Veritas™ HyperScale 2.0 for OpenStack Installation Guide

RHEL

2.0

February 2018

VERITAS™

Veritas HyperScale for OpenStack Installation Guide

Last updated: 2018-02-11

Documentation version: 2.0 Rev 0

Legal Notice

Veritas Technologies LLC
500 E Middlefield Road
Mountain View, CA 94043

http://www.veritas.com

Made in Singapore.

10 9 8 7 6 5 4 3 2 1

Technical Support

Technical Support maintains support centers globally. Technical Support's primary role is to respond to specific queries about product features and functionality. The Technical Support group also creates content for our online Knowledge Base. The Technical Support group works collaboratively with the other functional areas within the company to answer your questions in a timely fashion.

Our support offerings include the following:

- A range of support options that give you the flexibility to select the right amount of service for any size organization
- Telephone and/or Web-based support that provides rapid response and up-to-the-minute information
- Upgrade assurance that delivers software upgrades
- Global support purchased on a regional business hours or 24 hours a day, 7 days a week basis
- Premium service offerings that include Account Management Services

For information about our support offerings, you can visit our website at the following URL:

www.veritas.com/support

All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policy.

Contacting Technical Support

Customers with a current support agreement may access Technical Support information at the following URL:

www.veritas.com/support

Before contacting Technical Support, make sure you have satisfied the system requirements that are listed in your product documentation. Also, you should be at the computer on which the problem occurred, in case it is necessary to replicate the problem.

When you contact Technical Support, please have the following information available:

- Product release level
- Hardware information
- Available memory, disk space, and NIC information

- Operating system
- Version and patch level
- Network topology
- Router, gateway, and IP address information
- Problem description:
  - Error messages and log files
  - Troubleshooting that was performed before contacting Technical Support
  - Recent software configuration changes and network changes

Licensing and registration

If your product requires registration or a license key, access our technical support Web page at the following URL:

www.veritas.com/support

Customer service

Customer service information is available at the following URL:

www.veritas.com/support

Customer Service is available to assist with non-technical questions, such as the following types of issues:

- Questions regarding product licensing or serialization
- Product registration updates, such as address or name changes
- General product information (features, language availability, local dealers)
- Latest information about product updates and upgrades
- Information about upgrade assurance and support contracts
- Advice about technical support options
- Nontechnical presales questions
- Issues that are related to CD-ROMs, DVDs, or manuals

Support agreement resources

If you want to contact us regarding an existing support agreement, please contact the support agreement administration team for your region as follows:

Worldwide (except Japan)	CustomerCare@veritas.com
Japan	CustomerCare_Japan@veritas.com

Documentation feedback

Your feedback is important to us. Suggest improvements or report errors or omissions to the documentation. Include the document title, document version, chapter title, and section title of the text on which you are reporting. Send feedback to:

doc.feedback@veritas.com

You can also see documentation information or ask a question on the Veritas community site:

http://www.veritas.com/community/

Veritas Services and Operations Readiness Tools (SORT)

Veritas Services and Operations Readiness Tools (SORT) is a website that provides information and tools to automate and simplify certain time-consuming administrative tasks. Depending on the product, SORT helps you prepare for installations and upgrades, identify risks in your datacenters, and improve operational efficiency. To see what services and tools SORT provides for your product, see the data sheet:

https://sort.veritas.com/data/support/SORT_Data_Sheet.pdf

Contents

Chapter 1

Planning for your Veritas HyperScale for OpenStack installation

This chapter includes the following topics:

HyperScale product environments

This chapter lists the HyperScale installation requirements for the minimum and preferred production environments.

Table 1-1 HyperScale production environments

Environment	Configuration
Minimum production environment	■ Three controller nodes configured in a high availability mode ■ Two data nodes ■ Three compute nodes
Preferred production environment	■ Three data nodes, out of which at least two will be used as HyperScale data nodes. ■ Ten compute nodes ■ The OpenStack controller is hosted separately from the data nodes and is configured in a high availability mode.

OpenStack Controller HA requirement

If you are installing HyperScale in a Red Hat OpenStack environment, then the OpenStack controller services must be deployed in a high availability (HA) mode.

This is required due to an issue in Red Hat OpenStack due to which the HyperScale's RabbitMQ user and exchanges get deleted.

This issue is tracked using the following bug:

https://bugzilla.redhat.com/show_bug.cgi?id=1516673

Note: This issue is only applicable to Red Hat OpenStack (RHOSP) 12.0 (Pike) deployments and may get addressed in a future release of OpenStack.

Data node requirements

In OpenStack terminology, a HyperScale data node is a Cinder node.

If you install HyperScale in the minimum production environment, the first data node is installed on the same physical node as the OpenStack controller. The installation turns the controller into a HyperScale controller. This controller displays the HyperScale dashboard and enables you to manage and monitor HyperScale compute nodes and data nodes. You can add the second data node to any Cinder node in your environment that meets the installation requirements.

In the preferred production environment, the OpenStack controller is hosted separately from the first data node. Each data node can be installed on any Cinder node. In a preferred production environment, the third Cinder node is used in case

of an irrecoverable data node failure. After the failure, it is added as a new data node in HyperScale deployment, replacing the data node that failed.

Regardless of the production environment you use, before you install HyperScale, the OpenStack deployment must meet the following requirements:

- It must not have any projects or users, except for the projects and the users it creates by default.
- No OpenStack instances should be running.
- The network manager service (`NetworkManager`) must be disabled on the controller.

Data node hardware requirements

Table 1-2 Data node hardware requirements

Hardware	Minimum production environment	Preferred production environment
Memory	64 GB	128 GB
Network interface cards	3 10 Gbps (1 public, 2 private)	2 10 Gbps (1 public, 1 private) 1 40 Gbps (private)
Storage		
Solid-state drive (SSD)	0.8 TB HyperScale supports the following SSDs: ■ Fusion-io Supported kernel version is 4.4.0.21. ■ Intel SATA ■ Intel PCIe For Intel SSDs, the supported kernel version is 3.10.0-327.el7.x86_64. The SSD should not have any partitions and should not be mounted.	1 or 2 TB Same supported drives as the minimum environment.

Table 1-2 Data node hardware requirements *(continued)*

Hardware	Minimum production environment	Preferred production environment
Hard disk drive (HDD)	1 boot disk along with multiple disks amounting to total storage in the range of 20 TB - 40TB The HDDs should not have any partitions. Disks must be labeled `msdos`; otherwise, HyperScale does not acknowledge the devices.	1 boot disk along with multiple disks amounting to total storage of 64 TB or more Same disk labeling requirements as the minimum environment.

Storage notes:

- The storage in the data nodes must be equal or greater than the total storage across all compute nodes.
- RAID6 is recommended in case of disk failures.

Data node software requirements

Table 1-3 Data node software requirements

Software	Requirements (same for minimum and preferred environments)
Operating system	Red Hat Enterprise Linux (RHEL) 7.4 Kernel version: - Linux 3.10.0-* The `multipathd.service` must be running.
OpenStack distribution	Red Hat Enterprise Linux OpenStack (RHOSP) Pike Data nodes must be added to the deployment as a Block Storage Nodes.

Table 1-3 Data node software requirements *(continued)*

Software	Requirements (same for minimum and preferred environments)

OpenStack services	The following services must be enabled:
	■ `cinder-volume`
	■ `nfs-server`
	(required only for HyperScale backup functionality)
	To verify that the services are enabled, log into the controller and enter the following:
	`# systemctl status cinder-volume`
	`# systemctl status nfs-server`
	The status of each service should be `active`.

Network interface	■ The network interface card (NIC), which is configured for the data network, must be enabled. If it is not, enable it with the `ifup` command. ■ You must assign IP addresses to all data interfaces for all nodes. ■ HyperScale supports NIC bonding. Veritas recommends using the active backup mode (mode=1).

