

vRealize Hyperic Installation and Configuration Guide

vRealize Hyperic 5.8.4



vmware®

You can find the most up-to-date technical documentation on the VMware website at:

<https://docs.vmware.com/>

If you have comments about this documentation, submit your feedback to

docfeedback@vmware.com

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Copyright © 2015 VMware, Inc. All rights reserved. [Copyright and trademark information.](#)

Contents

vRealize Hyperic Installation and Configuration Guide	4
1 vRealize Hyperic Supported Configurations and System Requirements	5
2 Installing and Configuring vRealize Hyperic	9
Selecting a vRealize Hyperic Installation Package	9
Download vRealize Hyperic	10
Configuring JRE Locations for vRealize Hyperic Components	11
Set up the vRealize Hyperic Database	12
Identify PostgreSQL Connection Issues	16
Installing and Configuring the vRealize Hyperic Server	17
Using the vRealize Hyperic vApp Management Console	29
Install and Configure the vRealize Hyperic Agent	32
Configuring SSL Options	41
Install Multiple vRealize Hyperic Agents Simultaneously	44
Activating and Configuring Your vRealize Hyperic License	47
3 Migrating and Upgrading vRealize Hyperic Components	50
Upgrading vRealize Hyperic Components and Migrating the Database	50
Upgrading the vRealize Hyperic Agent	54
4 Deploy a vRealize Hyperic vApp Using vCloud Director	57
Create and Configure the vRealize Hyperic vApp	57
Power on the vRealize Hyperic Server	59
5 Post Installation Administration	60
Encrypt vRealize Hyperic Agent Property Values	60
Uninstall a vRealize Hyperic Agent	61

vRealize Hyperic Installation and Configuration Guide

The vRealize Hyperic Installation and Configuration Guide provides procedures for installing vRealize Hyperic components, including setting up the vRealize Hyperic database and installing the vRealize Hyperic server and vRealize Hyperic agents.

Intended Audience

This information is intended for operations personnel who set up and support the vRealize Hyperic infrastructure.

vRealize Hyperic Supported Configurations and System Requirements

1

These tables describe the supported operating systems for vRealize Hyperic server and vRealize Hyperic agent installations.

It is good practice that host operating systems employ a NTP time synchronization between the vRealize Hyperic server and vRealize Hyperic agent. If you do not use NTP, metrics are displayed according to the server time. To correlate data on the server with an agent that is not synchronized, you must calculate the time difference.

The supported operating system tables indicate whether the supported configuration is for production or development. Production support means you can run your production application on the platform.

The developer support designation indicates those products that are "known to work" and for which VMware will provide best-effort support to resolving reported issues. Developer Support certifications are not supported for use in production.

Supported Operating Systems for the vRealize Hyperic Server

The following table provides data about the operating systems that are supported for use with vRealize Hyperic.

Table 1-1. Supported Operating Systems for the vRealize Hyperic Server

Operating System	Processor Architecture	JVM	Production/Developer Support	Scaling Considerations
Pre-deployed virtual appliance (vAPP)	x86_64	Oracle Java SE 7	Production	Best configuration for environments with greater than 1000 managed platforms.
RedHat Enterprise Linux (RHEL) 5, or CentOS 5.x	x86_64	Oracle Java SE 7	Production	
RedHat Enterprise Linux (RHEL) 6	x86_64	Oracle Java SE 7	Production	For medium scale environments

Table 1-1. Supported Operating Systems for the vRealize Hyperic Server (Continued)

Operating System	Processor Architecture	JVM	Production/Developer Support	Scaling Considerations
SuSE Enterprise Linux (SLES) 11	x86_64	Oracle Java SE 7	Production	For medium scale environments
Microsoft Windows Server 2008	x86_64	Oracle Java SE 7	Production	For medium scale environments

On Unix-like platforms, the vRealize Hyperic server requires the `libXp.so.6` X library to create charts and other graphics in the vRealize Hyperic user interface. The location of this library varies according to version and provider.

Provider	Comment
Enterprise Linux	From RedHat Enterprise Linux 4 and CentOS 4, <code>libXp.so.6</code> is in the <code>xorg-x11-deprecated-libs</code> RPM.
Debian	Install the <code>libxp6</code> , <code>libxt6</code> , <code>libxtst6</code> , and <code>libx11-6</code> packages.
Fedora Core 5	From Fedora Core 5, the <code>libXp.so</code> library has been separated to its own package. Install the <code>libXp</code> RPM.
Other distributions	The required libraries can be found in either the <code>XFree86-libs</code> or the <code>xorg-x11-libs</code> package.

Supported Operating Systems for the vRealize Hyperic Agent

These configurations are supported for the agent in both development and production environments.

Table 1-2. Supported Operating Systems for the vRealize Hyperic Agent

Operating System*	Processor Architecture	JVM	Scaling Considerations
RedHat Enterprise Linux (RHEL) 5.x, 6.x, 7.x	x86_64, x86_32 (not tested for v.7.x)	Oracle Java SE7	RHEL 7 requires vRealize Hyperic agent 5.8.3, or higher
CentOS 5.x, 6.x, 7.x	x86_64, x86_32	Oracle Java SE7	
SUSE Enterprise Linux (SLES) 11.x, 12.x	x86_64	Oracle Java SE7	SLES 12 requires vRealize Hyperic agent 5.8.3, or higher
Windows 2008 Server, 2008 Server R2	x86_64, x64_32	Oracle Java SE7	
Windows 2012 Server, 2012 Server R2	x86_64, x64_32	Oracle Java SE7	
Solaris 10, 11	x86_64, x86_32, SPARC	Oracle Java SE7	x86 support requires vRealize Hyperic agent 5.8.1, or higher
HP-UX 11.11 or higher	PA-Risc	Oracle Java SE7	
AIX 6.1, 7.1	Power PC	IBM Java SE7	

Host Machine Requirements

The following table lists the host system requirements for the vRealize Hyperic server and the vRealize Hyperic database, and assumes that the database runs on a different host than the vRealize Hyperic server.

- vRealize Hyperic supports only one vRealize Hyperic server on a host. The host must have a static IP address for server communications.
- vRealize Hyperic supports only one vRealize Hyperic agent on a host.
- To manage more than 100 platforms, you must run the vRealize Hyperic database on a dedicated host, not that on which the vRealize Hyperic server runs.

In the Host Machine Requirements table,

- "Medium" is 50 to 250 managed platforms, or 5000-30000 managed resources.
- "Large" is 250 to 2,000 managed platforms, or greater than 30,000 managed resources.

Table 1-3. Minimum Host Machine Requirements

Resource	vRealize Hyperic Server	Required vRealize Hyperic Database
Processor	Medium: 4 or more server class CPUs, 2GHz Large: 6 or more server class CPUs, 2GHz	Medium: 4 or more server class CPUs, 2GHz Large: 6 or more server class CPUs, 2GHz
Memory	Medium: 8GB Large: 12GB	Medium: 6GB Large: 16GB
Free Disk Space	20GB	Medium: 200GB Large: 500GB

Databases Supported for vRealize Hyperic Server

The vRealize Hyperic database supports PostgreSQL or vPostgreSQL 9.1.x.

The vRealize Hyperic Server vApp includes a virtual machine with a vPostgreSQL database. The database that is installed when you run the vRealize Hyperic installer in default mode is PostgreSQL.

Supported Browsers

Hyperic supports the following browsers. Firefox is recommended.

The Skype plug-in for Firefox causes unexpected behavior in the Hyperic user interface. Disable the plug-in to work around this problem.

Table 1-4. Supported Browsers

Browser	Version
Firefox	11, 12
Internet Explorer	7, 8, 9

Table 1-4. Supported Browsers (Continued)

Browser	Version
Safari	5.0, 5.1
Chrome	21

Agent Server Compatibility Requirements

The vRealize Hyperic agents that report to the vRealize Hyperic server must be the same version as the server, or an earlier version than the server.

Installing and Configuring vRealize Hyperic

2

This section includes the following topics:

- [Selecting a vRealize Hyperic Installation Package](#)
- [Download vRealize Hyperic](#)
- [Configuring JRE Locations for vRealize Hyperic Components](#)
- [Set up the vRealize Hyperic Database](#)
- [Identify PostgreSQL Connection Issues](#)
- [Installing and Configuring the vRealize Hyperic Server](#)
- [Using the vRealize Hyperic vApp Management Console](#)
- [Install and Configure the vRealize Hyperic Agent](#)
- [Configuring SSL Options](#)
- [Install Multiple vRealize Hyperic Agents Simultaneously](#)
- [Activating and Configuring Your vRealize Hyperic License](#)

Selecting a vRealize Hyperic Installation Package

You can download vRealize Hyperic in a variety of packages. The format that you select depends on the operating system on which it will be installed, whether configuration will be automated or customized, and so on.

vRealize Hyperic installers can be downloaded from the VMware download page at <http://downloads.vmware.com/>. On the download page, under Application Management select **VMware vRealize Hyperic**.

The installation packages are described below. JREs are included in some packages and not others. To determine if you need to configure the location of your JRE, see [Configuring JRE Locations for vRealize Hyperic Components](#).

vRealize Hyperic vApp

A virtual appliance (vApp) is one or more virtual machine image files (.ovf), each with a preconfigured operating system environment and application. The vRealize Hyperic vApp contains two virtual machine images, one for the vRealize Hyperic server and one for the vRealize Hyperic database. Deploying the vRealize Hyperic vApp provides a simplified deployment in which the components are already configured to work, and to work with each other.

The vRealize Hyperic vApp is provided as an OVA archive that contains the .ovf descriptor, .mf, and .vmdk files that are necessary to deploy the vRealize Hyperic server and vRealize Hyperic database vApps using vSphere Client. You can also create a vRealize Hyperic vApp in your virtual cloud from a vApp template, using VMware vCloud Director.

For installation prerequisites and instructions, see [Install vRealize Hyperic vApp](#).

vRealize Hyperic Installer

The vRealize Hyperic installer is script-based. You can do a quick install that sets up defaults for most vRealize Hyperic server configuration options, or run it in full mode to respond to the configuration dialog yourself. You can also use this installer to install the vRealize Hyperic agent.

RHEL RPMs

RPMs are available. The vRealize Hyperic server RPM is the standard vRealize Hyperic installer, wrapped in an Expect script.

Download vRealize Hyperic

You can download vRealize Hyperic in a variety of packages. The format that you select depends on the operating system on which it will be installed, whether configuration will be automated or customized, and so on.

vRealize Hyperic installers can be downloaded from the VMware download page at <http://downloads.vmware.com/>. On the download page, under Application Management select **VMware vRealize Hyperic**.

Prerequisites

Review the available download packages to select the one that is most appropriate for your requirements. See [Selecting a vRealize Hyperic Installation Package](#).

Verify that you have the necessary system requirements to install the package that you download. See [Chapter 1 vRealize Hyperic Supported Configurations and System Requirements](#)

Procedure

- 1 On the Download VMware vRealize Hyperic 5.8 page, select the product to download and click **View Download**.
- 2 Select the installer package file to download and click **Download Manager**.
- 3 Log in to My VMware.
- 4 Accept the license agreement.
- 5 Begin the download process.
 - a (Optional) Click the **Download Manager** link to open the Download Manger utility.
The Download Manger allows you to pause and resume downloads of large file sets, as long as the utility remains running.
 - b (Optional) Click **Download Now** and run or save the installer file.

Configuring JRE Locations for vRealize Hyperic Components

Both the vRealize Hyperic server and vRealize Hyperic agents require a JRE. The platform-specific vRealize Hyperic installers include a JRE. Platform-independent vRealize Hyperic installers do not include a JRE.

Depending on your environment and the installation package you use, you might need to define the location of the JRE for the server or your agents. The following table shows when you need to configure the locations of the JRE for your server or agents.

Table 2-1. vRealize Hyperic Installations that Require JRE Location Configuration

Installation Package	JRE Location Configuration Required
Platform-specific server or agent installation on a machine that has its own JRE that you want to use	Location configuration required
Platform-independent server or agent installation	Location configuration required
Agent installation from an RPM	Location configuration required

How the vRealize Hyperic Agent Resolves its JRE

The vRealize Hyperic agent resolves its JRE based on platform type.

