key.

I

incremental snapshot

Incremental snapshots are backups consisting only of data changed on the protected machine since the last backup. They are saved to the Core regularly, based on the interval defined (for example, every 60 minutes). For more information, see snapshot.

L

license key

A license key is one method used to register your AppAssure software or appliance. (You can also use a license file.) You can obtain license keys or files when you register on the Dell AppAssure License Portal for an account. For more information, see License Portal.

License Portal

The Dell AppAssure License Portal is a Web interface where users and partners can download software, register AppAssure appliances, and manage license subscriptions. License Portal users can register accounts, download AppAssure Core and Agent software, manage groups, track group activity, register machines, register appliances, invite users, and generate reports. For more information, see the *Dell AppAssure License Portal User Guide*.

Live Recovery

AppAssure Live Recovery is an instant recovery technology for VMs and servers. It provides near-continuous access to data volumes in a virtual or physical server, letting you recover an entire volume with near-zero RTO and a RPO of minutes.

Local Console

The Local Console is a Web-based interface that lets you fully manage the AppAssure Core.

Local Mount Utility

The Local Mount Utility (LMU) is a downloadable application that lets you mount a recovery point on a remote AppAssure Core from any machine.

log truncation

Log truncation is a function that removes log records from the transaction log. For a SQL Server machine, when you force truncation of the SQL Server logs, this process identifies free space on the SQL server. For an Exchange Server machine, hen you force truncation of the Exchange Server logs, this action frees up space on the Exchange server.

M

management roles

The AppAssure Central Management Console introduces a new concept of management roles which lets you divide administrative responsibility among trusted data and service administrators as well as access control to support secure and efficient delegation of administration.

mountability

Exchange mountability is a corruption detection feature that alerts administrators of potential failures and ensures that all data on the Exchange servers is recovered successfully in the event of a failure.

O

Object File System

The AppAssure Scalable Object Store is an object file system component. It treats all data blocks, from which snapshots are derived, as objects. It stores, retrieves, maintains, and replicates these objects. It is designed to deliver scalable input and output (I/O) performance in tandem with global data deduplication, encryption, and retention management. The Object File System interfaces directly with industry standard storage technologies.

P

passphrase

A passphrase is a key used in the encryption the data. If the passphrase is lost, data cannot be recovered.

PowerShell scripting

Windows PowerShell is a Microsoft .NET Framework-connected environment designed for administrative automation. AppAssure includes comprehensive client SDKs for PowerShell scripting that enables administrators to automate the administration and management of AppAssure resources by the execution of commands either directly or through scripts.

prohibited characters

Prohibited characters are characters that should not be used when naming an object in the AppAssure Core Console. For example, when defining a display name for a protected machine, do not use any of the following special characters:

Table 3. Prohibited characters

Character	Character name	Prohibited from
?	question mark	machine display name, encryption key, repository, path description
|	pipe	machine display name, encryption key, repository, path description
:	colon	machine display name, encryption key, repository Use of this symbol is supported when specifying a path, for example c:\data.
/	forward slash	machine display name, encryption key, repository, path description

Table 3. Prohibited characters

Character	Character name	Prohibited from
\	back slash	machine display name, encryption key, repository Use of this symbol is supported when specifying a local or network path, for example c:\data or \\ComputerName\SharedFolder\
*	asterisk	machine display name, encryption key, repository, path description
"	quotation mark	machine display name, encryption key, repository, path description
<	open angle bracket	machine display name, encryption key, repository, path description
>	close angle bracket	machine display name, encryption key, repository, path description

prohibited phrases

Prohibited phrases are phrases (or sets of characters) that should not be used as the name for any object in the AppAssure Core Console, because they are reserved for the use of operating systems. It is best practice is to avoid using these phrases at all if possible. For example, when defining a display name for a protected machine, do not use any of the following phrases:

Table 4. Prohibited phrases

Phrase	General use	Prohibited from
con	console	machine display name, encryption key, repository, path description
prn	printer port	machine display name, encryption key,
aux	auxiliary port	machine display name, encryption key,
nul	null value	machine display name, encryption key,
com1 through com9	communication port	machine display name, encryption key,
lpt1 through lpt9	line print terminal port	machine display name, encryption key, repository, path description

protected machine

A protected machine, sometimes referred to as an agent, is a physical computer or virtual machine that is protected in the AppAssure Core. The machine must first have the AppAssure agent software installed.

Q

quorum

For a failover cluster, the number of elements that must be online for a given cluster to continue running. The elements relevant in this context are cluster nodes. This term can also refer to the quorum-capable resource selected to maintain the configuration data necessary to recover the cluster. This data contains details of all of the changes that have been applied to the cluster database. The quorum resource is generally accessible to other cluster resources so that any cluster node has access to the most recent database changes. By default there is only one quorum resource per server cluster. A particular quorum configuration (settings for a failover cluster) determines the point at which too many failures stop the cluster from running.