Table 1-3 Data node software requirements *(continued)*

Software	Requirements (same for minimum and preferred environments)
Packages	The following packages are required on the data node: ■ `coreutils` ■ `ipcalc` ■ `openstack-utils` ■ `java` ■ `openjdk-8-jre-headless` Ensure that `/usr/bin/java` is created after installing java packages. ■ `libcurl4-openssl-dev` ■ `lvm2` ■ `multipath-tools` ■ `python-amqp` ■ `python-anyjson` ■ `python-kazoo` ■ `nfs-kernel-server` ■ `python-kombu` ■ `python-sqlalchemy` If the controller is also deployed on the data node, the following are required in addition to the packages mentioned earlier: ■ `bc` ■ `crudini` ■ `dpkg-dev` ■ `default-jre` ■ `mysql-client` ■ `python-mysqldb` ■ `python-pycurl` ■ `python-openstackclient (3.8.1-6)` ■ `rabbitmq-server` ■ `software-properties-common` ■ `sshpass` ■ `zookeeper`

Compute node requirements

Compute node hardware requirements

Table 1-4 Compute node hardware requirements

Hardware	Minimum production environment	Preferred production environment
Memory	64 GB	128 GB
Network interface card	3 10 Gbps base-T NIC (1 public, 2 private)	4 10 Gbps base-T NIC (2 public, 2 private). This configuration does not assume port aggregation.
Solid-state drive (SSD)	800 GB HyperScale supports the following SSDs: ■ Fusion-io Supported kernel version is 4.4.0.21. ■ Intel SATA ■ Intel PCIe For Intel SSDs, the supported kernel version is 3.10.0-327.el7.x86_64. The SSD should not have any partitions and should not be mounted.	1 TB Same supported drives as the minimum environment.
Hard disk drive (HDD)	1 boot disk along with multiple disks amounting to total storage in the range of 20 TB - 40TB (No SAN) Disks must be labeled `msdos`; otherwise, HyperScale does not acknowledge the devices.	1 boot disk along with multiple disks amounting to total storage of 64 TB or more Same disk labeling requirements as the minimum environment.

Compute node software requirements

Table 1-5 Compute node software requirements

Software	Requirements (same for both minimum and preferred environments)
Operating system	Red Hat Enterprise Linux (RHEL) 7.4 Kernel version: ■ Linux 3.10.0-* OpenStack compute services are installed and running spare compute and memory to run HyperScale storage service.
OpenStack distribution	Red Hat Enterprise Linux OpenStack (RHOSP) Pike Data nodes must be added to the deployment as a Block Storage Nodes.
Network interface	■ The network interface must be enabled. If it is not, enable it with the `ifup` command. ■ You must assign IP addresses to all data interfaces for all nodes. ■ HyperScale supports NIC bonding. Veritas recommends using the active backup mode (mode=1).
Packages	The following packages are required on the compute node: ■ `bc` ■ `cgroup-bin` ■ `collectd (version 5.7 or later)` ■ `crudini` ■ `dnsmasq-base` ■ `expect` ■ `librrd4` ■ `libltdl7` ■ `lvm2` ■ `python 2.6 or later` ■ `python-ceilometerclient` ■ `python2-pip` ■ `pip` ■ `xfsprogs`
Other software	XFS journaling file system installed. The `expect` package must be installed.

Firewall and network ports requirements

If you have configured a firewall, then ensure that the firewall settings allow access to the services and ports used by HyperScale. Enable both inbound and outbound communication for these ports and services.

Note: Ports are not user-configurable. Ensure that they are not already in use.

Table 1-6 HyperScale ports and services

Port / Protocol	Source / Destination	Description
8753 / HTTP	HyperScale controller / HyperScale compute and data node	Used by the REST API service running on the controller Type of access: Inbound
42181, 2181, 2888, 3888 / TCP	HyperScale controller / HyperScale compute and data node	Used by the ZooKeeper service Type of access: Bi-directional
15672 / HTTP 5673 / TCP	HyperScale controller / HyperScale compute and data node	Used by the RabbitMQ server Type of access: Bi-directional
29982 / HTTP 2049 / TCP	HyperScale controller / HyperScale compute and data node	Used by the HyperScale controller services
9898 / TCP	HyperScale compute node	Used by the `hyperscale-mq-storage` service for I/O redirection from the storage driver to the HyperScale daemon on the compute node. This is required for live migration and to ensure resiliency in case of storage failures. Type of access: Bi-directional
9999 / TCP	HyperScale compute node / HyperScale data node	Used by `hyperscale-mq-storage` service for: ■ I/O reflection between compute nodes. ■ Snapshot data transfer between the HyperScale daemon on the compute node and the HyperScale dmld service on the data node. Type of access: Bi-directional

Table 1-6 HyperScale ports and services *(continued)*

Port / Protocol	Source / Destination	Description
11786 / HTTP	HyperScale data node / HyperScale controller	Used by the `hyperscale-bud` service (Backup Daemon) for data transfer Type of access: Bi-directional
11787 / HTTP	HyperScale data node / HyperScale controller	Used by the `hyperscale-bcd` service (Backup Control Daemon) for backup APIs Type of access: Bi-directional
8334 / HTTPS, HTTP	HyperScale ESP node / HyperScale data node	Used by the External Storage Provider (ESP) service for backups Type of access: Bi-directional
21786 / HTTPS	HyperScale data node / HyperScale ESP node	Used by the `hyperscale-bud` service (Backup Daemon) for data transfer Type of access: Bi-directional
21787 / HTTPS	HyperScale data node / HyperScale ESP node	Used by the `hyperscale-bcd` service (Backup Control Daemon) for backup APIs Type of access: Bi-directional
25826 / HTTP	HyperScale controller / HyperScale compute node	InfluxDB port used by HyperScale on the controller Type of access: Inbound
3000, 8041 / TCP	HyperScale controller / OpenStack Horizon	Required for accessing the Grafana user interface and the Gnocchi database on the controller Type of access: Bi-directional
17919 / HTTP	HyperScale controller	Used for HyperScale license management on the HyperScale controller. Type of access: Bi-directional

Iptables configuration

If Iptables is configured and enabled, you need to add custom rules to provide access to HyperScale services on these ports.

To make the custom rules persistent, modify the Iptables firewall rules by adding the following lines to `/etc/sysconfigd/iptables` just before the commit statement:

For the controller node

```
-I INPUT -p tcp -m multiport --dports 3000 -m state --state NEW,ESTABLISHED
-m comment --comment "HyperScale grafana Service" -j ACCEPT

-I INPUT -p tcp -m multiport --dports 8753,9999,42181,9898,15672,
5673,29982,21787,11787,11786,2049 -m state --state NEW,ESTABLISHED
-m comment --comment "HyperScale Services ipv4" -j ACCEPT

-I INPUT -p tcp -m multiport --dports 2181,2888,3888 -m state
--state NEW,ESTABLISHED -m comment --comment "140 Zookeeper Services ipv4"
-j ACCEPT
```

For the data node

```
-I INPUT -p tcp -m multiport --dports 8753,9999,42181,9898,15672,
5673,29982,21787,11787,11786,2049 -m state --state NEW,ESTABLISHED
-m comment --comment "HyperScale Services ipv4" -j ACCEPT

-I INPUT -p tcp -m multiport --dports 2181,2888,3888 -m state
--state NEW,ESTABLISHED -m comment --comment "140 Zookeeper Services ipv4"
-j ACCEPT
```

For the compute node

```
-I INPUT -p tcp -m multiport --dports 8753,9999,42181,9898,15672,
5673,29982,21787,11787,11786,2049 -m state --state NEW,ESTABLISHED
-m comment --comment "HyperScale Services ipv4" -j ACCEPT

-I INPUT -p tcp -m multiport --dports 2181,2888,3888 -m state
--state NEW,ESTABLISHED -m comment --comment "140 Zookeeper Services ipv4"
-j ACCEPT
```

Restart Iptables on all nodes after making the updates:

```
# sudo systemctl restart iptables
```

HyperScale sudo user requirements

Create a user account by the name `hyperscale` and give it password-less sudo access.

This sudo user account is required for the following operations:

- install and uninstall HyperScale controller

- add or remove data nodes from the HyperScale configuration
- enable or disable HyperScale on the compute nodes
- perform backup and restore operations

Where you create the `hyperscale` sudo user account depends on how you plan to deploy HyperScale in your OpenStack environment. In general, it is required on all the data nodes and compute nodes and on the External Storage Provider (ESP) or backup proxy node.

Table 1-7 Where to create hyperscale sudo user

Deployment scenario	Create sudo user on...
If installing HyperScale in a Red Hat OpenStack (RHOSP) environment	■ Data nodes (typically OpenStack cinder machines) ■ Compute nodes
For backup and restore	External Storage Provider (ESP) or backup proxy node

To create a hyperscale user and grant password-less sudo access

1. Log on to the system.
2. Create a user account by the name `hyperscale` and add the user to a group named `hyperscale`.

   ```
   # sudo useradd -m -c hyperscale -U hyperscale
   ```

3. Set a password for the new user account.

   ```
   # sudo passwd hyperscale
   ```

   Set and confirm the password.