On UNIX-like platforms, the vRealize Hyperic agent resolves the JRE to use in the following order:

- 1 HQ_JAVA_HOME environment variable
- 2 Embedded JRE
- 3 JAVA_HOME environment variable

On Windows platforms, the vRealize Hyperic agent installer is available with or without a JRE.

You must configure the JRE location with the `HQ_JAVA_HOME` system variable if you do a platform-independent agent install on Windows, or a platform-specific install on a Windows machine that already has a JRE that you prefer to use.

On Windows platforms, the vRealize Hyperic agent resolves the JRE to use in the following order:

- 1 `HQ_JAVA_HOME` environment variable
- 2 Embedded JRE

You define a system variable from **My Computer > Properties > Advanced > Environment Variables > System Variables > New**.

Set up the vRealize Hyperic Database

Setting the vRealize Hyperic database requires you to complete a number of processes in a specific order, as described here.

You are not required to set up the database if you are deploying the vRealize Hyperic vApp or if you are going to use the embedded PostgreSQL database.

In large environments, set up the vRealize Hyperic database on a dedicated platform.

Prerequisites

Verify that you have the necessary host machine requirements for setting up the database. See [Chapter 1 vRealize Hyperic Supported Configurations and System Requirements](#).

Procedure

1 [Install PostgreSQL](#)

You install PostgreSQL on RHEL 5.

2 [Define Data and Log Locations](#)

Depending on your environment, you might want to select a location other than the default for data files. For example, you might want to store data files on a volume with plenty of space for housekeeping operations.

3 [Define PostgreSQL Environment Variables](#)

As postgres user, you can update your bash configuration file.

4 [Configure PostgreSQL Properties](#)

You must configure specific properties in the `postgresql.conf` file.

5 [Configure PostgreSQL Client Authentication](#)

PostgreSQL client authentication is defined in the `pg_hba.conf` file, which contains records that specify allowed connection types, users, client IP addresses, and the authentication method.

6 [Create a vRealize Hyperic Database User and Database](#)

You must create an account that the vRealize Hyperic Server will use to connect to the vRealize Hyperic database.

Install PostgreSQL

You install PostgreSQL on RHEL 5.

You must be logged in as root to perform the installation.

If database initialization fails, look for error messages in `/var/lib/pgsql/9.1/pgstartup.log`.

Procedure

- 1 Run `wget http://yum.postgresql.org/9.1/redhat/rhel-5-x86_64/pgdg-redhat91-9.1-5.noarch.rpm` to download a 9.1 PostgreSQL RPM from the PostgreSQL yum repository.
- 2 Run `yum install pgdg-redhat91-9.1-5.noarch.rpm` to install the RPM.
- 3 Run `yum install postgresql91-server postgresql91-contrib` to install the PostgreSQL server and contrib modules.
- 4 Run `service postgresql-9.1 initdb` to initialize PostgreSQL.

The `initdb` command creates the directories to contain database information, generates shared catalog tables, and creates the `template1` and `postgres` databases. New databases that you create are based on the `template1` database. The `postgres` database is a default database for use by all users, utilities and third party applications.

What to do next

Define data and log locations. See [Define Data and Log Locations](#).

Define Data and Log Locations

Depending on your environment, you might want to select a location other than the default for data files. For example, you might want to store data files on a volume with plenty of space for housekeeping operations.

In this procedure, you replace `PathToPreferredDisk` with the path to a disk location that has the optimal space and throughput.

Procedure

- 1 Run `/usr/pgsql-9.1/bin/pg_ctl -D /var/lib/pgsql/9.1/data -l ~/logs/logfile stop -m fast` to stop PostgreSQL.
- 2 Run `Export $PGDATA PathToPreferredDisk/data` to set the `$PGDATA` environment variable to point to the required location.
- 3 Run `mkdir -p PathToPreferredDisk` to create a directory on the required volume.
- 4 Run `mv /var/lib/pgsql/9.1/data PathToPreferredDisk` to move the data files to the new location.

- 5 Run `/usr/pgsql-9.1/bin/pg_ctl -D $PGDATA -l $PGDATA/pg_log/logfile start` to restart PostgreSQL.

What to do next

Define the PostgreSQL environment variables. See [Define PostgreSQL Environment Variables](#)

Define Data and Log Locations when the I/O Device is Saturated

Depending on your environment, you might select a location other than the default for data files. For example, you might choose to store data files on a volume with plenty of space for housekeeping operations. In the event that your I/O device is saturated, the procedure that you use differs from the usual process.

If you do not have an I/O device that is saturated, use the process described in [Define Data and Log Locations](#).

Procedure

- 1 Run `/usr/pgsql-9.1/bin/pg_ctl stop -m fast` to stop PostgreSQL.
- 2 Run `mv PathToPreferredDisk/data/pg_xlog /var/tmp/` to move the logs.
- 3 Run `ln -s /var/tmp/pg_xlog PathToPreferredDisk/data/pg_xlog` to create a symbolic link to the new location of the `pg_xlog` .
- 4 Run `/usr/pgsql-9.1/bin/pg_ctl -D $PGDATA -l $PGDATA/pg_log/logfile start` to restart PostgreSQL.

What to do next

Define the PostgreSQL environment variables. See [Define PostgreSQL Environment Variables](#)

Define PostgreSQL Environment Variables

As postgres user, you can update your bash configuration file.

Procedure

- ◆ Run the command `export PGDATA="/data/pgdata" export PGHOME="PostresqlHome" export PATH="$PGHOME/bin/:$PATH"` where `PostgresqlHome` is the path to the PostgreSQL installation.

What to do next

Configure the PostgreSQL properties. See [Configure PostgreSQL Properties](#).

Configure PostgreSQL Properties

You must configure specific properties in the `postgresql.conf` file.

Procedure

- 1 Under the `listen_addresses` property, enable database connections on all interfaces on the platform. `listen_addresses = '*'`

- Under the `max_connections` property, set the maximum number of connections based on the sizing profile that corresponds to the scale of your environment. Use the following values.

Value	Description
?	Small
?	Medium
500	Large

- Under the `shared_buffers` and `effective_cache_size` property, assuming the database runs on a dedicated platform, set `shared_buffers` to 70-80% of memory, and `effective_cache_size` to 10-20%, leaving some memory available for the operating system.

For example, given 12GB of memory, `shared_buffers` = 8GB and `effective_cache_size` = 2GB.

- Under the `checkpoint_segments` property, verify that the value on the sizing profile corresponds to the scale of your environment. Use the following values.

Value	Description
<code>checkpoint_segments = 3 (default)</code>	Small
<code>checkpoint_segments = 3 (default)</code>	Medium
<code>checkpoint_segments = 32 (default)</code>	Large

What to do next

Configure the PostgreSQL client authentication properties. See [Configure PostgreSQL Client Authentication](#).

Configure PostgreSQL Client Authentication

PostgreSQL client authentication is defined in the `pg_hba.conf` file, which contains records that specify allowed connection types, users, client IP addresses, and the authentication method.

For more information about `pg_hba.conf`, see <http://www.postgresql.org/docs/9.1/static/auth-pg-hba-conf.html>.

Procedure

- In the `pg_hba.conf` file, locate the line `# TYPE DATABASE USER ADDRESS METHOD`.
- Immediately below the located line, add `host all all 0.0.0.0/0 password`, aligning these values under the parameter names in the row above.

```
# TYPE DATABASE USER ADDRESS METHOD
host all all 0.0.0.0/0 password
```

You can use the address parameter to limit access to just the vRealize Hyperic server or any clients that need to query the database.

What to do next

Create a vRealize Hyperic database user and database. See [Create a vRealize Hyperic Database User and Database](#).

Create a vRealize Hyperic Database User and Database

You must create an account that the vRealize Hyperic Server will use to connect to the vRealize Hyperic database.

Procedure

- 1 Change the user to **postgres** and run `# psql` to connect to the database locally.
 - 2 Run `CREATE USER hqadmin WITH ENCRYPTED PASSWORD 'hqadmin';` to create a user named hqadmin with login and createdb privileges.
- The ENCRYPTED keyword is optional.
- 3 Run `CREATE DATABASE "HQ" OWNER hqadmin ENCODING 'UTF8';` to create a default database for vRealize Hyperic.

Placing quote marks around the HQ string makes the database name uppercase.

What to do next

Install the vRealize Hyperic server. See [Installing and Configuring the vRealize Hyperic Server](#).

Identify PostgreSQL Connection Issues

If the vRealize Hyperic server fails to start, it might be due to problems with the PostgreSQL configuration.

Check the PostgreSQL logs for connection failures or errors.

Procedure

- (Optional) Troubleshoot connection issues using the command `telnet <dbserver hostname> 5432` from the vRealize Hyperic server host.
- (Optional) If network connections to the database fail, troubleshoot the issue in PostgreSQL log files using the UNIX [®] `tail` command with the `-f` parameter.

`tail -f` displays the lines at the end of a file, and additional log messages that follow to the terminal. This is useful for watching log files, or any other file that might be appended over time. Failed connection messages are written to the following files:

- `/var/lib/pgsql/data/pg_log/postgresql-day.log`
- `/var/lib/pgsql/pgstartup.log`
- (Optional) If the vRealize Hyperic server fails to connect to the PostgreSQL database, determine if there is a firewall issue by turning off the firewall on RHEL or CentOS and running the `/etc/init.d/iptables stop` command as root.

Installing and Configuring the vRealize Hyperic Server

You can install and configure the vRealize Hyperic Server in three different ways. You can use the vRealize Hyperic vApp, you can run the vRealize Hyperic installation script, or you can use a Windows setup wizard.

- [Install vRealize Hyperic vApp](#)

A virtual appliance (vApp) is one or more virtual machine image files (OVF), each with a preconfigured operating system environment and application. The vRealize Hyperic vApp contains two virtual machine images, one for the vRealize Hyperic server and one for the vRealize Hyperic database.

- [Deploy vRealize Hyperic vApp in a Medium or Large Scale Environment](#)

Use this procedure when you are deploying vRealize Hyperic vApp in a medium or large scale environment.

- [Run the vRealize Hyperic Installer Setup Script](#)

You can make a new installation of the vRealize Hyperic server by running an installation script.

- [Install vRealize Hyperic Server Using Windows Setup Wizard](#)

You can use a Windows setup wizard to install the vRealize Hyperic server.

- [Using RPM to Install vRealize Hyperic Server](#)

You can use a RedHat Package Manager package to install the vRealize Hyperic server.

- [Configure the vRealize Hyperic Server Properties File After Installation](#)

If you did not configure the vRealize Hyperic server properties before running the RPM server installer, you are prompted to do so after installation. You cannot use the server until the properties have been specified.

Install vRealize Hyperic vApp

A virtual appliance (vApp) is one or more virtual machine image files (OVF), each with a preconfigured operating system environment and application. The vRealize Hyperic vApp contains two virtual machine images, one for the vRealize Hyperic server and one for the vRealize Hyperic database.

To deploy vRealize Hyperic vApp in a large scale environment, see [Deploy vRealize Hyperic vApp in a Medium or Large Scale Environment](#).

Prerequisites

- vCenter Server must be installed and running.
- vSphere Client must be installed.
- To assign fixed IP addresses to the vRealize Hyperic server and vRealize Hyperic database, which is the recommended best practice, have the list of IP addresses available when running the deployment wizard.

- If you are using an external PostgreSQL database, you must set it up before installing vRealize Hyperic server.
- To configure the vRealize Hyperic server to use a keystore that you manage yourself for SSL communication, rather than Hyperic-generated keystores, set up a JKS format keystore for the vRealize Hyperic server on its host and import the SSL certificate for it. Make a note of the full path to the keystore, and its password. You will supply this information when you run the Hyperic installer (in `-full` mode).

The vRealize Hyperic server's keystore password and private key password must be the same, otherwise the vRealize Hyperic server's internal Tomcat-based server will be unable to start.

- Verify that the vRealize Hyperic database is available. During installation, the vRealize Hyperic server tests the database connection.