R

recovery points

Recovery points are a collection of snapshots of various disk volumes. For example, C:, D:, and E.

remote Core

A remote Core represents an AppAssure Core that is accessed by a non-core machine by way of the Local Mount Utility.

replication

Replication is self-optimizing with a unique read-match-write (RMW) algorithm that is tightly coupled with deduplication. It represents the relationship between the target and source cores in the same site or across two sites with slow link in which the source core asynchronously transmits the data to the target or source core on a per agent basis.

replicated machine

A replicated machine (also known as a target core) is a copy of a protected physical computer or virtual machine (known as a source core).

repository

A repository, which is managed by the AppAssure Core, is a folder used to store snapshots that are captured from the protected servers and machines. The repository can reside on different storage technologies such as Storage Area Network (SAN), Direct Attached Storage (DAS), or Network Attached Storage (NAS).

restore

The process of restoring one or more storage volumes on a machine from recovery points saved on the AppAssure Core is known as performing a restore. This was formerly known as rollback.

retention

Retention defines the length of time the backup snapshots of protected machines are stored on the AppAssure Core. Retention policy is enforced on the recovery points through the rollup process.

rollup

The rollup process is an internal nightly maintenance procedure that enforces the retention policy by collapsing and eliminating dated recovery points. AppAssure reduces rollup to metadata operations only.

S

seeding

In replication, the initial transfer of deduplicated base images and incremental snapshots of protected agents, which can add up to hundreds or thousands of gigabytes of data. Initial replication can be seeded to the target core using external media, which is useful for large sets of data or sites with slow links.

server cluster

See Windows failover cluster.

SharePoint backup

A SharePoint backup is a copy of data that is used to restore and recover that data on a SharePoint server after a system failure. From the SharePoint backup, you can perform recovery of the complete SharePoint farm, or one or more components of the farm.

single copy cluster

A shared storage failover cluster solution, that uses a single copy of a storage group on storage that is shared between the nodes in the cluster. It is one of two types of clustered mailbox server deployments available in Exchange 2007.

Smart Agent

The AppAssure Smart Agent is installed on the machines protected by the AppAssure Core. The smart agent tracks the changed blocks on the disk volume and snapshots the changed blocks at a predefined interval of protection.

snapshot

A snapshot is a common industry term that defines the ability to capture and store the state of a disk volume at a given point, while applications are running. The snapshot is critical if system recovery is needed due to an outage or system failure. AppAssure snapshots are application aware, which means that all open transactions and rolling transaction logs are completed and caches are flushed prior to creating the snapshot. AppAssure uses Microsoft Volume Shadow Services (VSS) to facilitate application crash consistent snapshots.

SQL attachability

SQL attachability is a test run within the AppAssure Core to ensure that all SQL recovery points are without error and are available for backup in the event of a failure.

SQL backup

A SQL backup is a copy of data that is used to restore and recover that data on a SQL server after a system failure. From the SQL backup, you can perform recovery of the complete SQL database, or one or more of the components of the SQL database.

SQL differential backup

A differential database backup is a cumulative copy of all changes in data since the last full backup of the SQL database. Differential backups are typically faster to create than full database backups, and reduce the number of transaction logs required to recover the database.

T

target Core

The target core, which is sometimes referred to as *replica core*, is the AppAssure Core receiving the replicated data from the source core.

target replica machine

The instance of a protected machine on a target core is known as the target agent or replica agent.

Transport Layer Security

Transport Layer Security (TLS) is a modern cryptographic network protocol designed to ensure communication security over the Internet. This protocol, defined by the Internet Engineering Task Force, is the successor to Secure Sockets Layer (SSL). The SSL term is still generally used, and the protocols are interoperable (a TLS client can downgrade to communicate to an SSL server).

True Scale

True Scale is the scalable architecture of AppAssure.

U

Universal Recovery

AppAssure Universal Recovery technology provides unlimited machine restoration flexibility. It enables you to perform monolithic recovery to- and from- any physical or virtual platform of your choice as well as incremental recovery updates to virtual machines from any physical or virtual source. It also lets you perform application-level, item-level, and object-level recovery of individual files, folders, email, calendar items, databases, and applications.

V

Verified Recovery

Verified Recovery technology is used to perform automated recovery testing and verification of backups. It supports various file systems and servers.

virtual standby

Virtual Standby is a physical-to-virtual (P2V) process that creates a clone virtual machine of a protected machine or agent. A Virtual Standby can be created using an *ad-hoc* or a *continual update* export process. A Virtual Standby created using a *continual update* is incrementally updated after every snapshot captured from the source agent.

Volume Manager

The AppAssure Volume Manager manages objects and then stores and presents them as a logical volume. It leverages dynamic pipeline architecture to deliver TruScale scalability, parallelism, and asynchronous input-and-output (I/O) model for high throughput with minimal I/O latency.

W

white labeling

AppAssure provides the ability for providers of backup and disaster recovery services to white label or re-brand AppAssure with their own identity; and then sell or distribute it as their own product or service.

Windows failover cluster

A group of independent computers that work together to increase the availability of applications and services. The clustered servers (called nodes) are connected by physical cables and by software. If one of the cluster nodes fails, another node begins to provide service (a process known as failover). Users experience a minimum of disruptions in service. AppAssure supports the protection of a number of SQL Server and Exchange Server cluster types.

Return to top

About Dell

Dell listens to customers and delivers worldwide innovative technology, business solutions and services they trust and value. For more information, visit www.software.dell.com.

Contacting Dell

Technical support:
Online support

Product questions and sales:
(800) 306-9329

Email:
info@software.dell.com

Technical support resources

Technical support is available to customers who have purchased Dell software with a valid maintenance contract and to customers who have trial versions. To access the Support Portal, go to http://software.dell.com/support/.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. In addition, the portal provides direct access to product support engineers through an online Service Request system.

The site enables you to:

- Create, update, and manage Service Requests (cases)
- View Knowledge Base articles
- Obtain product notifications
- Download software. For trial software, go to Trial Downloads.
- View how-to videos
- Engage in community discussions
- Chat with a support engineer