4. Edit the `/etc/sudoers` file using the `visudo` command as follows:
   - `# sudo visudo`
   - Add the following entry to the sudoers file:
     `# hyperscale ALL=(ALL) NOPASSWD: ALL`
   - Save the changes to the file.
5. Edit the `/etc/sudoers` file using the `visudo` command as follows:
   - `# sudo visudo`
   - Add the following entry to the sudoers file:
     `Defaults:hyperscale !requiretty`
   - Save the changes to the file.

6 Modify the ssh configuration file (`/etc/ssh/sshd_config`) to enable password-based authentication for the `hyperscale` user account.

- Open the sshd_config file and add the following lines:

  ```
  PasswordAuthentication yes
  ```

- Save and close the file.
- Restart sshd.

  ```
  # sudo systemctl restart sshd
  ```

7 Repeat these steps, as applicable, on all the remaining systems where you wish to deploy HyperScale.

External storage provider requirements

HyperScale's external storage provider lets infrastructure administrators create and export HyperScale instance snapshots to an external storage device. The exported snapshot can then be used to launch a new instance if the original becomes corrupted, is accidentally deleted, or another data loss or disaster recovery situation occurs.

Making sure that data is securely and efficiently transferred between the HyperScale nodes (compute and data nodes) and external storage provider requires some advance planning.

- You should connect the external storage provider to the same management subnet and public network as the rest of the HyperScale deployment.
- If the external storage provider is located in the same rack as the other HyperScale nodes, connect its interfaces to the appropriate ports of the middle of the rack switch.
  Each of the subnets should have its own virtual LAN (VLAN).
- The same holds true even if the external storage provider is located in a different rack. However, the middle of the rack switches must be properly connected and trunked to enable intra-VLAN traffic between the two racks.
- Before you integrate an external storage provider with HyperScale, the provider must be installed on a separate node that is not part of the HyperScale environment.

The node must be a server-grade system with high I/O throughput. In addition, it must meet the following requirements:

Table 1-8 Hardware requirements

Hardware	Requirement
RAM	32 GB minimum, 64 GB or higher recommended
Virtual CPUs	4 cores minimum, 8 cores recommended
Network speed	1 Gbps minimum, 10 Gbps recommended
Hardware type	Physical machine
Network interface cards (NICs)	2 required ■ A 1 Gbps management NIC. Configure this NIC to be on the same management subnet as the compute node and data node. ■ A backup network NIC (10 Gbps or higher) to access the node, check logs, and perform troubleshooting. The backup network can either be on a separate subnet of its own or can be configured to use the public NIC.
Physical storage	Enough physical storage to maintain instance snapshots. HyperScale uses this space while performing restore operations. HyperScale maintains a full copy of instance vDisks and metadata if it is exported to the external storage provider node. For example, if you export five instances that have a total of 5 TB of data written to their vDisks, the external storage provider node requires 5 TB of storage.
Ports	Port 8334 is open or the firewall on the external storage provider node is disabled. Verify this before you try to register the external storage provider from the HyperScale user interface.

Note: When the external storage provider imports or exports a snapshot, it communicates with the primary and secondary data nodes using the management subnet and the backup network. Therefore, the external storage provider node should either be in the same physical rack as the HyperScale nodes, or, if it is outside the rack, it should be able to communicate to other HyperScale nodes using the management and backup networks.

Figure 1-1 Controller node and data node communication with the external storage provider node

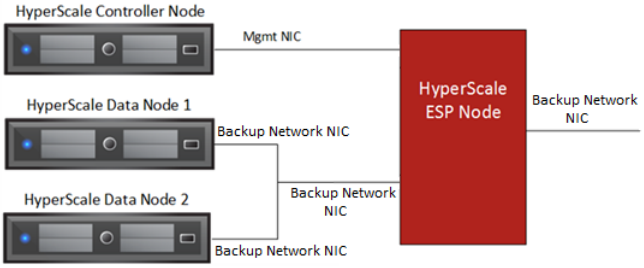


Table 1-9 Software requirements

Software	Requirement
Operating system	Red Hat Enterprise Linux (RHEL) 7.2 Infrastructure Edition (minimum)
Web framework	Flask 0.10.1 or later
Python	2.7 or later
Packages	The following packages are required on the External Storage Provider node: ■ `rpcbind` ■ `nfs-utils`

NetBackup Media Server requirements

The NetBackup administrator must set up a dedicated Media Server in the production environment. The Media Server is used as a backup proxy.

Table 1-10 NetBackup Media Server (backup proxy) requirements

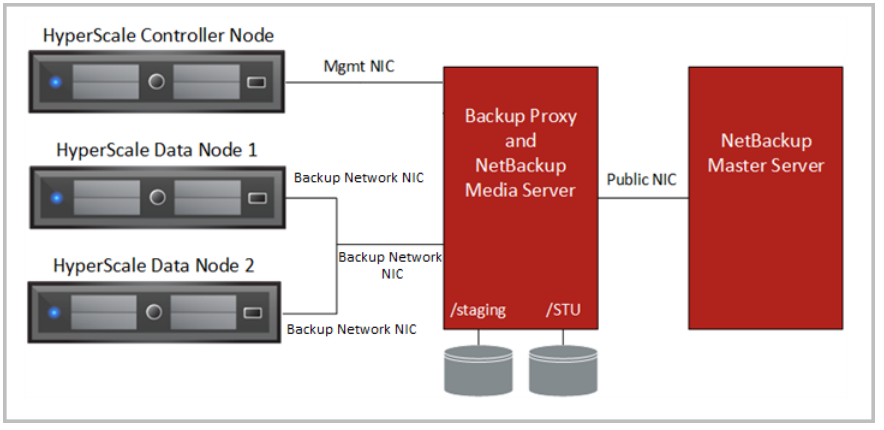
Software / Hardware	Requirement
RAM	32 GB minimum, 64 GB or higher recommended
Virtual CPUs	4 cores minimum, 8 cores recommended
Network speed	1 Gbps minimum, 10 Gbps recommended
Hardware type	Physical machine

Table 1-10 NetBackup Media Server (backup proxy) requirements *(continued)*

Software / Hardware	Requirement
Network interface cards (NICs)	2 required
Physical storage	Enough physical storage to maintain instance snapshots. HyperScale maintains a full copy of an instance's vDisks and metadata when it is exported to the external storage provider node. For example, if you export five instances that have a total of 5 TB of data written to their vDisks, the external storage provider node requires 5 TB of storage.
Network	Port 8334 is open or the firewall on the backup proxy node is disabled. Verify this before you try to register NetBackup from the HyperScale user interface.
Packages	The following packages are required on the backup proxy node: ▪ `rpcbind` ▪ `nfs-utils`

When the NetBackup administrator completes the configuration, HyperScale, the NetBackup Media Server (backup proxy), and NetBackup Master Server should have the following communication path.

Figure 1-2 Communications between the HyperScale controller and data nodes and NetBackup Media and Master servers



In addition, the NetBackup administrator must configure the directory path that HyperScale uses for restore operations; this is known as the staging location. This location needs to be on a separate set of disks than that of the storage unit (STU) disks (in case of a hard-disk backed STU).

NetBackup integration requirements

Note: This HyperScale release is only qualified with NetBackup Version 8.0.

To integrate NetBackup with HyperScale, the NetBackup administrator must verify the following:

- The NetBackup Media Server and Master Server must be installed. When you install the servers, use the fully qualified domain names. Also use the fully qualified domain name when you register the Media Server with the Master Server.
- The Media Server must be running Red Hat Enterprise Linux (RHEL) 7.2. The "Minimal Install" is not supported; however, the RHEL software "Infrastructure Server" and above are supported.
- The firewall on the backup proxy node is disabled, or port 8443 is open. (This must be verified before you try to register NetBackup from the HyperScale user interface.)
- The date, time, and zone on the backup proxy node must be in sync with the date, time, and zone on the Master Server. If the dates and times are not sync, backups still occur, but they are not displayed on the HyperScale user interface.

After this information is verified, the NetBackup administrator must communicate the following information to the HyperScale infrastructure administrator:

- The Media Server fully qualified domain name.
- The Media Server password.
- The OpenStack password.
- The staging location which is the directory path that HyperScale uses for restore operations. This location needs to be on a separate set of disks than that of the storage unit (STU) disks (in case of a hard-disk backed STU).

OpenStack changes to support HyperScale live migration

To perform live migrations in HyperScale, you must make the following changes to the OpenStack configuration.