Procedure

- 1 Log in to vSphere Client as administrator.

You must not use `root` or `admin` as your login account name.

- 2 Select **File > Deploy OVF Template**.

The Deploy OVF Template wizard opens.

- 3 Proceed through the pages of the wizard, entering appropriate values. This table lists additional information to assist you to choose specific options.

Page	Change
Source	Enter the URL from which to download the Hyperic OVF file, or a disk location accessible from your computer
Name and Location	(Optional) Edit the name and select the folder location within the inventory where the vApp will reside.
Host/Cluster	Select the host or cluster on which to deploy the OVF template.

Page	Change
Disk Format	<p>Select one of the following disk formats to store the virtual machine virtual disks.</p> <ul style="list-style-type: none"> ■ Thick Provision Lazy Zeroed Creates a virtual disk in a default thick format. The space required for the virtual disk is allocated when the virtual disk is created. Data remaining on the physical device is not erased during creation, but is zeroed out on demand at a later time on first write from the virtual machine. Using the default flat virtual disk format does not zero out or eliminate the possibility of recovering deleted files or restoring old data that might be present on this allocated space. You cannot convert a flat disk to a thin disk. ■ Thick Provision Eager Zeroed A type of thick virtual disk that supports clustering features such as Fault Tolerance. Space required for the virtual disk is allocated at creation time. In contrast to the flat format, the data remaining on the physical device is zeroed out when the virtual disk is created. It might take much longer to create disks in this format than to create other types of disks. ■ Thin Provision Use this format to save storage space. For the thin disk, you provision as much datastore space as the disk would require based on the value that you enter for the disk size. However, the thin disk starts small and at first, uses only as much datastore space as the disk needs for its initial operations. If the thin disk needs more space later, it can grow to its maximum capacity and occupy the entire datastore space provisioned to it. Also, you can manually convert the thin disk into a thick disk.
Network Mapping	<p>Right-click the Destination Network column in the infrastructure, to select a network and set up the network mapping.</p>
IP Address Allocation	<p>Select one of the following options.</p> <ul style="list-style-type: none"> ■ Fixed A preferred option for production environments. You are prompted to enter the IP addresses in the Appliance Properties page. ■ Transient IP addresses are allocated from a specified range when the appliance is powered on. The IP addresses are released when the appliance is powered off. ■ DHCP A DHCP server is used to allocate the IP addresses.

4 Review the values that you have specified in the Ready to Complete page and optionally select **Power on after deployment** to start the server when the deployment process completes.

5 Click **Finish**.

What to do next

- 1 After successfully installing vRealize Hyperic server, delete `InstallerHome/logs/hq-install.log` and `hq-install.log.verbose`, or the whole the exploded installer. This is necessary to delete sensitive data that is written to the installation log files.
- 2 Install the vRealize Hyperic license. See [Activating and Configuring Your vRealize Hyperic License](#).
- 3 Log in to the vRealize Hyperic vApp. See [Log in to the vRealize Hyperic vApp](#).

Log in to the vRealize Hyperic vApp

The vRealize Hyperic vApp comprises two vApp machines, the vRealize Hyperic server and the vRealize Hyperic database. Log in to both vApps.

Procedure

- 1 Log in to the vApp machines using **root**.
- 2 Enter the password that you specified during deployment of the vRealize Hyperic server Administrator account.

What to do next

Start, stop or restart the vRealize Hyperic server.

Deploy vRealize Hyperic vApp in a Medium or Large Scale Environment

Use this procedure when you are deploying vRealize Hyperic vApp in a medium or large scale environment.

Prerequisites

- Verify that you are connected to the vRealize Hyperic server virtual machine using SSH and that the server is stopped.
- Power off the database virtual machine.

Procedure

- 1 On the **Hardware** tab of the database Virtual Machine Properties dialog box, edit the **Provisioned Size** setting to increase the hard disk size, and click **OK**.
 - Increase the disk size to 200 GB for a medium scale environment.
 - Increase the disk size to 500 GB for a large scale environment.The disk size is increased.
- 2 (Optional) If you are unable to perform step 1 because the setting cannot be edited, complete all the other steps in this procedure.
 - a Using SSH, connect to the ESX server on which the database virtual machine is installed.
 - b Change the hard disk adapter type from IDE to LSI logic.

This process changes the virtual IDE disk to a virtual SCSI disk. The procedure is described in the knowledge base at <http://kb.vmware.com/kb/1016192>.
 - c Edit the virtual machine settings to increase the hard disk size.
 - Increase the disk size to 200 GB for a medium scale environment.
 - Increase the disk size to 500 GB for a large scale environment.

- d Power on the database virtual machine.
- e Using SSH, connect to the database virtual machine.
- f Run the following commands in the order listed.

Command	Comment
<code>fdisk /dev/sda</code>	To run the fixed disk command utility.
<code>d</code>	To delete a partition.
<code>3</code>	To specify that the third partition is the partition to delete.
<code>n</code>	To add a new partition.
<code>p</code>	To specify that the partition is a primary partition.
<code>3</code>	To specify that the new partition is to be partition 3.

- g Press Enter twice to define the first and last sector of the partition.
- h Run the following commands in the order listed.

Command	Comment
<code>a</code>	To toggle the partition as bootable.
<code>3</code>	To specify the partition that is bootable.
<code>p</code>	(Optional) To print the partitions and verify is marked as a bootable device.
<code>w</code>	To write the table to disk and exit.
<code>reboot</code>	To initiate a full reboot of the virtual machine.

- i Using SSH, reconnect to the database virtual machine.
- j Run the following commands in the order listed.

Command	Comment
<code>resize2fs /dev/sda3</code>	To resize the file system.
<code>df -h</code>	To verify that the size of the file system is as you specified.

- 3 Using SSH, connect to the vRealize Hyperic server and start the server.

Run the vRealize Hyperic Installer Setup Script

You can make a new installation of the vRealize Hyperic server by running an installation script.

The `setup.bat` setup script for Windows and `setup.sh` setup script for UNIX-like environments is located in the vRealize Hyperic installation package. You can use the setup script to install the vRealize Hyperic server, the vRealize Hyperic agent, or both.

When you run the setup script, you can supply a qualifier that sets the installation mode:

Qualifier

-	<p>Run the setup script without a qualifier to perform a quick installation. Selected components will be installed with default values for most configuration options. If you install the vRealize Hyperic server, it will be configured to use its built-in vPostgreSQL database. A quick installation is useful if you are evaluating vRealize Hyperic.</p> <p>With a quick installation, by running the installer without a qualifier or with the <code>-postgresql</code> qualifier, the installer does not prompt you for the ports on which the vRealize Hyperic server listens for agent and user interface requests. The default ports are automatically configured.</p> <p>With a quick installation, you cannot specify the location and password for a user-managed server keystore. The vRealize Hyperic server uses a vRealize Hyperic-managed keystore.</p>
<code>-full</code>	<p>In full mode, the installer dialog prompts for all setup configuration options. Use this option if you are using an external database, or to configure the vRealize Hyperic server to use an SSL keystore that you manage yourself, rather than using a vRealize Hyperic-generated keystore.</p> <p>For best security and best configurability, run the installer in full mode.</p>
<code>-upgrade</code>	<p>Use upgrade mode to upgrade a 5.x vRealize Hyperic server to a later version. If you have an existing vRealize Hyperic server installation, the server configuration and the contents of the existing vRealize Hyperic database are preserved.</p>
<code>-updateScale</code>	<p>Use the <code>updateScale</code> option to change the sizing profile for the vRealize Hyperic server.</p>

Run the Installer Script

This vRealize Hyperic installer script uses the `full` qualifier. It can be used to install both the vRealize Hyperic server and the agent.

If you use a different mode, some of the prompts described in this procedure will not appear. Default values will be used instead for most configuration options.

Prerequisites

- If you are using an external PostgreSQL database, you must set it up before installing vRealize Hyperic server.
- To configure the vRealize Hyperic server to use a keystore that you manage yourself for SSL communication, rather than Hyperic-generated keystores, set up a JKS format keystore for the vRealize Hyperic server on its host and import the SSL certificate for it. Make a note of the full path to the keystore, and its password. You will supply this information when you run the Hyperic installer (in `-full` mode).

The vRealize Hyperic server's keystore password and private key password must be the same, otherwise the vRealize Hyperic server's internal Tomcat-based server will be unable to start.

- Define the server `HQ_JAVA_HOME` location.

vRealize Hyperic platform-specific server installers include a JRE, platform-independent installers do not. Depending on your environment and the installer you use, you may need to define the location of the JRE to ensure that the server can find the JRE to use. See [Configuring JRE Locations for vRealize Hyperic Components](#).

- Verify that the vRealize Hyperic database is available. During installation, the vRealize Hyperic server tests the database connection.

Procedure

- 1 Create a directory for the vRealize Hyperic installation.

On Windows operating systems, the installation dialog assumes your vRealize Hyperic installation directory is `c:\Program Files`.

On UNIX-like operating systems, the installation dialog assumes your vRealize Hyperic installation directory is `/home/hyperic`.

- 2 Unpack the tarball or ZIP archive.

Operating System	Description and Command
UNIX-like OS	Use GNU Tar to unpack vRealize Hyperic tarballs. <code>tar zxvf hyperic-hq-installer-4.x.y-xxx.tgz</code>
Windows OS	You must run the installer on a local drive.

- 3 Open a command shell where mode is one of the values in the table.

Operating System	Description and Command
UNIX-like OS	<code>PathToInstaller/setup.sh -full</code>
Windows OS	<code>PathToInstaller\setup.bat -full</code>

with the exception of upgrade .

- 4 Accept the VMware License agreement.
- 5 Follow the installer prompts, taking note of the following comments.

Prompt	Comment
Choose which software to install 1: Hyperic HQ Server 2: Hyperic HQ Agent	To install both the server and the agent, type 1, 2
HQ server installation path [default '.....']:	The account that you are running under must have write access the directory location.
Would you like us to use your own java keystore? [default '2'] 1: Yes 2: No	Type 1 to configure the server to use a certificate you manage, rather than generate its own. If you accept the default 2, a default keystore will be generated at <code>ServerHome/conf/hyperic.keystore</code> with the password <code>hyperic</code> .
Enter the base URL for the Hyperic server's web-based GUI [default...]	This value is used in alert notification emails. The value can be changed on the Administration page in the vRealize Hyperic portal.
Enter the fully qualified domain name of the SMTP server that Hyperic will use to send email messages [default FQDN.local]	If the installer does not find a local SMTP server, and you do not specify one, Hyperic cannot send alert notifications
What is the installation profile? default '1':]	If you are using the embedded PostgreSQL database, rather than an external database (required for production environments), select the <code>small</code> sizing profile.
Override the JDBC connection URL for the PostgreSQL database [default 'jdbc:postgresql://localhost: 5432/HQ?protocolVersion=2']:	Correct the URL, if necessary.

Prompt	Comment
Would you like to use an auto generated encryption key to encrypt the database password? [default '1']	If you accept the default, the installer will generate a key for encrypting the database password. Type 2 to supply the string yourself and, when prompted, type a string of at least 8 characters.
If the installer detects a database from a previous Hyperic installation, it will prompt you to: 1: Upgrade the HQ server database 2: Overwrite the HQ server database	Type 1 to preserve your existing vRealize Hyperic data. You also need to migrate your server and database to the new version. Type 2 to erase all of the data in your vRealize Hyperic database.

6 Exit the installer.

The installer indicates that the installation was successful. The URL for the vRealize Hyperic portal and the default username and password appear.

What to do next

- After successfully installing vRealize Hyperic server, delete `InstallerHome/logs/hq-install.log` and `hq-install.log.verbose`, or the whole the exploded installer. This is necessary to delete sensitive data that is written to the installation log files.
- Integrate vRealize Hyperic with your existing enterprise directory. The vRealize Hyperic server does not include a strength-of-password policy, or a lockout policy for failed login attempts.
- Install the vRealize Hyperic license.

Install vRealize Hyperic Server Using Windows Setup Wizard

You can use a Windows setup wizard to install the vRealize Hyperic server.