Table 1-11 OpenStack configuration changes to support HyperScale live migration

For this item ...	Make this change ...
`nova.conf`	`rpc_response_timeout = 180` (The default is 60.)
`cinder.conf`	`rpc_response_timeout = 180` (The default is 60.)
MySQL	`Max Connections = 2048`

Table 1-11 OpenStack configuration changes to support HyperScale live migration *(continued)*

For this item ...	Make this change ...

RabbitMQ configuration file	Add the following lines:
	`[binary,`
	`{packet, raw}`
	`,`
	`{reuseaddr, true}`
	`,`
	`{backlog, 128}`
	`,`
	`{nodelay, true}`
	`,{linger,{true,0}},`
	`{exit_on_close, false}`
	`,`
	`{sndbuf,32768}`
	`,`
	`{recbuf,32768}`
	`,`
	`{keepalive, true}`
	`]}`

Note: If SELinux is enabled, you need to perform certain additional steps for OpenStack live migration to work. Refer to the OpenStack documentation for more information.

Chapter 2

Installing Veritas HyperScale for OpenStack

This chapter includes the following topics:

About HyperScale deployment in an OpenStack environment

At a high level, deploying HyperScale in an OpenStack environment involves the following tasks:

- Installing the HyperScale controller components
- Adding data nodes to the HyperScale configuration
- Enabling HyperScale on the compute nodes

For deployingHyperScale in a Red Hat OpenStack (RHOSP) environment, you first download the HyperScale packages and configuration files on the undercloud controller node, then export the HyperScale installer packages to the overcloud controller, and then install the HyperScale components using the standard OpenStack deployment commands.

See “Deploying HyperScale in a Red Hat OpenStack environment” on page 29.

Deploying HyperScale in a Red Hat OpenStack environment

HyperScale supports deployment in a Red Hat OpenStack environment that is set up using TripleO, which uses OpenStack itself to install and configure OpenStack. It involves two clouds-Undercloud and Overcloud. The Undercloud is a minimal OpenStack configuration that is used to deploy a production-grade OpenStack configuration, called as the Overcloud, that runs the actual workloads.

Refer to the Red Hat documentation for more information:

https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/12/html/partner_integration/architecture

Installing HyperScale in a RHOSP environment involves exporting HyperScale packages as puppet modules from the OpenStack Undercloud and then using OpenStack commands to install HyperScale components on the nodes configured in the OpenStack Overcloud. HyperScale also provides a set of configuration files that contain HyperScale-specific configuration parameters required for the installation.

You can install HyperScale using the automated installer script.

Refer to the following:

See “Before deploying HyperScale in a Red Hat environment” on page 29.

See “Installing HyperScale in a Red Hat OpenStack environment” on page 30.

Before deploying HyperScale in a Red Hat environment

Verify the following in your Red Hat OpenStack (RHOSP) environment:

- Verify that your RHOSP configuration is up and running.
  Refer to the Red Hat documentation for more information:
  https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/12/html/director_installation_and_usage/
- Ensure that all the nodes are registered with Red Hat and the `yum` repository is configured.
- Ensure that you review the system requirements for setting up a HyperScale environment.
  See “*Planning for your Veritas HyperScale for OpenStack installation*” on page 9.
- Verify that the system clocks on all the nodes are synchronized and belong to the same time zone.

Refer to the following OpenStack documentation:
https://docs.openstack.org/ocata/install-guide-rdo/environment-ntp-controller.html

- Ensure that the Iptables firewall rules are set to allow access to HyperScale services on all the required ports.
  See "Iptables configuration" on page 18.
- Have the following ready. These details are needed during the installation:
  - Management IP addresses of all the OpenStack controllers in the Overcloud.
    Run the following command on the Undercloud controller:
    ```
    # nova list
    ```
    The command output displays a list of nodes in the Overcloud along with other details. Make a note of all the controllers and their IP addresses.
  - Management IP address of the active OpenStack controller in the Overcloud.
    An active node is the node that is currently functioning as the OpenStack controller, when there are multiple controllers deployed in a high availability mode. Out of all the controller management IP addresses you noted down earlier, one of those IP addresses belongs to the active controller node.
    Run the following command on any one of the Overcloud controllers:
    ```
    # ip addr | grep $(hiera -c /etc/puppet/hiera.yaml
    controller_virtual_ip) | wc -l
    ```
    The command output displays a 1 or a 0, depending on whether or not it is the active controller.
    If the command output is 0, then run the command again on the next controller node. Repeat this on all the controllers in the Overcloud, one node at a time, until you get the output as 1. If the command output displays a 1, then that's the active controller.
  - OpenStack container names for `Horizon`, `RabbitMQ server`, and `MySQL`.
    Run the following command on the Overcloud controller:
    ```
    # sudo docker ps -a
    ```
  - Passwords for MySQL and RabbitMQ.

Installing HyperScale in a Red Hat OpenStack environment

HyperScale provides a python script that you can use to install HyperScale in your Red Hat OpenStack (RHOSP) environment. The script ties together all the tasks involved in deploying HyperScale components and automates the entire installation process.

The installer script performs the following tasks:

- verifies the systems for all the required packages, and installs them where necessary
- prepares the environment files with Hyperscale-specific parameters required for the installation
- copies the HyperScale installer rpm package files to the controller in the Undercloud
- exports the installer files as a Swift image to all the nodes in the Overcloud
- deploys HyperScale controller and other components on all the nodes in the Overcloud

To deploy HyperScale in a RHOSP environment

1. Log on to the Undercloud controller node using a stack user.
2. Source the `stackrc` file to set the environment variables.

   ```
   # source stackrc
   ```

3. Download the following HyperScale installation file to the stack user's home directory:

   ```
   Veritas_HyperScale_for_OpenStack_2.0.0.000.IE.tar.gz
   ```

   This tar.gz file contains the rpm package files that are required to install HyperScale. Additionally, the tar.gz files includes the installer script, the puppet modules, and the environment configuration files that contain HyperScale-specific parameters required for the installation.

   Note: The actual file name might vary depending on the product release version.

4. Extract the installation file in the stack user's home directory on the Undercloud controller.

   ```
   # tar -xvf Veritas_HyperScale_for_OpenStack_2.0.0.000.IE.tar.gz
   ```

   You should see the following contents extracted:

   ```
   osp12
    |- /lib
    |- /puppet_modules
    |- hs_install.conf
    |- README
    |- hs_install.py
    |- os_config.yaml
   ```

5 Edit the HyperScale configuration file, `hs_install.conf`, located inside the `osp12` directory that is extracted from the HyperScale installation tar.gz file.

Specify appropriate values for the following parameters depending on your configuration:

Parameter name	Description
horizon_container_name	The container name for OpenStack Horizon. Default value used is *horizon*.
rabbitmq_container_name	The container name for RabbitMQ server on the master controller. A master controller is the active controller whose IP is specified in the `master_controller_ip` parameter. Default value used is *rabbitmq-bundle-docker-0*.
mysql_container_name	The container name for OpenStack MySQL database on the master controller. A master controller is the active controller whose IP is specified in the `master_controller_ip` parameter. Default value used is *galera-bundle-docker-0*.
mysql_password	The password for the MySQL database. Default value used is *elacsrepyh*.
rabbitmq_passwd	The password for the RabbitMQ server. Default value used is *elacsrepyh*.
master_controller_ip	The management IP address of the active controller in the Overcloud. The active controller node is the node that is currently functioning as the OpenStack controller in an environment where there are multiple controller nodes deployed in a high availability mode. There is no default value used and you must specify a value before you proceed to the next step.

Parameter name	Description
controller_ips	A comma-separated list (without spaces) of the management IP addresses of all the controllers in the Overcloud. Run the following command on the Undercloud controller to get a list of all the controllers and their IP addresses: `# nova list` There is no default value used and you must specify a value before you proceed to the next step. **Note:** If the OpenStack controller is not deployed in a high availability mode, there will only be a single controller node. Therefore the single controller node will be the active controller node, and in such a case, this parameter value will be the same as that specified for the `master_controller_ip` parameter earlier.

Specify the parameter values and then save your changes to the file. If no values are specified, the default values take effect, where applicable.

6 Launch the automated installer script file, `hs_install.py`, located inside the `osp12` directory that is extracted from the HyperScale installation tar.gz file.

Run the following command on the Overcloud controller to begin the HyperScale installation:

```
# python hs_install.py
```

The command prompt displays multiple messages that indicate the progress of the operation. Depending on the environment, this command may take some time to complete.

The following message confirms that the process has completed successfully:

```
Stack overcloud UPDATE_COMPLETE
```

You can also check the status of the operation by running the following command on the Undercloud controller:

```
# openstack stack list
```

7 Log on to the Overcloud controller node.

8 Switch to the root user.

```
# sudo -i
```

9 Copy the OpenStack rc file, `overcloudrc.v3`, from the Undercloud controller to the Overcloud controller and then source it to set the environment variables.

```
# source overcloudrc.v3
```

Perform this step on each controller from where you wish to run HyperScale commands.

The OpenStack rc file, `overcloudrc.v3`, is created automatically when you deploy the OpenStack Undercloud and Overcloud environment.

10 Verify that the HyperScale controller is deployed and running in the Overcloud.

Run the following command on the Overcloud controller:

```
# hyperscale nodes
```

The command output confirms that the controller is up and running and may look similar to the following:

```
+------------+------------------------+---------------+-------------+--------+---------------+
| hostid     | hostname               | ip_address    | personality | status |  prod_version |
+------------+------------------------+---------------+-------------+--------+---------------+
| {00015254} | overcloud-controller-0 | 192.168.24.10 | controller  | up     | 2.0.0.000     |
+------------+------------------------+---------------+-------------+--------+---------------+
```

The actual command output may display additional details. The data displayed here is truncated to fit the view.

This completes the HyperScale installation in the Overcloud. You can now proceed to adding HyperScale data nodes to the configuration.

See “Adding a data node” on page 34.

Adding a data node

Adding a data node is the first task you perform after installing HyperScale. You can add a data node from the HyperScale graphical user interface (GUI) or from the command line.

About adding data nodes

HyperScale supports up to two data nodes. In the minimum production environment, the first data node you add must be on the same node as the HyperScale controller. In the preferred production environment, both data nodes are hosted on different physical nodes.

Data nodes are used to store point in time snapshots of virtual machine volumes. Every 15 minutes, the compute nodes write new data to the data plane. This operation is known as Episodic Data Synchronization.

In addition, if there are two data nodes in the environment, for the VRTSSilver flavor and above, one data node is the primary data node and the other is the secondary data node. In this configuration, data is reflected from the primary data node to the secondary data node. If one of the data nodes fail, all requests are served from the other data node.

For virtual machines with the VRTSBronze flavor, there is no reflection at the data node level. If the data node that is associated with a Bronze virtual machine fails, you cannot access all the volume versions.

Before adding a data node

Check for the following:

- Ensure that you have installed HyperScale on the OpenStack controller
- Make sure that the disks you intend to use for data node storage have the `msdos` label type.
- Keep the following information ready. You will need to specify these details while adding a data node.
  - The IP address of the Cinder node on which you add the data node.
    In the minimum production environment, this node is the controller system.
  - The hyperscale sudo user account password.
  - The OpenStack admin password.

Adding a data node from the HyperScale GUI

To add a data node from the graphic user interface

1. Launch a web browser and type the URL that you noted during the HyperScale installation.

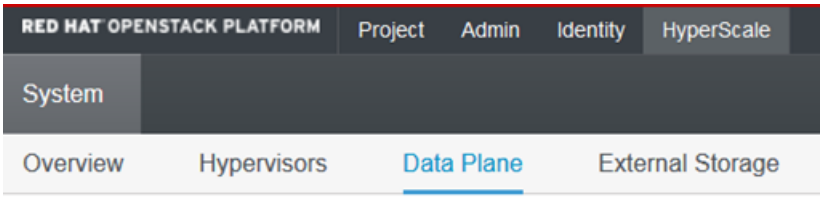
   The URL format is as follows:

   `http://<ControllerNameorIP>/horizon/storage`

   Here, `<ControllerNameorIP>` is the fully qualified domain name or public IP address of the OpenStack controller node.

2. Specify the OpenStack admin user name and password on the login page and click **Connect**.

3 Navigate to **HyperScale > System > Data Plane**.



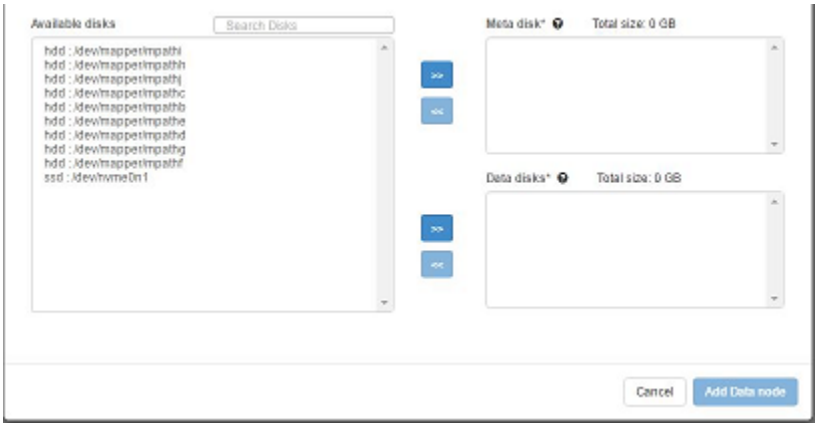
4 On the **Data Nodes** page, click **Add Data Node**.

5 Complete the **Add Data Node** dialog box.

6 Click **Discover**.

HyperScale discovers the network interfaces and storage for the hypervisor that is associated with the data node. The **Add Data Node** dialog box expands.

7 Click **Add Data Node**. The dialog box expands to display more fields.



8 Use the **Data plane interface** drop-down list to select the network interface. Make sure that the network interface is enabled and that an IP address is assigned to it. If the network interface is not enabled, the drop-down list is empty.

Note: Do not select the OpenStack management IP address.

---

9 Assign storage for the meta disk and data disks. Select storage from the **Available disks** area and use the right arrow icons to move the storage to the **Meta disk** area and **Data disks** area.

At a minimum you need two hard disk drives (HDDs): one for the meta disk and one for the data disk.

It is recommended that you choose an SSD for the meta disk. If you have two or more SSDs available, HyperScale gives you to the option to mirror metadata. That way, if one SSD permanently fails, the data node continues to work and it does not have to switch to the other data node. If you select multiple SSDs, mirrored volumes are created on them.

Use all the remaining storage for the data disks. For optimal data disk performance, you should have 40 spinning disks.

10 Click **Add Data Node**.

The **Data Nodes** table is updated to reflect the new data node.

This completes the data node addition. The next step is to add additional data nodes and then enable HyperScale on the compute nodes.

See "Adding additional data nodes for resiliency" on page 39.

See "Enabling HyperScale on a compute node" on page 39.

Adding a data node using HyperScale commands

Perform the following steps on the controller node to add a data node using HyperScale commands.

To add a data node from the command line

1. On the controller node, source the OpenStack RC file:

   ```
   # source keystonerc_admin
   ```

   This operation sets the required environment variables.

2. Add the data node using the HyperScale `dn-add` command.

   Use the following syntax:

   ```
   # hyperscale dn-add [--data_disks <data_disks>]
   [--meta_disk <meta_disk>] <datanode-mgmt-ip> <datanode-pwd>
   <openstack-pwd> <dndata-itface> <data_cidr_range>
   <dnpublic-itface> <public_cidr_range>
   ```

   Here,

   - *<datanode-mgmt-ip>* is the IP address associated with the management network interface on the data node
   - *<datanode-pwd>* is the password of the hyperscale sudo user account that you created before installing HyperScale
   - *<openstack-pwd>* is the OpenStack admin password
   - *<dndata-itface>* is the name of the interface associated with the data network
   - *<dnpublic-itface>* is the name of the public interface on the data node

   Use commas to specify multiple disks.

   Example:

   ```
   # hyperscale dn-add --data_disks /dev/mapper/mpathb,/dev/mapper/mpathc
   --meta_disk /dev/fioa
   172.202.202.1 sudoroot123 openstackadmin123
   eth5 172.202.202.0/24 ens4 <public IP range>
   ```

3 Monitor the add operation as it progresses.

Check the following log file on the controller node:

`/var/opt/VRTSofcore/log/<datanode_mgmtip>_config.log`

4 Verify that the data node was created. The nodes table has an entry for the new data node and the data node's status appears as `up`.

Run the following command:

```
# hyperscale nodes
```

The following abbreviated output shows the hostid and the status of the new data node:

```
+--------------------------------------+ ...  +--------+ ...
| hostid                               | ...  | status | ...
+--------------------------------------+ ...  +--------+ ...
{0001a036-9f33-3ea0-0000-000000000000}   | ...  | up     | ...
```

If an error occurs when you add a data node, information is written to the `/var/opt/VRTSofcore/log/<datanode_mgmtip>_config.log` file on the controller node.