Prerequisites

- Verify that any existing vRealize Hyperic Windows service is removed. See [Remove an Existing vRealize Hyperic Windows Service](#).
- If you are using an external PostgreSQL database, you must set it up before installing vRealize Hyperic server.
- To configure the vRealize Hyperic server to use a keystore that you manage yourself for SSL communication, rather than Hyperic-generated keystores, set up a JKS format keystore for the vRealize Hyperic server on its host and import the SSL certificate for it. Make a note of the full path to the keystore, and its password. You will supply this information when you run the Hyperic installer (in `-full` mode).

The vRealize Hyperic server's keystore password and private key password must be the same, otherwise the vRealize Hyperic server's internal Tomcat-based server will be unable to start.

- Define the server `HQ_JAVA_HOME` location.

vRealize Hyperic platform-specific server installers include a JRE, platform-independent installers do not. Depending on your environment and the installer you use, you may need to define the location of the JRE to ensure that the server can find the JRE to use. See [Configuring JRE Locations for vRealize Hyperic Components](#).

- Verify that the vRealize Hyperic database is available. During installation, the vRealize Hyperic server tests the database connection.

Procedure

- 1 In Windows Explorer, double-click the `vRealize-hyperic-hqee-server-5.x.x.exe` file.
- 2 Follow the prompts in the wizard.
- 3 Click **Finish** to complete the installation.

What to do next

- 1 After successfully installing vRealize Hyperic server, delete `InstallerHome/logs/hq-install.log` and `hq-install.log.verbose`, or the whole the exploded installer. This is necessary to delete sensitive data that is written to the installation log files.
- 2 Start the vRealize Hyperic server.

Remove an Existing vRealize Hyperic Windows Service

Before you can install a new version of vRealize Hyperic server using the Windows setup wizard, you must remove any existing Windows vRealize Hyperic service.

Prerequisites

Stop the vRealize Hyperic server by running `hq-server.bat stop`.

Procedure

- ◆ Remove the Windows service for the previous instance by running `hq-server.bat remove`.

The service is removed.

What to do next

Install vRealize Hyperic server using the Windows setup wizard. See [Install vRealize Hyperic Server Using Windows Setup Wizard](#).

Using RPM to Install vRealize Hyperic Server

You can use a RedHat Package Manager package to install the vRealize Hyperic server.

Configure the vRealize Hyperic Server Properties File Before RPM Server Installation

It is good practice to configure the vRealize Hypericserver properties before you install the server using the RedHat Package Manage package.

The directory and file name in this procedure must be exactly as specified for the RPM installer to work correctly.

The vRealize Hyperic server installation settings are listed in [vRealize Hyperic Server Properties for RPM Installation](#).

Procedure

- 1 Running as root, or using sudo, run `mkdir -p etc/vmware/vcenter/hyperic` to create a directory for the properties file.
- 2 Specify the server installation settings in a file named `vcenter_hyperic_server.properties`.
- 3 Save your changes.

What to do next

Install the vRealize Hyperic server RPM. See, [Install the vRealize Hyperic Server RPM](#).

Install the vRealize Hyperic Server RPM

You can install vRealize Hyperic server from a RedHat Package Manager package.

By default, the installer configures the vRealize Hyperic server using a default properties file. You can create a customized properties file if you prefer. See [Configure the vRealize Hyperic Server Properties File Before RPM Server Installation](#).

Prerequisites

Before you run the installer, verify that the following prerequisites have been satisfied.

- The database has been configured. See [Set up the vRealize Hyperic Database](#).
- SSL has been configured. See [Configuring SSL Options](#).
- Verify that your SMTP server is listening on port 25 on the vRealize Hyperic server host.

Procedure

- 1 On the platform on which the vRealize Hyperic server is being installed, run `yum install vrealize-hyperic-server`.

`yum` resolves dependencies and displays the packages to install.
- 2 (Optional) If this is the first time that you have installed a vRealize Hyperic component on the virtual machine, accept the end user license agreement when prompted.
- 3 Type `y` at the prompt to start the installation.

What to do next

- 1 If you are prompted to do so, configure the vRealize Hyperic server properties. See [Configure the vRealize Hyperic Server Properties File After Installation](#).
- 2 After successfully installing vRealize Hyperic server, delete `InstallerHome/logs/hq-install.log` and `hq-install.log.verbose`, or the whole the exploded installer. This is necessary to delete sensitive data that is written to the installation log files.

vRealize Hyperic Server Properties for RPM Installation

Before installing the vRealize Hyperic server RPM package, you configure the vRealize Hyperic server properties.

This `vrealize_hyperic_server.properties` file lists the properties that can be configured for vRealize Hyperic server prior to installing the RPM package.

```
# Properties file for vRealize Hyperic Server Configuration
#
# This file must be place in /etc/vmware/vrealize/hyperic/ with a name of
# vrealize_hyperic_server.properties to be used by the vrealize-hyperic-server
# rpm for vRealize Hyperic Server configuration.
#
#
# To configure the build-in local Postgresql database uncomment the below
# sections. See below for other database types.
#
#####
# Configuration of local built-in Postgresql database

# Use the local built-in Postgresql database instead of other database types
BUILT_IN_POSTGRESQL=yes

# Do you accept the terms of the agreement?
HQ_ACCEPT_EULA=y

# HQ server installation path
HQ_SERVER_INSTALL_PATH=/opt/vmware/hyperic

# email address that HQ will use as the sender for email messages
HQ_SENDER_EMAIL_ADDRESS=hqadmin@eng.vmware.com

# username for the initial admin user HQ_ADMIN_USER=hqadmin

# password for the initial admin user HQ_ADMIN_PASSWORD=hqadmin Password must contain at least six
# characters.

# email address be for the initial admin user
HQ_ADMIN_EMAIL_ADDRESS=hqadmin@eng.vmware.com

# End of configuration for local built-in Postgresql database
#####
```

```

# To configure HQ with a local or remote database other than the built-in
# local instance of Postgresql comment out the above section and uncomment
# the properties in the section below. Supported databases include local or
# remote versions of vPostgresql or Postgresql.
#
#####
# For configuration with local or remote vPostgresql or Postgresql databases

### Do you accept the terms of the agreement?
#HQ_ACCEPT_EULA=y

### HQ server installation path
#HQ_SERVER_INSTALL_PATH=/opt/vmware/hyperic

### email address that HQ will use as the sender for email messages
#HQ_SENDER_EMAIL_ADDRESS=hqadmin@eng.vmware.com

### database type of [PostgreSQL]
#HQ_DB_TYPE=PostgreSQL

### database connection string
#HQ_DB_URL=jdbc:postgresql://localhost:5432/HQ?protocolVersion=2

### username to use to connect to the database
#HQ_DB_USERNAME=hqadmin

### password to use to connect to the database
#HQ_DB_PASSWORD=hqadmin

### username be for the initial admin user
#HQ_ADMIN_USER=hqadmin

### password be for the initial
admin user
#HQ_ADMIN_PASSWORD=hqadmin

### email address be for the initial admin user
#HQ_ADMIN_EMAIL_ADDRESS=hqadmin@eng.vmware.com

### HQ server installation profile [small|medium|large]
#HQ_SERVER_INSTALLATION_PROFILE=medium

# End of configuration
#####

```

Configure the vRealize Hyperic Server Properties File After Installation

If you did not configure the vRealize Hyperic server properties before running the RPM server installer, you are prompted to do so after installation. You cannot use the server until the properties have been specified.

You can find a sample of the `vfabric_hyperic_server.properties` file in [vRealize Hyperic Server Properties for RPM Installation](#). You can copy this file and edit it to complete this procedure.

Do not change installation directory property. The value of `HQ_SERVER_INSTALL_PATH` must remain `/opt/vmware/hyperic`.

Procedure

- 1 Create a file named `vrealize_hyperic_server.properties`.
- 2 Configure other settings in the file as required.

You can edit the properties file to configure Hyperic Server to use an external PostgreSQL database. If you do not change the value, the vRealize Hyperic server will use the built in local PostgreSQL database.

- 3 Save your changes, and copy `vrealize_hyperic_server.properties` to the `/etc/vmware/vrealize/hyperic/` directory.
- 4 Log in as root and in `/opt/hyperic/hyperic-hqee-installer`, run the `setup_from_properties_file.sh` script.

What to do next

Start the vRealize Hyperic server.

Using the vRealize Hyperic vApp Management Console

The vRealize Hyperic vApp provides a web interface for performing common server administration tasks.

The vApp management console is available only when the vApp is powered on. The functions provided by the vApp console can also be performed from a command shell.

- [Connect to the vRealize Hyperic vApp Management Console](#)

When you log in to the vApp management console, you can view system and network information for the vRealize Hypericserver, and can upgrade the server to a newer version.

- [Set the vRealize Hyperic Sever vApp Time Zone](#)

You can use the vApp management console to set the time zone to your system time.

- [View vRealize Hyperic Server vApp Network Status](#)

You can view general network details for the vApp, including information for each network interface on the vApp.

- [Manage vRealize Hyperic vApp Network Address Settings](#)

You can configure the method by which the vApp management console obtains its IP address.

- [Configure vRealize Hyperic vApp Proxy Settings](#)

You can configure a proxy server for HTTP communications between the vApp and the internet.

- [Reboot or Shutdown the vRealize Hyperic Server](#)

You can shutdown or reboot the vRealize Hyperic from the vApp management console.

Connect to the vRealize Hyperic vApp Management Console

When you log in to the vApp management console, you can view system and network information for the vRealize Hypericserver, and can upgrade the server to a newer version.

Prerequisites

You must have root admin credentials to log on to the vApp management console.

Procedure

- ◆ Log in to the management console using the URL `https://host:5480`, where `host` is the IP address or DNS name of the vRealize Hyperic vApp host.

You must log on as `root` and supply the password that was defined for the vRealize Hyperic admin account. The default is `hqadmin`.

Set the vRealize Hyperic Sever vApp Time Zone

You can use the vApp management console to set the time zone to your system time.

Prerequisites

You must be connected to the vRealize Hyperic vApp management console. See [Connect to the vRealize Hyperic vApp Management Console](#).

Procedure

- 1 On the vRealize Hyperic vApp management console, select the **System** tab and click **Time Zone**.
- 2 Select a time zone from the **System Time Zone** menu.
- 3 Click **Save Settings** to apply your changes.

You can click **Cancel Changes** to retain your existing settings.

View vRealize Hyperic Server vApp Network Status

You can view general network details for the vApp, including information for each network interface on the vApp.

Prerequisites

You must be connected to the vRealize Hyperic vApp management console. See [Connect to the vRealize Hyperic vApp Management Console](#).

Procedure

- 1 On the vRealize Hyperic vApp management console, select the **Network** tab and click **Status**.
Details of the network status and network interfaces appear.
- 2 (Optional) Click **Refresh** to update the displayed information.

Manage vRealize Hyperic vApp Network Address Settings

You can configure the method by which the vApp management console obtains its IP address.

Prerequisites

You must be connected to the vRealize Hyperic vApp management console. See [Connect to the vRealize Hyperic vApp Management Console](#).

Procedure

- 1 On the vRealize Hyperic vApp management console, select the **Network** tab and click **Address**.
- 2 Use the dropdown menus to select the required network address settings and click **Save Settings**.
You can click **Cancel Changes** to retain your existing settings.

Configure vRealize Hyperic vApp Proxy Settings

You can configure a proxy server for HTTP communications between the vApp and the internet.

Prerequisites

You must be connected to the vRealize Hyperic vApp management console. See [Connect to the vRealize Hyperic vApp Management Console](#).

Procedure

- 1 On the vRealize Hyperic vApp management console, select the **Network** tab and click **Proxy**.
- 2 Select the **Use a proxy server** check box.
- 3 Type appropriate strings in the proxy settings text boxes.
The **HTTP Proxy Server** and **Proxy Port** parameters are mandatory.

- 4 Click **Save Settings** to apply your changes.

You can click **Cancel Changes** to retain your existing settings.

Reboot or Shutdown the vRealize Hyperic Server

You can shutdown or reboot the vRealize Hyperic from the vApp management console.

Prerequisites

You must be connected to the vRealize Hyperic vApp management console. See [Connect to the vRealize Hyperic vApp Management Console](#).

Procedure

- 1 On the vRealize Hyperic vApp management console, select the **System** tab and click **Information**.
- 2 Click **Reboot** or **Shutdown**, according to your requirements.

Install and Configure the vRealize Hyperic Agent

You must perform several tasks to install the vRealize Hyperic agent.

The tasks must be performed in this order.

Procedure

- 1 [Prepare to Install the vRealize Hyperic Agent](#)

Before you can install the vRealize Hyperic agent, you must perform preparatory tasks.

- 2 [Select a vRealize Hyperic Agent Installer Package](#)

You can download the vRealize Hyperic agent in a variety of packages.

- 3 [Configure the vRealize Hyperic Agent to Server Communication Properties](#)

Prior to first agent startup, you can define the properties that enable the vRealize Hyperic agent and vRealize Hyperic server to communicate with each other, and other agent behaviors, in the `agent.properties` file of an agent, . There are a number of steps to complete the configuration.

- 4 [\(Optional\) Configure Unidirectional Communication](#)

You can configure the vRealize Hyperic agent to initiate all communications with the server. You configure unidirectional communications at first startup. Unidirectional communications are always via SSL.

- 5 [\(Optional\) Configure a vRealize Hyperic Agent Keystore](#)

You can configure your own keystore for the vRealize Hyperic agent to use, instead of having the agent generate and use a self-signed certificate for SSL communication with the vRealize Hyperic server.

- 6 [\(Optional\) Configure the vRealize Hyperic Agent Using the Configuration Dialog](#)

The agent configuration dialog appears in the shell when you launch a vRealize Hyperic agent that lacks the configuration values that specify the location of the vRealize Hyperic server. The dialog queries for the address and port of the vRealize Hyperic server, and other connection-related data.

Prepare to Install the vRealize Hyperic Agent

Before you can install the vRealize Hyperic agent, you must perform preparatory tasks.

Prerequisites

- To configure the vRealize Hyperic agent to use a keystore that you manage yourself for SSL communication, rather than a vRealize Hyperic-generated keystore, set up a JKS-format keystore for the vRealize Hyperic agent on its host and import its SSL certificate. Make a note of the full path to the keystore, and its password. You configure this data in the agent's `agent.properties` file.

Verify that the agent keystore password and the private key password are identical.

- Define the agent `HQ_JAVA_HOME` location.

vRealize Hyperic platform-specific server installers include JRE 1.7.x. Platform-independent installers do not. Depending on your environment and the installer you use, you may need to define the location of the JRE to ensure that the agent can find the JRE to use. See [Configuring JRE Locations for vRealize Hyperic Components](#).

- Verify if you need to open a firewall.

If a firewall is blocking incoming traffic to a platform on which you are installing vRealize Hyperic agents, you must open the agent listen port (by default, 2144 for plain text, or 2443 for SSL) so that the agent will accept connections from the vRealize Hyperic server.

In Windows environments, you must open the agent listen port. The default behavior for a firewall built into Windows is to block remote connections.

Select a vRealize Hyperic Agent Installer Package

You can download the vRealize Hyperic agent in a variety of packages.

You can select how to install the vRealize Hyperic agent from one of the following agent installer package options.

Prerequisites

Verify that all the prerequisites described in [Prepare to Install the vRealize Hyperic Agent](#) are satisfied.

- [Install a vRealize Hyperic Agent-Only Package](#)

You can install the vRealize Hyperic agent from an agent-only tar.gz archive for non-Windows systems, or from a ZIP archive for Windows systems.

- [Run the vRealize Hyperic Installer](#)

You can install the vRealize Hyperic agent using the vRealize Hyperic installer.

- [Install the vRealize Hyperic Agent RPM](#)

You can install vRealize Hyperic agent from a RedHat Package Manager (RPM) package. The agent in the package does not include a JRE.

Install a vRealize Hyperic Agent-Only Package

You can install the vRealize Hyperic agent from an agent-only tar.gz archive for non-Windows systems, or from a ZIP archive for Windows systems.

Agent-only archives are useful when you roll out agents to a large number of platforms with various operating systems and architectures. Agent archives are available for Windows and UNIX-like environments, with and without built-in JREs.

If you have multiple agents to install, see [Install Multiple vRealize Hyperic Agents Simultaneously](#).

This procedure can be used for both tar.gz and ZIP archives.

Prerequisites

Verify that you have satisfied all the prerequisites listed in [Prepare to Install the vRealize Hyperic Agent](#).

Procedure

- 1 Create a directory for the vRealize Hyperic installation.
- 2 Unpack the archive into the agent directory.
Use GNU Tar to unpack the tar.gz archive.

Run the vRealize Hyperic Installer

You can install the vRealize Hyperic agent using the vRealize Hyperic installer.

The installer is used to install both the server and the agent. See [Run the Installer Script](#).

Prerequisites

Verify that you have satisfied all the prerequisites listed in [Prepare to Install the vRealize Hyperic Agent](#).

Install the vRealize Hyperic Agent RPM

You can install vRealize Hyperic agent from a RedHat Package Manager (RPM) package. The agent in the package does not include a JRE.

The RPM performs the following actions:

- Creates the user and group named `hyperic` if they do not exist.
- Sets the home directory of the `hyperic` user to `/opt/hyperic`.
- Installs the agent files into `/opt/hyperic/hyperic-hqee-agent`.
- Installs an init script to `/etc/init.d/hyperic-hqee-agent`.
- Adds the init script to `chkconfig` and sets it to on for run levels 2, 3, 4, and 5.

Prerequisites

- Verify that you have satisfied all the prerequisites listed in [Prepare to Install the vRealize Hyperic Agent](#).

- Agent hosts must have the J2RE virtual package installed.
- Verify that a JDK or JRE is installed on the platform.

Procedure

- 1 Run `yum install vfabric-hyperic-agent` to install the agent on the platform that the agent will monitor.
- 2 Log on as root and edit the `/etc/init.d/hyperic-hqee-agent` file to set the `HQ_JAVA_HOME` parameter to the home directory of the JDK or JRE that the agent is to use.

What to do next

Configure the vRealize Hyperic agent in the `properties` file. See [Activate Agent to Server Communication Properties](#).

Configure the vRealize Hyperic Agent to Server Communication Properties

Prior to first agent startup, you can define the properties that enable the vRealize Hyperic agent and vRealize Hyperic server to communicate with each other, and other agent behaviors, in the `agent.properties` file of an agent, . There are a number of steps to complete the configuration.

Some agent properties can be defined either before or after the initial start up. However, you must always configure properties that control the following behaviors before initial startup.

- When the agent must use an SSL keystore that you manage, rather than a vRealize Hyperic-generated keystore.
- When the agent must manage VMware vSphere components.
- When the agent must connect to the vRealize Hyperic server via a proxy server.

If you are deploying multiple agents, consider configuring the agent in its `properties` file, to streamline the process.

Prerequisites

The vRealize Hyperic server must be running.

What to do next

Start the vRealize Hyperic agent.

Open the Agent Properties File

Each time that you install a new vRealize Hyperic agent, you must configure the `agent.properties` file that contains data controlling agent to server communication. If the agent does not have a properties file, create one.

A vRealize Hyperic agent looks for its properties file in two locations, in this order:

1 HqUserHome/.hq

If this directory exists and contains `agent.properties`, the vRealize Hyperic agent uses the property values defined there.

2 AgentHome/conf

This is the default location of `agent.properties`.

If the agent does not find the the required properties for establishing communications with the vRealize Hyperic server in either of these locations, it prompts for the property values at initial start up of the agent.

It is good practice to configure the properties in the `agent.properties` file before initial start up of the agent. This is a requirement for some communication actions, and is the most efficient process when you are deploying multiple agents.

Procedure

- ◆ Make a copy of the `agent.properties` file from the agent installation.

Activate Agent to Server Communication Properties

In the `agent.properties` file, properties relating to communication between the vRealize Hyperic agent and server are inactive by default. You must activate them.

Procedure

- 1 In the `agent.properties` file, locate the following section.

```
## Use the following if you'd like to have the agent setup
## automatically from these properties. The values for these
## properties are used to answer the setup questions
##
## If any of these properties are left undefined, the setup
## process will prompt for their values
##
## If the value that should be used is the default when interactive
## setup is done, use the string *default* as the value for the option
```

- 2 Activate the following properties by removing the hash tag at the beginning of each line.

```
#agent.setup.camIP=localhost
#agent.setup.camPort=7080 #agent.setup.camSSLPort=7443
#agent.setup.camSecure=yes #agent.setup.camLogin=hqadmin
#agent.setup.camPword=hqadmin #agent.setup.agentIP=*default*
#agent.setup.agentPort=*default*
#agent.setup.resetupTokens=no
```

The first time that you start the vRealize Hyperic agent, if `agent.setup.camPword` is inactive, and has a plain text value, the agent encrypts the value.

Specify the vRealize Hyperic Agent Communication Properties

The `agent.properties` file contains properties that you can configure to govern both agent-initiated and server-initiated communication.

The properties described here are the minimum required for agent-server communication.

Procedure

- 1 Specify the location and credentials the vRealize Hyperic agent must use to contact the vRealize Hyperic server.

Property	Property Definition
<code>agent.setup.camIP</code>	Specify the address or hostname of the vRealize Hyperic server.
<code>agent.setup.camPort</code>	The default value is the standard plain text vRealize Hyperic server listen port. If your server is configured for a different listen port, specify the port number.
<code>agent.setup.camSSLPort</code>	The default value is the standard SSL vRealize Hyperic server listen port. If your server is configured for a different listen port, specify the port number.
<code>agent.setup.camSecure</code>	The default value is yes (use SSL). SSL configuration is strongly recommended, and is required if you are going to configure the agent for unidirectional communications. Change the value to no if you do not require the agent to use secure communications when contacting the vRealize Hyperic server.
<code>agent.setup.camLogin</code>	Specify the user name for the agent to use when connecting to the server. If you change the value from the <code>hqadmin</code> default value, verify that the user account is correctly configured on the vRealize Hyperic server.
<code>agent.setup.camPword</code>	Specify the password for the agent to use, together with the user name specified in <code>agent.setup.camLogin</code> , when connecting to the server. Verify that the password is the one configured in vRealize Hyperic for the user account.

- (Optional) Specify the address or hostname and the listen port for the vRealize Hyperic server to use to contact the vRealize Hyperic agent.

If you are creating a standard `agent.properties` file that can be used for all agents, deactivate these properties, but do not change their values.

Property	Property Definition
<code>agent.setup.agentIP</code>	If you leave the default setting * default* the agent detects an IP address on the platform and selects it as the listen address.
<code>agent.setup.agentPort</code>	If you leave the default setting * default* the agent uses the default listen port (2144 for plain text, or 2443 for SSL) as its listen address. If that port is unavailable, the agent detects a free port and selects it as its listen port.

(Optional) Configure Unidirectional Communication

You can configure the vRealize Hyperic agent to initiate all communications with the server. You configure unidirectional communications at first startup. Unidirectional communications are always via SSL.

As an alternative to this procedure, you can specify unidirectional communications via the agent prompt that appears at startup.

Procedure

- Stop the agent.

- Remove the agent's `\data` directory.

If you remove the `\data` directory, at next startup, the agent checks its `agent.properties` file for the startup settings that it requires to connect to the server. If the properties file does not contain the startup settings, it opens a shell to prompt for them.

- In the `agent.properties` file, set `setagent.setup.unidirectional=yes`, and start the agent.

- In the user interface, on the Inventory tab of the platform, navigate to **Type & Network Properties** and click **Edit**.

The **Agent Connection** menu shows the currently selected port for bidirectional communications. For example, `10.2.0.213:2144`, where `10.2.0.213` is the IP address of the platform, and `2144` is the bidirectional port number previously used.

- Expand the menu and select the entry that shows the same IP address, and `-1` as the port number.

For example, `10.2.0.213:-1`.

The agent uses unidirectional communication with the server.