This completes the data node addition. The next step is to add additional data nodes and then enable HyperScale on the compute nodes.

See "Adding additional data nodes for resiliency" on page 39.

See "Enabling HyperScale on a compute node" on page 39.

Adding additional data nodes for resiliency

Add one or more additional data nodes to your environment to ensure that the data written to the primary data node is resilient. This ensures that if an error occurs on the first data node, data is available on the second data node, and there is no data loss.

See "Adding a data node" on page 34.

Enabling HyperScale on a compute node

You can enable HyperScale on a compute node from the graphical user interface (GUI) or from the command line.

Before you proceed

The following information is required to enable HyperScale on a compute node:

- The hyperscale sudo user account password
- The OpenStack admin password

About enabling HyperScale on compute nodes

The HyperScale compute node hosts the workloads. Although you can run all services on a single compute node, you can add additional compute nodes to scale out and expand your HyperScale cloud environment.

To make sure that the data on your compute nodes is resilient, you must enable HyperScale on enough compute nodes to account for the reflection factor (the number of nodes to which the data is written) plus one. You need the additional compute node in case one of the reflection targets goes down. The reflection factor is based on the HyperScale flavor you assign to the virtual machines.

The following table shows the minimum number of compute nodes you need for each virtual machine flavor.

Table 2-1 Minimum number of compute nodes to provide resiliency

Flavor	Reflection factor	Compute nodes (including the source compute node)
VRTSSilver	1	3
VRTSGold	2	4

If an error occurs when you enable HyperScale on a compute node, information is written to the following log file on the controller node:

`/var/opt/VRTSofcore/logs/<computenode_mgmtip>_config.log`

Enabling HyperScale on a compute node using the GUI

To enable HyperScale from the GUI

1. Launch a web browser and type the HyperScale dashboard URL.

   The URL format is as follows:

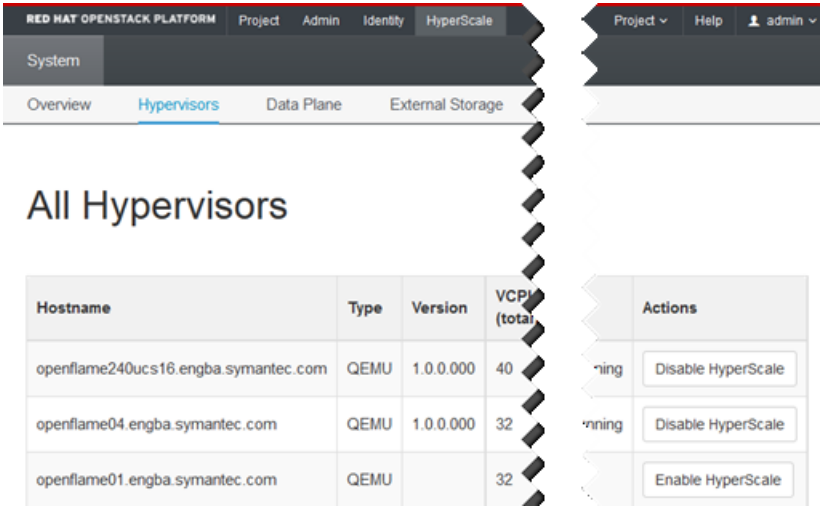
   `http://<ControllerNameorIP>/horizon/storage`

   Here, `<ControllerNameorIP>` is the fully qualified domain name or the public IP address of the controller node.

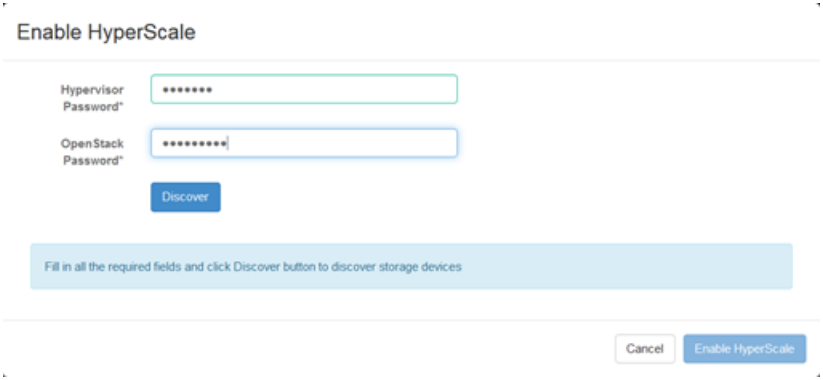
2. Specify the OpenStack admin user name and password on the login page and click **Connect**.

3 Navigate to **HyperScale > System > Hypervisors**.

4 In the **All Hypervisors** table, locate the hypervisor on which you want to enable HyperScale and on the right side of the table, under the **Actions** column, click **Enable HyperScale**.

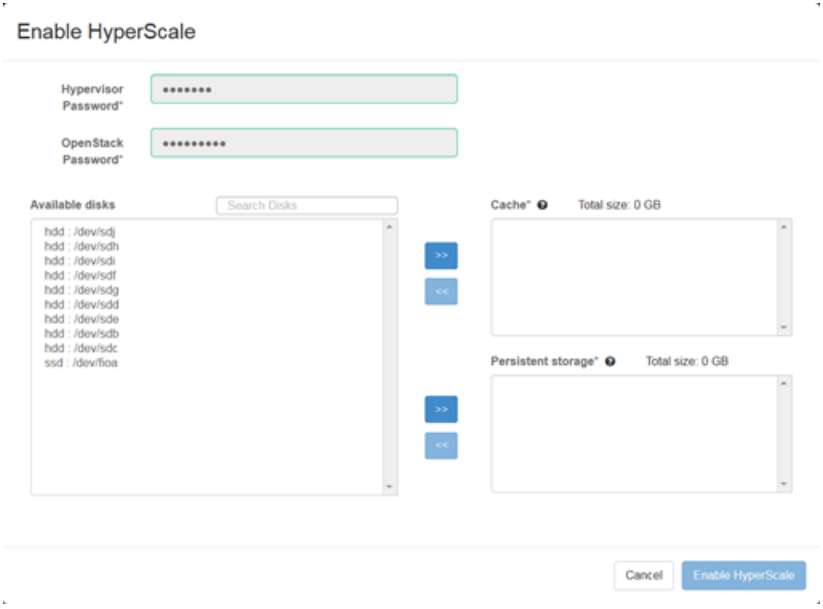


5 Provide the required details on the **Enable HyperScale** dialog box.



6 Click **Discover**.

7 On the expanded Enable HyperScale dialog box, use the Available disks list to select the disks for cache and persistent storage. If possible, you should use solid-state drives (SSDs) for the cache. Click a disk name in the list and use the right arrows (**>>**) to move the disk name to the appropriate category.



8 Click **Enable HyperScale**.

9 Verify that HyperScale is enabled.

On the **All Hypervisors** page, the **HyperScale** column updates as HyperScale is enabled, and the compute services start. When the operation is completes, the column displays **Enabled/Running**.

Enabling HyperScale from the command line

Perform the following steps on the controller node to enable HyperScale on a compute node using the command line.

To enable HyperScale from the command line

1 On the controller node, source the OpenStack RC file to set the required environment variables.

```
# source keystonerc_admin
```

2 Enable HyperScale on the hypervisor using the `compute-enable` command.

```
# hyperscale compute-enable [--data_intf <data-intf>]
<compute-ip> <compute-pwd> <openstack-pwd>
<compute-meta-disks> <compute-data-disks>
```

Here,

- *<data-intf>* is the name of the interface associated with the data network
- *<compute-ip>* is the IP address associated with the management network interface
- *<compute-pwd>* is the password of the `hyperscale` sudo user account that you created before installing HyperScale

Use commas to specify multiple disks.