Configure Bidirectional Communication

If you specified that the vRealize Hyperic agent use unidirectional communication with the vRealize Hyperic server, you can revert the setting so that it uses bidirectional communication.

As an alternative to this procedure, you can specify bidirectional communications via the agent prompt that appears at startup.

Procedure

1 Stop the agent.

2 Remove the agent's `\data` directory.

If you remove the `\data` directory, at next startup, the agent checks its `agent.properties` file for the startup settings that it requires to connect to the server. If the properties file does not contain the startup settings, it opens a shell to prompt for them.

3 In the `agent.properties` file, set `setagent.setup.unidirectional=no`, and start the agent.

4 In the user interface, on the Inventory tab of the platform, navigate to **Type & Network Properties** and click **Edit**.

The **Agent Connection** menu shows the currently selected port for unidirectional communications. For example, `10.2.0.213:-1`, where `10.2.0.213` is the IP address of the platform, and `-1` is the unidirectional port number previously used.

5 Expand the menu and select the entry that shows the same IP address, and 2144 as the port number. For example, `10.2.0.213:2144`.

If you configured a port number other than the default, select the entry with that port number.

The agent uses bidirectional communication with the server.

(Optional) Configure a vRealize Hyperic Agent Keystore

You can configure your own keystore for the vRealize Hyperic agent to use, instead of having the agent generate and use a self-signed certificate for SSL communication with the vRealize Hyperic server.

Procedure

1 In the `agent.properties` file, activate the `# agent.keystore.path=` and `# agent.keystore.password=` properties.

Define the full path to the keystore with `agent.keystore.path` and the keystore password with `agent.keystore.password`.

2 (Optional) If you configured the agent for unidirectional communication, add `[agent.keystore.alias` to the properties file, and set it to the alias of the primary certificate or private key entry of the keystore's primary certificate.

3 Verify that `agent.setup.acceptUnverifiedCertificate` is `false`.

(Optional) Configure the vRealize Hyperic Agent Using the Configuration Dialog

The agent configuration dialog appears in the shell when you launch a vRealize Hyperic agent that lacks the configuration values that specify the location of the vRealize Hyperic server. The dialog queries for the address and port of the vRealize Hyperic server, and other connection-related data.

The agent configuration dialog appears in these cases:

- The first time you start an agent, if you have not supplied the properties in `agent.profile`.
- When you start an agent for which saved server connection data is corrupt or has been removed.
- When you run the agent launcher with the `setup` option, which causes the agent to prompt for new connection property settings.

You can also run the agent launcher to re-run the configuration dialog.

Procedure

- 1 Open a terminal window on the platform on which the agent is installed.
- 2 Navigate to the `AgentHome/bin` directory.
- 3 Run the agent launcher using the `start` or `setup` option:

Platform	Command
UNIX-like	<code>sh hq-agent.sh start</code>
Windows	Install the Windows service for the agent, then start it: <code>sh hq-agent.bat install</code> <code>sh hq-agent.bat start</code>

- 4 Respond to the prompts, noting the following as you move through the process.

Prompt	Description
What is the HQ server IP address Enter the listen address of your Hyperic Server.	The server must be running. If the server is on the same machine as the agent, you can type <code>localhost</code> . If there is a firewall blocking traffic from the agent to the server, specify the address of the firewall.
Should Agent communications to HQ always be secure [default=yes].	Accept the default to configure vRealize Hyperic to use SSL for agent-to-server communication. It is good security practice to configure SSL, and is required if you configured unidirectional communications. To configure vRealize Hyperic to use plain HTTP for agent-to-server communication, type <code>no</code> .

Prompt	Description
What IP should HQ use to contact the agent [default=n.n.n.n]	The prompt shows the first IP address that the agent detects on the host. You can specify another IP address on the host. If a firewall is blocking traffic from the server to the agent, type the IP address of the firewall, and configure the firewall to forward traffic intended for the vRealize Hyperic agent to the listen address of the agent host.
What port should HQ use to contact the agent [default=2144]	Specify the agent port for the vRealize Hyperic server to use when it initiates contact with the agent. Specify the port that the agent binds to at startup, which by default is 2144. If you have previously edited <code>agent.properties</code> to define a different listen port, using the <code>agent.listenPort</code> property, specify that value. If there is a firewall blocking traffic from the server to the agent, configure the device to forward traffic on TCP port 2144 to the vRealize Hyperic agent.

The vRealize Hyperic agent initiates a connection to the vRealize Hyperic server and the server verifies that it can communicate with the agent.

The vRealize Hyperic agent discovers the platform and supported products running on it.

Configuring SSL Options

vRealize Hyperic supports the use of SSL communication for both server-to-agent and agent-to-server communications. It is good practice to configure vRealize Hyperic components to communicate with each other using SSL as part of the installation process.

Server-to-agent communication is always SSL.

You configure SSL for agent-to-server communication when you configure agent-server communications.

The vRealize Hyperic agent can manage products over SSL if it is supported by the product plug-in.

When the vRealize Hyperic server and a vRealize Hyperic agent communicate over SSL, each component validates the other's SSL certificate.

vRealize Hyperic Certificate Processing

The first time a vRealize Hyperic agent initiates a connection to the vRealize Hyperic server following installation, the server presents its SSL certificate to the agent. If the agent trusts the certificate that the server presented, the agent imports the server's certificate into its own keystore.

The agent trusts a server certificate:

- If that certificate already exists in the agent's keystore.
- If the certificate has the same CA as the agent's certificate.

By default, if the agent does not trust the certificate presented by the server, the agent issues a warning. You can terminate the configuration process and configure SSL. The vRealize Hyperic server and the vRealize Hyperic agent do not import untrusted certificates unless you respond yes to the warning prompt.

It is possible to configure both components to accept untrusted certificates automatically, without warning. For security reasons, this practice is strongly discouraged. Check the values of `agent.setup.acceptUnverifiedCertificate` (in `AgentHome/conf/agent.properties`) and `accept.unverified.certificates` in `ServerHome/conf/hq-server.conf`.

vRealize Hyperic Server and SSL

If you are using the standard vRealize Hyperic `setup.sh` or `setup.bat` installer, you install the vRealize Hyperic server's keystore before installing the server.

If you do not configure the server to use an existing keystore, and supply its location and password during server installation, the vRealize Hyperic installer creates a keystore for the server with a self-signed certificate. The keystore, named `hyperic.keystore`, is located in `ServerHome/conf` and uses the password `hyperic`. The server presents the self-signed certificate when communicating with agents.

vRealize Hyperic Agent and SSL

To use SSL for agent-to-server communication, you install the vRealize Hyperic agent's keystore prior to first startup. If you use the vRealize Hyperic-generated keystores, you will need to update the password for each generated keystore.

Configure SSL for vRealize Hyperic Agent to Server Communication

You can configure your vRealize Hyperic agents to use SSL when communicating with the vRealize Hyperic server.

You must configure SSL for each vRealize Hyperic agent.

Prerequisites

- Verify that the vRealize Hyperic agent's keystore was installed prior to initial startup of the agent. Each agent must have its own keystore.
- Verify that you have SSL certificates for the vRealize Hyperic server and each vRealize Hyperic agent.
- Verify that you have a JKS-format keystore for the vRealize Hyperic server on its host, and that you have imported its SSL certificate.
- Note the full path to the JKS-format keystore and its password. When you run the vRealize Hyperic installer in `-full` mode, the installer prompts for this information.

Procedure

- 1 Setup a keystore for the vRealize Hyperic agent on its host.
- 2 Import the SSL certificate for the agent.

- 3 In the agent's `agent.properties` file, specify values for the following properties.

Property	Value
<code>agent.keystore.path</code>	Specify the location of the agent keystore.
<code>agent.keystore.password</code>	Specify the password for the agent keystore. The password of the keystore for the vRealize Hyperic agent and the private key password must be the identical.

- 4 (Optional) If you are configuring the vRealize Hyperic agent for unidirectional communication, specify the keystore name in the `agent.keystore.alias` property.
- 5 Save the file and restart the agent.

Change a vRealize Hyperic SSL Certificate to Use a User-Managed Keystore

If you do not configure the vRealize Hyperic server and vRealize Hyperic agents to use keystores that you create and manage before you first start the server and agents, vRealize Hyperic will generate default keystores with self-signed certificates. You can change the SSL certificates to use a user-managed keystore.

Prerequisites

Verify that you have a a trusted PKC12-format keystore for vRealize Hyperic server, and that an SSL certificate of the correct format is installed the the vRealize Hyperic server host.

Procedure

- 1 Open `ServerHome/conf/hq-server.conf` in a text editor and make the following changes.
 - a Set the value of `accept.unverified.certificates` to `false`.
 - b Define the location of your trusted keystore with the `server.keystore.path` property.
 - c Define the password for your trusted keystore with the `server.keystore.password` property.
 - d Save your changes and restart the vRealize Hyperic server.
- 2 For each vRealize Hyperic agent reporting to the vRealize Hyperic server
 - a Obtain an SSL certificate from your CA and install it on the vRealize Hyperic agent host.
 - b Open `AgentBundle/AgentHome/agent.properties` in a text editor.
 - c Set the value of `agent.setup.acceptUnverifiedCertificate` to `"false"`.
 - d Define the location of your trusted keystore with the `agent.keystore.path` property.
 - e Define the password for your trusted keystore with the `agent.keystore.password` property.
 - f Save your changes and restart the vRealize Hyperic agent.

Change SSL Configuration from a User-Managed Keystore to a vRealize Hyperic Keystore

You can change your SSL configuration from using a user-managed keystore to a vRealize Hyperic-generated keystore.

For best security, do not configure vRealize Hyperic to use self-signed certificates.

Prerequisites

Verify that the vRealize Hyperic and the vRealize Hyperic agents are stopped.

Procedure

- 1 Open `ServerHome/conf/hq-server.conf` in a text editor.
- 2 Set the value of `accept.unverified.certificates` to `true`.
- 3 Restart the vRealize Hyperic server.
- 4 For each vRealize Hyperic agent reporting to the vRealize Hyperic server, open `AgentBundle/AgentHome/agent.properties` in a text editor.
- 5 Set the value of `agent.setup.acceptUnverifiedCertificate` to `true`.
- 6 Save your changes and restart the agents.

Install Multiple vRealize Hyperic Agents Simultaneously

If you have multiple agents to install at one time, you can create a single standardized `agent.properties` file that can be used by all the agents.

Installing multiple vRealize Hyperic agents entails a number of steps, as described below. Perform the steps in the order listed.

Prerequisites

Verify that the following prerequisites are satisfied.

- 1 Set up an install server.

An install server is a server that can access the the target platforms from which to perform remote installation.

The server must be configured with a user account that has permissions to SSH into each target platform without requiring a password.

- 2 Verify that each target platform on which a vRealize Hyperic agent will be installed has the following items.
 - A user account that is identical to that created on the install server.
 - An identically named installation directory, for example `/home/vrealizeyperic`.
 - A trusted keystore, if required.

Procedure

1 [Create a Standard vRealize Hyperic Agent Properties File](#)

You can create a single properties file that contains property values that are used by multiple agents.

2 [\(Optional\) Deploy and Start Multiple Agents One-By-One](#)

You can perform remote installations to deploy agents that use a single `agent.properties` file one-by-one.

3 [\(Optional\) Deploy and Start Multiple Agents Simultaneously](#)

You can perform remote installations to simultaneously deploy agents that use a single `agent.properties` file.

Create a Standard vRealize Hyperic Agent Properties File

You can create a single properties file that contains property values that are used by multiple agents.

To enable mass agent deployment, you create an `agent.properties` file that defines the agent properties required for the agent to start up and connect with the vRealize Hyperic server. If you supply the necessary information in the properties file, each vRealize Hyperic agent will locate its setup configuration at startup, rather than prompting you for the location. You can create a standard agent profile that you can copy to the agent installation, or to a location available to the agent installation.

Prerequisites

Verify that the prerequisites in [Install Multiple vRealize Hyperic Agents Simultaneously](#) have been satisfied.

Procedure

- 1 Create an `agent.properties` file in `HqUserHome/.hq` on the install server.
- 2 Configure the properties as required. The minimum configuration is the IP address and port of the vRealize Hyperic install server.