Example:

```
# hyperscale compute-enable --data_intf ens224
172.101.101.7 sudoroot123 openstackadmin@123
/dev/dbcomputestore/ssddisk3,/dev/dbcomputestore/ssddisk4
/dev/dbcomputestore/hdddisk3,/dev/dbcomputestore/hdddisk4
```

3 Monitor the operation as it progresses.

Check the following log file on the controller:

`/var/opt/VRTSofcore/log/<computenode_mgmtIPaddress>_config.log`

4 Verify that HyperScale was enabled on the compute node.

```
# hyperscale nodes
```

If there are any failures during the operation, refer to the following log file on the controller node to troubleshoot the issues:

`/var/opt/VRTSofcore/log/<computenode_mgmtIPaddress>_config.log`

Chapter 3

Uninstalling Veritas HyperScale for OpenStack

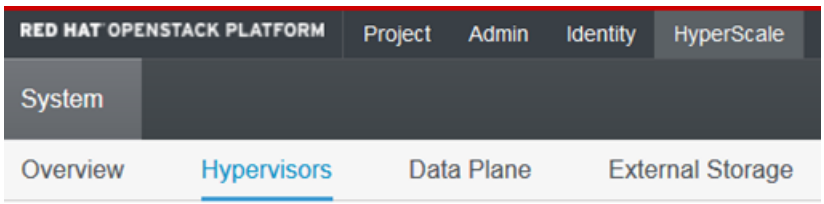
This chapter includes the following topics:

Disabling HyperScale on a compute node

Before you perform the steps in this section, terminate all instances running on the compute node. Otherwise, you cannot disable HyperScale.

To disable HyperScale on a compute node

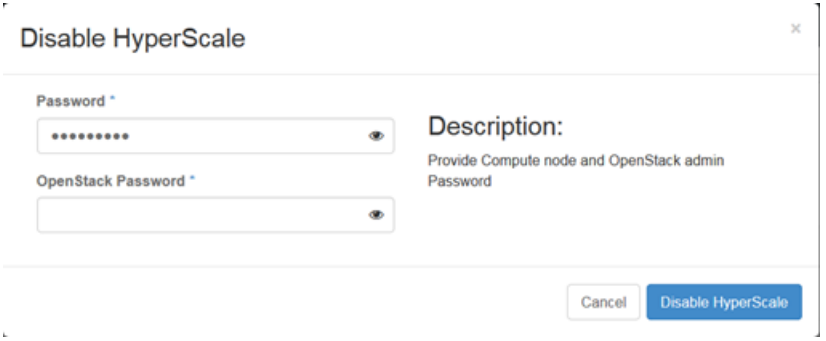
1 Navigate to **Storage > System > Hypervisors**.



2 In the **Hypervisors** table, locate the hypervisor on which you want to disable HyperScale.

3 In the **Actions** column, select **Disable HyperScale**.

4 On the **Disable HyperScale** dialog box, enter the hyperscale sudo user account password and the OpenStack admin password.



5 Click **Disable HyperScale**.

If there are any failures, refer to the log file to troubleshoot the issues. To access the log file for a specific compute node, on the Hypervisors tab, under the Logs column, click the **Download** link for that node.

Removing a data node

Before you remove a data node, make sure of the following:

- HyperScale must be disabled on the compute nodes.
- You have the hyperscale sudo user account password and OpenStack admin password of the data node you want to remove.

To remove a data node

1 Navigate to **HyperScale > System > Data Nodes**.

2 On the **All Data Nodes** table, locate the data node, and in the **Actions** column click **Remove Data Node**.

3 On the **Remove Data** node dialog box, enter the hyperscale sudo user account password and the OpenStack admin password.

4 Click **Remove Data node**.

5 On the **All Data Nodes** page, the **Status** column is updated as the data node is removed.

If any issues occur during this task, you can download the data node log from the **All Data Nodes** page, or view the log on the following path:

`/var/opt/VRTSofcore/log/<datanode_mgmtip>_config.log`

Removing a data node that has computes enabled is possible only in the following cases:

- A network failure on the data node has triggered a datanode failover and that has caused the vdisks to be evacuated to a peer data node
- A permanent data node service failure where the service does not start even after three restart attempts
- If there is a storage failure

Uninstalling HyperScale

Remove HyperScale using the methods described in this section depending on how you deployed HyperScale in your OpenStack environment.

Before uninstalling HyperScale

Ensure the following before you remove HyperScale:

- Terminate all the instances running on the compute nodes.
- Disable HyperScale on all the compute nodes
  See “Disabling HyperScale on a compute node” on page 44.
- Remove data nodes
  See “Removing a data node” on page 45.
- If you have installed HyperScale using the installer bin file, have the following information ready:
  - OpenStack controller's management IP address
  - Openstack admin password
  - MySql HyperScale user password
  - OpenStack Horizon dashboard host IP address
  - hyperscale sudo user account password

Removing HyperScale from a Red Hat OpenStack environment

Run the following command on all the controllers in the Overcloud, one node at a time:

```
# /optVRTSofcore/bin/ofexec \
--operation controller_unconf_tripleo \
--params "openstack_passwd=[password];rabbit_host=[RabbitMQserverVIP];
```

```
rabbitmq_passwd=[password]" \
--run
```

Chapter 4

Troubleshooting HyperScale installation

This chapter includes the following topics:

About HyperScale log files

HyperScale includes one or more log files for the following components.

Table 4-1 Controller logs

Log file	Contents
`/var/opt/VRTSofcore/logs/ofengine.log`	Installation information.
`/var/opt/VRTSofcore/logs/<datanode_mgmtIP>_config.log`	Information about add and remove operations performed on a data node.

Table 4-1 Controller logs *(continued)*

Log file	Contents
/var/opt/VRTSofcore/logs/<computenode_mgmtIP>_config.log	Information about enable and disable HyperScale operations performed on a compute node.
/var/opt/VRTSofcore/logs/controller.log	Information about all HyperScale tasks.
/var/opt/VRTSofcore/logs/default.log	Statistics logs.
/var/opt/VRTSofmn/logs/zookeeper.log	Zookeeper service logs.

Table 4-2 Compute node logs

Log file	Contents
var/opt/VRTSofcore/logs/hypervisor.log	HyperScale MQ service logs on the compute node.
var/log/hypervisor.log	Compute storage service logs.

Table 4-3 Data node logs

Log file	Contents
/var/opt/VRTSofcore/logs/dnhypervisor.log	HyperScale MQ service logs on the data node.
/var/log/HyperScale.log	Data node storage service logs.
/var/log/ofbud.log	Information used to troubleshooting the HyperScale bud service.

Table 4-4 External storage provider and NetBackup logs

Log file	Contents
/var/opt/VRTSofesp/logs/hyperscale-esp.log	Logs each export or import operation initiation request.
/var/opt/VRTSofesp/logs/Export_*ID*/export.log	Detailed information on export job Export_*ID*.
/var/opt/VRTSofesp/logs/Import_*ID*/export.log	Detailed information on import job Import_*ID*.

Table 4-4 External storage provider and NetBackup logs *(continued)*

Log file	Contents
*media server FQDN*_esp_add.log	Logs the progress of registering the NetBackup Media Server with HyperScale.

HyperScale installation may fail with a dpkg error

When you run the HyperScale.bin file to install HyperScale, the installation may fail due to a dpkg error.

The following error may appear in the install log:

```
dpkg: error: dpkg status database is locked by another process
```

Workaround:

This error indicates that another installation or update is already in progress. You can run the following to check which process is holding a lock:

```
# fuser /var/lib/dpkg/lock
```

Wait for the process to complete and then reboot the node where the installation failed. This releases the lock held by another process that may be using the dpkg package.

Then run the HyperScale installer again.

Issues with OpenStack Horizon dashboard after HyperScale installation

After installing HyperScale on the controller, you might not be able to logon to the OpenStack Horizon user interface (UI) or the UI might appear distorted or might even stop working altogether.

Workaround:

Perform the following steps:

1. Clear your browser cache and restart the browser, if prompted.
2. Do the following on the controller node:
   - Connect to the controller node using ssh.
   - Navigate to the openstack-dashboard directory:
     `/usr/share/openstack-dashboard`

- Run the `manage.py` file:

  ```
  # python manage.py compress --force
  ```

- Restart apache2.service:

  ```
  # systemctl restart apache2.service
  ```

Launch the Horizon UI again and verify that you are able to log on and view the HyperScale dashboard.

Enable HyperScale operation for a compute node may fail due to a system clock drift

After installing HyperScale, when you log on to the Horizon user interface (**Horizon | HyperScale > Hypervisors**) and try to enable HyperScale on a compute node, the operation may fail and the UI may display the following message:

```
Failed to enable compute
```

The following error may appear in the compute node log file:

```
ERROR - Compute node is not in time sync with controller.
DEBUG - compute_precheck.execute for module compute_precheck returned
out = Compute node is not in time sync with controller and err = -1
```

The log message indicates that there may be a time drift between the system clocks on the compute node and the controller node. It may also be the case that the compute node and the controller node are not in the same time zone.

Workaround:

Make sure that the system clocks on all the nodes in the HyperScale configuration are synchronized and the nodes belong to the same time zone.