Do not specify values for the agent listen address and port. At first startup, if explicit values for IP address and port are not set, the vRealize Hyperic agent - which detects the network interfaces on the platform - uses the first detected interface as its listen address, and port 2144 or 2443 as its listen port, depending on whether you configure the agent for plain text or SSL communications.

- 3 Save your configurations.

The first time that the agents are started, they read the `agent.properties` file to identify the server connection information, will connect to the server and register themselves.

What to do next

Perform remote agent installations. See [\(Optional\) Deploy and Start Multiple Agents One-By-One](#) or [\(Optional\) Deploy and Start Multiple Agents Simultaneously](#).

(Optional) Deploy and Start Multiple Agents One-By-One

You can perform remote installations to deploy agents that use a single `agent.properties` file one-by-one.

Prerequisites

Verify that the prerequisites in [Install Multiple vRealize Hyperic Agents Simultaneously](#) have been satisfied.

You must have configured a standard agent properties file and copied it to the agent installation, or to a location available to the agent installation

Procedure

- 1 Log in to the install server user account that you configured with permissions to SSH into each target platform without requiring a password.
- 2 SSH to the remote platform.
- 3 Copy the agent archive to the agent host.
- 4 Unpack the agent archive.
- 5 Copy the `agent.properties` file to the `/.hq` directory under the home directory of the standard agent installation user account.
- 6 Start the new agent.

The vRealize Hyperic agent registers itself with the vRealize Hyperic server and the agent runs an autodiscovery scan to discover its host platform and supported managed products that are running on the platform.

What to do next

Check the Auto-Discovery portlet in the vRealize Hyperic dashboard to verify that the platform was discovered.

(Optional) Deploy and Start Multiple Agents Simultaneously

You can perform remote installations to simultaneously deploy agents that use a single `agent.properties` file.

Prerequisites

Verify that the prerequisites in [Install Multiple vRealize Hyperic Agents Simultaneously](#) have been satisfied.

You must have configured a standard agent properties file and copied it to the agent installation, or to a location available to the agent installation

Procedure

- 1 Create a `hosts.txt` file on your install server that maps the hostname to the IP address of each platform on which you are installing an agent.
- 2 Open a command line shell on the install server.
- 3 Type the following command in the shell, supplying the correct name for the agent package in the `export` command.

```
$ export AGENT=hyperic-hq-agent-4.6.0-x86-linux.tgz
$ for host in `cat hosts.txt`; do scp $AGENT $host:</path/to/agent/install>
&& ssh $host "tar zxf $AGENT && ./hyperic-hq-agent-4.6.0/hq-agent.sh start"; done
```

- 4 (Optional) If the target hosts have sequential names, for example `host001`, `host002`, `host003`, and so on, you can skip the `hosts.txt` file and use the `seq` command, as follows.

```
$ export AGENT=hyperic-hq-agent-4.6.0-x86-linux.tgz
$ for i in `seq 1 9`; do scp $AGENT host$i: && ssh host$i "tar zxf $AGENT &&
./hyperic-hq-agent-4.6.0/hq-agent.sh start"; done
```

The vRealize Hyperic agents registers themselves with the vRealize Hyperic server and the agents run an autodiscovery scan to discover their host platform and supported managed products that are running on the platform.

What to do next

Check the Auto-Discovery portlet in the vRealize Hyperic dashboard to verify that the platforms were discovered.

Activating and Configuring Your vRealize Hyperic License

vRealize Hyperic is part of several VMware products, including vRealize Operations Management Suite, vCloud Suite, and vRealize Suite, each with its own licensing mechanism. vRealize Hyperic is also available as a standalone product. You configure the license according to the requirements of the product of which it is a part.

- If your vRealize Hyperic is part of vRealize Operations Management Suite, see [Configure the vRealize Hyperic License for vRealize Operations Manager](#).
- If your vRealize Hyperic is part of vCloud Suite, see [Configure the vRealize Hyperic License for vCloud Suite](#).
- If your vRealize Hyperic is part of vRealize Suite, see [Configure the vRealize Hyperic License for vRealize Suite](#).

If you have more than one license, you configure each of them.

vRealize Hyperic is licensed on a per managed platform basis, where a platform is:

- A physical machine or a virtual machine on which a vRealize Hyperic agent is running.

If an agent manages a vSphere vCenter instance, it consumes a license for the platform that hosts vCenter, a license for each vSphere vHost that is administered by the vCenter instance and, if an agent is installed on each virtual machine, a license for each vSphere virtual machine on each vHost.

- A network device or network host that is managed remotely by a vRealize Hyperic agent. See *vRealize Hyperic Resource Configuration and Metrics* for information about vRealize Hyperic functionality for managing remote devices and hosts.

If you have licenses for both vRealize Operations Management Suite and vRealize Suite, you are licensed for the total number of licenses provided by your vRealize Hyperic license plus the number of licenses provided in your vRealize Operations Management Suite license.

Configure the vRealize Hyperic License for vRealize Operations Manager

You configure the vRealize Hyperic license in VMware vRealize Operations Management Suite by editing the `hq-server.conf` file.

Prerequisites

Stop the vRealize Hyperic server.

Procedure

- 1 Open the `ServerHome/conf/hq-server.conf` file for editing.
- 2 Add the line `vcops.license.key=LicenseKey`, where *LicenseKey* is the vRealize Operations Management Suite license key.

What to do next

Restart the vRealize Hyperic server.

Configure the vRealize Hyperic License for vCloud Suite

You configure the vRealize Hyperic license in vCloud Suite by editing the `hq-server.conf` file.

Prerequisites

Stop the vRealize Hyperic server.

Procedure

- 1 Open the `ServerHome/conf/hq-server.conf` file for editing.
- 2 Add the line `vccloud.license.key=LicenseKey`, where *LicenseKey* is the vCloud Suite license key.

What to do next

Restart the vRealize Hyperic server.

Configure the vRealize Hyperic License for vRealize Suite

The method you use to activate your license is dependent on whether the product was acquired as standalone, or as part of vRealize Suite.

The vRealize Hyperic evaluation distributions include a time-limited license for 60 platforms.

After you purchase vRealize Hyperic, a production license specifies the number of platforms that you may manage and, unless you have a perpetual license, the license expiration date.

Prerequisites

- If you obtained your vRealize Hyperic license as part of vRealize Suite, refer first to the license information and procedure in *Getting Started with vRealize Suite*. If necessary, complete these additional licensing tasks.
- Stop the vRealize Hyperic server.

Procedure

- (Optional) If you acquired vRealize Hyperic as a standalone product, create a file named `vf.hyp-serial-numbers.txt` that contains the product serial number provided by VMware. Install the file in one of the following directories, as appropriate. If the file does not exist, create it.
 - `/etc/opt/vmware/vfabric/` for Unix-like platforms.
 - `%ALLUSERSPROFILE%\vmware\vfabric` on Windows-like platforms
- (Optional) If you acquired vRealize Hyperic as part of the vRealize Suite, configure the location of the VMware license server that administers the network license for vRealize Suite by adding the `vrealize.licenseserver.url` property to `ServerHome/conf/hq-server.conf`.

What to do next

Restart the vRealize Hyperic server.

View License Terms

You can view the terms of your license and usage on the **Administration** tab of the vRealize Hyperic user interface.

Migrating and Upgrading vRealize Hyperic Components

3

It is good practice to upgrade your vRealize Hyperic components. New versions offer enhancements to existing functionality, performance, and so on.

- [Upgrading vRealize Hyperic Components and Migrating the Database](#)

Upgrading your existing vRealize Hyperic installation to the most recent version provides you with access to the latest features and enhancements. If you have been using a database other than vPostgreSQL, you must migrate it to continue access to its data.

- [Upgrading the vRealize Hyperic Agent](#)

You can upgrade the vRealize Hyperic agent by pushing it from an upgraded vRealize Hyperic server, by performing a manual upgrade that retains the configuration of the original agent.

Upgrading vRealize Hyperic Components and Migrating the Database

Upgrading your existing vRealize Hyperic installation to the most recent version provides you with access to the latest features and enhancements. If you have been using a database other than vPostgreSQL, you must migrate it to continue access to its data.

The installer installs a new version of vRealize Hyperic, configuring the new server instance based on the configuration information from your previous server installation configuration files.

If you use the vRealize Hyperic internal database, the installer creates a new database instance that contains the data from the existing instance. The new instance has an updated schema. The PostgreSQL server is not upgraded to a new version.

If you use an external database, the installer updates the existing instance.

- [Upgrade vRealize Hyperic Server to Version 5.x on a Windows Platform](#)

You upgrade the vRealize Hyperic server on Windows platforms using the upgrade option of the vRealize Hyperic installer.

- [Upgrade vRealize Hyperic Server to Version 5.x on a Unix-Based Platform](#)

You upgrade the vRealize Hyperic server on Unix-based platforms using the upgrade option of the vRealize Hyperic installer.

- [Upgrade the vRealize Hyperic vApp](#)

You can upgrade the version of the vRealize Hyperic vApp to the latest version.

- [Migrate to a vPostgreSQL Database](#)

To use the latest version of vRealize Hyperic you must use a vPostgreSQL database. If you have an earlier version of vRealize Hyperic that supported other database types, you can migrate the database to vPostgreSQL after you have installed the upgraded vRealize Hyperic server.

Upgrade vRealize Hyperic Server to Version 5.x on a Windows Platform

You upgrade the vRealize Hyperic server on Windows platforms using the upgrade option of the vRealize Hyperic installer.

Use this procedure to upgrade the vRealize Hyperic server on Windows platforms.

Prerequisites

- Verify that the current server instance is stopped.
Use the Windows Services Control Panel.
- (Optional) If you use an external vRealize Hyperic database, verify that it is backed up.
- (Optional) It is good practice to archive your existing vRealize Hyperic server directory, so that you can revert to it if necessary.

Procedure

- 1 Run the vRealize Hyperic installer in upgrade mode.
`c:\hyperic\hyperic-hq-installer\setup.bat -upgrade.`
- 2 Acknowledge the VMware license prompt.
- 3 Type the full path to the previous vRealize Hyperic server instance when prompted.
For example, `/opt/hyperic/server-5.0.0`
- 4 Type the full path to the directory under which the new server instance will be installed.
For example, to install the new instance under your existing vRealize Hyperic home directory, type `c:\hyperic\.`

The installer completes the upgrade.

If you use the in-product vRealize Hyperic database, the upgrade process migrates your database schema to the latest edition.

What to do next

- 1 Update the Windows Service with the new version information. For example,
`c:\hyperic\server-5.1.0\bin\hq-server.bat install`
- 2 Start the new server instance. For example type `/opt/hyperic/server-5.1.0/bin/hq-server.sh start.`

Upgrade vRealize Hyperic Server to Version 5.x on a Unix-Based Platform

You upgrade the vRealize Hyperic server on Unix-based platforms using the upgrade option of the vRealize Hyperic installer.

Use this procedure to upgrade the vRealize Hyperic server on Unix-based platforms.

Prerequisites

- Verify that your instance of vRealize Hyperic server does not use an internal database.
You cannot upgrade the vRealize Hyperic server if the server does use an internal database.
- Verify that the current server instance is stopped.
For example, run `/opt/hyperic/server-5.8.4/bin/hq-server.sh stop`.
- Back up your vRealize Hyperic database.
- (Optional) It is good practice to archive your existing vRealize Hyperic server directory, so that you can revert to it if necessary.
For example, `tar -zcvf hq-server-5.8.4-archive.tgz hq-server-5.8.4-EE`.

Procedure

- 1 Download the new version of the installer from the downloads page.
- 2 Run the vRealize Hyperic installer in upgrade mode.
`/opt/hyperic/hyperic-hq-installer/setup.sh -upgrade`.
- 3 Acknowledge the VMware license prompt.
- 4 Type the full path to the previous vRealize Hyperic server instance when prompted.
For example, `/opt/hyperic/server-5.8.4`
- 5 Type the full path to the directory under which the new server instance will be installed.
For example, to install the new instance under your existing vRealize Hyperic home directory, type `/opt/hyperic`.

The installer completes the upgrade.

What to do next

Start the new server instance. For example type `/opt/hyperic/server-5.8.4/bin/hq-server.sh start`.

Upgrade the vRealize Hyperic vApp

You can upgrade the version of the vRealize Hyperic vApp to the latest version.

Prerequisites

- You must be connected to the vRealize Hyperic vApp management console. See [Connect to the vRealize Hyperic vApp Management Console](#).
- Take snapshots of the vRealize Hyperic and vPostgreSQL database to retain as backup.
- Identify the URL link to the new vRealize Hyperic installer TAR or ZIP file. This file can be downloaded from the VMware product downloads page.

Procedure

- 1 On the **Hyperic Server Upgrade** tab of the vRealize Hyperic vApp management console, paste the link to the latest version of the vRealize Hyperic installer in the upgrade text box.
- 2 Click **Upgrade**.

vRealize Hyperic vApp is upgraded to the latest version.

Migrate to a vPostgreSQL Database

To use the latest version of vRealize Hyperic you must use a vPostgreSQL database. If you have an earlier version of vRealize Hyperic that supported other database types, you can migrate the database to vPostgreSQL after you have installed the upgraded vRealize Hyperic server.

This process involves exporting the data from your existing database, then importing it into the vPostgres database.

During the import process, the server is not running.

Prerequisites

Verify that the following prerequisites have been satisfied, before proceeding with the migration process.

- The latest version of vRealize Hyperic server is installed under the same account that you are running the import process.
- The older vRealize Hyperic server is stopped.
- You have superuser privileges to import the database.

By default, the process uses the database credentials defined by the server `.database-user` and server `.database-password` properties in `hq-server.conf`. If the database user account defined in `hq-server.conf` is not a superuser, you must supply superuser account credentials at the the command line during the import phase.

- The latest version of vPostgreSQL database is installed.

- The installer directory is owned by the vRealize Hyperic user. You can use the command `chown -R hyperic /opt/hyperic/hyperic-hqee-installer/installer/` to verify the owner.

Procedure

1 Export the existing database and the server configuration.

- Copy the `hq-migration-5.8.zip` migration package from the `installer/bin` directory in your new vRealize Hyperic installer package to the host server from which you are migrating the database, for example an Hyperic 4.x host.

- Set `JAVA_HOME`.

- Unpack the ZIP file on the host.

The root of the unpacked ZIP, `hq-migration-5.8`, is now referred to as `MigrationHome`.

- To export all configuration and metric data, in a shell run the command `PathToMigrationHome/hq-migrate.sh hq-export - Dhqserver.install.path=PathToServerHome`, where *PathToServerHome* is the full path to the vRealize Hyperic server installation directory, or the path relative to `MigrationHome`.

You can add `-DconfigOnly=true` to the command line to export only the configuration data.

The script reads the `hq-server.conf` file in the older vRealize Hyperic version, connects to its database, exports the database, and creates a tarball with key artifacts and the database dump in `hq-migration-export-HqVersion.tgz`.

By default, the archive is written to the `migration_home/tmp/export-data` import staging directory, or the value of `staging.dir`, if specified.

2 Import the database and the server configuration.

- Copy `hq-migration-export-HqVersion.tgz` to the new server host, or make it available to the machine on which the server is installed.

Expect some latency if you do not copy the tarball to the server host.

- Run the `PathToMigrationHome/hq-migrate.sh hq-import - Dhqserver.install.path=PathToServerHome - Dexport.archive.path=PathToExportArchive` command, ensuring that the command is on a single line.

Enter appropriate values for the *PathToServerHome* and *PathToExportArchive* properties.

The database and server configuration properties are migrated.

Upgrading the vRealize Hyperic Agent

You can upgrade the vRealize Hyperic agent by pushing it from an upgraded vRealize Hyperic server, by performing a manual upgrade that retains the configuration of the original agent.

Push a vRealize Hyperic Agent Bundle from the vRealize Hyperic Server

You can update one or more vRealize Hyperic agents by pushing the new agent bundle to it from the vRealize Hyperic server, using the vRealize Hyperic user interface.

When you update an agent bundle, the configuration settings in the agent's `AgentHome/conf/agent.properties` file are not changed. However, the first time you start an agent that you have updated from version 4.5 or earlier, passwords specified in the file are encrypted.

Prerequisites

The bundle must reside in the `ServerHome/hq-engine/hq-server/webapps/ROOT/WEB-INF/hq-agent-bundles` directory.

Procedure

- 1 On the **Resources** tab, select the server on which the agent bundle resides.
- 2 On the **Views** tab, click **Agent Commands**.
- 3 Select **Upgrade** from the **Select an agent operation to run** menu.
- 4 Select the appropriate bundle from the **Select upgradeable agent bundle** menu.

The bundle includes an update to the JRE. If you do not want to update the JRE, select the bundle that does not include a platform in the file name, for example `agent-version.number.tar.gz`.

- 5 Click **Execute**.

The bundle is copied to the `bundles` directory and self-extracts. On completion of the extraction process, you can see the version information for the upgraded agent on the **Administration > Agents** tab.

Upgrade a vRealize Hyperic Agent Bundle

If you do not want to push the upgrade agent bundle from the vRealize Hyperic server, you can use this process to upgrade the bundle in your vRealize Hyperic agent installation.

When you update an agent bundle, your previous agent configuration is preserved. The `AgentHome/conf/agent.properties` file is not overwritten.

Prerequisites

Procedure

- 1 Copy the `agent-5.x.y-nnn.tgz` or `agent-5.x.y-nnn.zip` agent bundle from `ServerHome/hq-engine/hq-server/webapps/ROOT/WEB-INF/hq-agent-bundles` to `AgentHome/bundles`.
- 2 Unpack the agent bundle.
- 3 Edit the `rollback.properties` file in `AgentHome/conf` to specify the location of the new agent bundle and the bundle it will replace.

Example:

What to do next

Deploy a vRealize Hyperic vApp Using vCloud Director

4

You can create a vRealize Hyperic vApp in your virtual cloud from a vApp template using VMware vCloud Director.

To deploy a vRealize Hyperic vApp using vCloud Director requires you to complete the following procedures, in the order specified below.

Prerequisites

- The vRealize Hyperic server and vRealize Hyperic database OVF files must have been uploaded to a vCloud catalog to which you have access.
- Your browser must be configured appropriately for accessing and using the vCloud web-based console. See *vCloud Director User Guide* for information.
- You must be a competent vCloud Director user.

Procedure

1 Create and Configure the vRealize Hyperic vApp

To deploy a vRealize Hyperic vApp using vCloud Director, you must first create and configure the vApp in vCloud Director.

2 Power on the vRealize Hyperic Server

After you have created your vRealize Hyperic vApp, you must update the vRealize Hyperic server with vRealize Hyperic database location and then power on the vRealize Hyperic server.

Create and Configure the vRealize Hyperic vApp

To deploy a vRealize Hyperic vApp using vCloud Director, you must first create and configure the vApp in vCloud Director.

You cannot configure the vCloud Director server for a user-managed keystore during deployment of the vCloud Director vApp. To configure a user-managed keystore after deployment, see [Configuring SSL Options](#).

Prerequisites

Verify that you have completed the prerequisites described in [Chapter 4 Deploy a vRealize Hyperic vApp Using vCloud Director](#).

Procedure

- 1 Log in to the vCloud Director Web console.
- 2 Click **My Cloud > vApps > Build New vApp** to complete the vApp profile.

Option	Description
Name this vApp	Type a name for the vApp and, optionally, a description.
Runtime UI Text	Select a runtime
Storage Lease UI text	Select a storage lease

- 3 Click **Next**.
- 4 In the Add Virtual Machines pane, select the Hyperic vApp templates.

You find the Hyperic vApp templates in two possible locations:

- **Look In > My organization's catalogs**
- **Look In > Public catalogs**

- 5 Browse to and select the vRealize Hyperic server and database virtual machines:

Option	Action
vRealize Hyperic server	Select Hyperic Server v5.0 Virtual Appliance .
vRealize Hyperic database	Select Hyperic Database v5.0 Virtual Appliance .

- 6 Click **Next**.
- 7 Select the check box to signify your acceptance of the end user license agreement and click **Next**.
- 8 In the Configure Virtual Machines panel, select a virtual datacenter where the vRealize Hyperic vApp will run.
- 9 Accept the default values, or specify other values.
VMware recommends that you assign static IP addresses.
- 10 Click **Next**.
- 11 In the Application panel, configure the user credentials for the PostgreSQL and vRealize Hyperic databases.
Do not enter data in the **What is the address of the vPostgres database** text box.
- 12 Click **Next**.
- 13 In the Networking panel, select the **Always use assigned IP addresses until this vApp or associated networks are deleted** check box and click **Next**.
- 14 Review the summary for the vApp and click **Finish**.

What to do next

Power on the vRealize Hyperic server. See [Power on the vRealize Hyperic Server](#).

Power on the vRealize Hyperic Server

After you have created your vRealize Hyperic vApp, you must update the vRealize Hyperic server with vRealize Hyperic database location and then power on the vRealize Hyperic server.

You must power on the vRealize Hyperic database, then configure the vRealize Hyperic server with the location of the database. You then start the vRealize Hyperic server.

Prerequisites

Create and configure the vRealize Hyperic vApp. See [Create and Configure the vRealize Hyperic vApp](#).

Procedure

- 1 In vCloud Director, go to **vApps > Virtual Machines**, select the vPostgres virtual machine and click **Power On**.
- 2 In vCloud Director, go to **vApps > Virtual Machines** and select the vRealize Hyperic virtual machine.
- 3 Right-click on the virtual machine to display the vApp menu and select **Properties > Custom Properties**.
- 4 Type the IP address of the vRealize Hyperic database in the **What is the address of the vPostgres database** text box.
- 5 Power on the the vRealize Hyperic server.

Post Installation Administration

After you have installed the vRealize Hyperic server and agents, you might need to change some of the configuration options that you first specified. There are various properties that you can change.

This section includes the following topics:

- [Encrypt vRealize Hyperic Agent Property Values](#)
- [Uninstall a vRealize Hyperic Agent](#)

Encrypt vRealize Hyperic Agent Property Values

Although vRealize Hyperic automatically encrypts some property values, such as for a keystore password, you can encrypt agent property values yourself.

You specify the encryption values in the `agent.properties` file. Following a successful initial startup of the vRealize Hyperic agent, credentials are stored in the agent's `/data` directory. Each time the agent is restarted, it looks first in that directory for server connection details. It does not look directly in the `agent.properties` file.

To encrypt values after the agent has been started the first time, you must stop the agent and delete the agent's `/data` directory. You then make the changes in the properties file and restart the agent, as if it is being started for the first time.

Prerequisites

Verify that the vRealize Hyperic agent can access `AgentHome/conf/agent.scu`. Following the encryption of any agent-to-server connection properties, the agent must be able to access this file to start.

Procedure

- 1 Stop the vRealize Hyperic agent.
- 2 Go to `AgentHome` and delete the `/data` directory.
- 3 In the `agent.properties` file, locate `agent.setup.camPword=` and type a password using a plain text value.
- 4 Save the `agent.properties` file.

5 Restart the agent.

The `/data` directory is recreated. The plain text value in the `agent.properties` file is encrypted. If you open this file you will see the encrypted value has replaced the plain text value that you entered.

What to do next

If your agent deployment strategy involves distributing a standard `agent.properties` file to all agents, you must also distribute `agent.scu`. See [Install Multiple vRealize Hyperic Agents Simultaneously](#).

Uninstall a vRealize Hyperic Agent

From time to time you might need to uninstall a vRealize Hyperic agent.

Prerequisites

Determine whether the agent is managed by vRealize Hyperic, or is installed as a Windows service.

Procedure

- ◆ Select the uninstall option that is appropriate to the agent installation environment.

Agent Installation Environment	Action
Agent managed by vRealize Hyperic	Remove the platform for the agent and delete the agent's installation folder.
Agent installed as a Windows service	Run <code>hq-agent.bat remove</code> to remove the Windows service.