- To align the time zones, run the following command on all the nodes:

  ```
  # sudo timedatectl set-timezone <desiredtimezone>
  ```

  For example, to set the time zone to America/New_York, the command is:

  ```
  # sudo timedatectl set-timezone America/New_York
  ```

- To synchronize the system clock with the time on the controller node, run the following command on the compute node:

  ```
  # date --set="$(ssh root@<controllernodepublicIPaddress> date)"
  ```

Note: If there is a time drift on a data node, run this command on the data node.

- After changing the system time, run the following on the compute node:

  ```
  # service nova-compute restart
  ```

  For a data node, run the following command:

  ```
  # service cinder-volume restart
  ```

After fixing the time drift on all the nodes, try to enable HyperScale on the compute node again.

Enable HyperScale operation for a compute node may fail due to a vxvm package error

When you try to enable HyperScale on a compute node, the operation may some times fail due to an installation error related to the vxvm packages. In such a case, HyperScale performs a roll back operation to revert the node to its previous state. However, the roll back is not able to remove all the vxvm packages cleanly.

As a result, you cannot perform the enable HyperScale operation and the compute node cannot be added to the HyperScale configuration.

Messages similar to the following may appear in the installation log file `/var/opt/VRTSofcore/logs/ofengine.log`, on the HyperScale controller:

```
INFO - cmd_exec Executing: /opt/VRTSofcore/adm/sshpass -e ssh -4
-o PubkeyAuthentication=no -o StrictHostKeyChecking=no
-o ConnectTimeout=20 hyperscale@172.16.10.7
sudo '/usr/bin/dpkg -i /var/tmp/vrtsvxvm*.deb'

INFO - cmd_exec returning: 1
ERROR V-5-1-684 IPC failure: Configuration daemon is not accessible', '

ERROR: The following external ASL package(s) are currently installed:
', '/etc/vx/lib/discovery.d', '

Please remove these before the upgrade install of VxVM.', '
dpkg: error processing archive /var/tmp/vrtsvxvm*.deb (--install):', '
subprocess new pre-installation script returned error exit status 1', '

modprobe: FATAL: Module vxdmp not found.', '
```

```
modprobe: FATAL: Module vxspec not found.', '

WARNING: Unable to unload drivers.', '
You should reboot the system to ensure that the drivers', 'are fully
unloaded.', 'VxVM vxdmpadm

ERROR: No appropriate modules found. Error in loading module "vxdmp".
dpkg: error while cleaning up:
', ' subprocess installed post-installation script returned error
 exit status 1', '
Errors were encountered while processing:', ' /var/tmp/vrtsvxvm*.deb']

ERROR OFEngine - vxvm configuration failed
```

This may also occur on compute nodes on which a previous HyperScale uninstallation was not able to remove all the packages from the node. The enable HyperScale operation may fail because of the previous vxvm packages remaining on the node.

Workaround:

Perform the following steps on the compute node where the enable HyperScale operation failed:

1. Get a list of all the vrtsvxvm packages that exist on the node:

   ```
   # rpm -qa | grep -i vrts
   ```

2. For each vrtsvxvm package listed in the command output, run the following command:

   ```
   # rpm -e [packagename]
   ```

   This will remove all the stale vxvm packages from the node.

After performing the cleanup, you can try to enable HyperScale on the compute node again.

Datanode addition may fail with a unable to resolve host issue

This issue may occur if you deploy HyperScale in a distributed OpenStack environment.

While adding a datanode, if the controller is not able to discover the network interfaces on the specified node, the add operation may fail with the following error:

```
Error: sudo: unable to resolve host <hostname>
```

This occurs because the controller cannot resolve the datanode hostname.

Workaround:

Edit the `/etc/hosts` file on the controller and the data nodes and add the following entry:

```
127.0.0.1 localhost <hostname>
```

HyperScale commands may fail due to an authentication failure

HyperScale commands may sometimes fail with a password authentication failure message. To ensure that the HyperScale commands work without any issues, you may need to modify the project-specific OpenStack environment file (OpenStack rc file).

Here's a sample of the authentication information that should be added to the rc file:

```
export OS_USERNAME=[adminusername]
export OS_PASSWORD=[adminuserpassword]
export OS_AUTH_URL=[http://[keystone IP]:5000/v3]
export PS1='[\u@\h\W(keystone_admin)]\$'
export OS_PROJECT_NAME=[adminprojectname]
export OS_USER_DOMAIN_NAME=Default
export OS_PROJECT_DOMAIN_NAME=Default
export OS_IDENTITY_API_VERSION=3
```

After adding this information you need to source the rc file. It ensures that the environment variables are set for the shell from where you run the HyperScale commands.

Appendix A

Reference

This appendix includes the following topics:

HyperScale packages and services

Veritas HyperScale for OpenStack software is packaged in the form of separate rpm files that are based on the components and the functionality that they provide. The components in each of these package files are mapped to one or more HyperScale services. These services are automatically stopped and restarted whenever you are upgrading your HyperScale configuration or installing a patch.

The following table describes the list of Veritas HyperScale for OpenStack packages and the services that they are mapped to.

Table A-1 HyperScale packages and services

Package name	Services included
`VRTSofmn`	■ `hyperscale-mq-controller` ■ `hyperscale-mq-consumer` ■ `hyperscale-serengiti` ■ `hyperscale-zookeeper`
`VRTSofdn`	■ `hyperscale-dmld` ■ `hyperscale-bud` ■ `hyperscale-bcd` ■ `hyperscale-mq-dnhypervisor`

Table A-1 HyperScale packages and services *(continued)*

Package name	Services included
VRTSofcn	■ hyperscale-mq-storage ■ hyperscale-mq-hypervisor
VRTSofesp	hyperscale-esp

About HyperScale licensing

A product license entitles you to a Veritas Support contract for your Veritas HyperScale for OpenStack deployment. The licensing is based on a subscription model with a validity of 1 year (365 days), 2 years, and 3 years. After the license period expires, you can either renew the subscription to extend the support contract, or upgrade to a newer license depending on the HyperScale release and upgrade version.

The licensing is tied only to the product support and is required only if you wish to avail the services of Veritas Technical Support. You do not need a license to install and deploy HyperScale. You have unrestricted access to all the features provided by HyperScale. Not having a valid license does not impact any feature or functionality usage.

A HyperScale license works on a per deployment basis. You will need a separate license for every single deployment of HyperScale for OpenStack.

How to apply a HyperScale license

You can apply a license either during HyperScale installation or at a later point in time. The license key is installed on the HyperScale controller node.

For the licensing functionality to work, ensure that port `17919` is opened on the controller. Use the same process to install, modify, or upgrade a license.

To apply a HyperScale license

1. Download a valid HyperScale license file on the HyperScale controller.
2. Run the following command on the HyperScale controller node:

   ```
   # hyperscale set-license [absolutepathoflicensefile]
   ```

   For example, the following command installs the license key located at root of the controller node:

   ```
   # hyperscale set-license
   /root/A1825559965_QTY30_HYPERSCALE_2_0_OPENSTACK_MAINT_1842037161.slf
   ```

   The following command output confirms that the license is installed successfully:

   ```
   +----------+-----------------+
   | Property | Value
   +----------+-----------------+
   | Status   | [0, u'Success']
   +----------+-----------------+
   ```

Viewing the HyperScale license status

Run the following command on the HyperScale controller:

```
# hyperscale get-license-status
```

The command output confirms whether or not a license is installed and active.

For example, the following command output indicates that a license is not installed:

```
+-------------+-------------+------------------+
| host_ip     |  is_active  | state
+-------------------------------+------------------+
| 172.100.10.1 |  False      | No License Found
+-------------+-------------+------------------+
```

The following command output indicates that a license is installed and active:

```
+-------------+-------------+---------+
| host_ip     |  is_active  | state
+-------------------------------+---------+
| 172.100.10.1 |  True       | Active
+-------------+-------------+---------+
```

About HyperScale telemetry

During installation, HyperScale automatically enables a telemetry service that collects anonymous system and product usage information from all the HyperScale nodes in the configuration. This data is sent to Veritas and is used for planning product improvements and future enhancements.

The information that is collected does not include any confidential or proprietary data and is limited to product-specific configuration details and heuristics.

You can disable the telemetry service altogether by deleting the following cronjob from all the HyperScale controller nodes:

```
23 18 21 * *  /usr/bin/python /usr/bin/hyperscale-telemetry > /dev/null 2> /dev/null
```