Veritas Access Installation Guide

Linux

7.4.2.400

August 2021

VERITAS™

Veritas Access Installation Guide

Last updated: 2021-08-04

Legal Notice

Veritas Technologies LLC
2625 Augustine Drive
Santa Clara, CA 95054

http://www.veritas.com

Technical Support

Technical Support maintains support centers globally. All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policies. For information about our support offerings and how to contact Technical Support, visit our website:

https://www.veritas.com/support

You can manage your Veritas account information at the following URL:

https://my.veritas.com

If you have questions regarding an existing support agreement, please email the support agreement administration team for your region as follows:

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

Documentation

Make sure that you have the current version of the documentation. Each document displays the date of the last update on page 2. The latest documentation is available on the Veritas website:

https://www.veritas.com/content/support/en_US/dpp.Appliances.html

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:

APPL.docs@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

Licensing in Veritas Access

This chapter includes the following topics:

About Veritas Access product licensing

You need to procure a license to use the Veritas Access software.

Veritas Access supports the following base licensing models:

- Per-TB
  See "Per-TB licensing model" on page 9.

  Note: Veritas recommends that you use the Per-TB license model with Veritas Access 7.4.2.400.

- TB-Per-Core
  See "TB-Per-Core licensing model" on page 10.
- Per-Core
  See "Per-Core licensing model" on page 12.

You can also procure an add-on license to use the Veritas Data Deduplication service.

The licensing models support the following licensing methods:

- Perpetual
- Subscription
- Trialware

For more information about functional enforcements and notes related to licensing:

See "Notes and functional enforcements for licensing" on page 12.

Per-TB licensing model

In 74.2 release, Veritas introduces the Per-TB licensing model. The license model is based on both capacity and time-period. You can use the license model for Veritas Access as per your requirement for the raw capacity.

You can use the Veritas Access software to manage the license model. The licensing model uses the Per-TB license meter. Veritas provides support of 12 months, 24 months, and 36 months for the licensing model.

The Per-TB license model supports the following licensing methods:

- Perpetual license: You can procure a perpetual license to use the Veritas Access software for an unlimited period.
- Subscription license: You can procure a subscription license to use the Veritas Access software for a specified period. After the subscription period is over, you need to renew the subscription license. The subscription period can be of 1 year, 2 years, 3 years, and so on.

  Two types of maintenance support are available based on pricing for the subscription license method:

  - Basic
  - Essential
- Trialware license: You can download a trial version of the Veritas Access software from the Veritas website to evaluate it for 60 days.

If you exceed the licensed storage capacity, the product usage is not affected. However, Veritas recommends that you procure a new license or renew your license to a higher storage capacity.

Table 1-1 Licensing for new and existing customers

Customer type	Purchase options	Existing licensing model	New licensing model	Existing meter	New meter
New	New Licenses	N/A	Perpetual	N/A	Per-TB
			Subscription		
Existing	New Licenses	Perpetual	Perpetual	Per-Core	Per-TB
		Subscription	Perpetual or Subscription		
	Renewal	Perpetual	Perpetual	Per-Core	Per-Core
		Subscription	Subscription		

Note: The Per-Core and TB-Per-Core licensing models of the earlier releases are supported in this release only if you upgrade to Veritas Access 7.4.2.400 from an earlier version of Veritas Access that uses any one of the two license models. However, Veritas recommends that you use the Per-TB licensing model instead of the Per-Core and TB-Per-Core licensing models.

For more information about the Per-Core and TB-Per-Core licensing models, see the following sections:

See "Per-Core licensing model" on page 12.

See "TB-Per-Core licensing model" on page 10.

See "Notes and functional enforcements for licensing" on page 12. for more information about functional enforcements and notes related to licensing.

You can use the `vxlicrep` command to see the details of the installed licenses. The **Count** field displays the licensed capacity.

TB-Per-Core licensing model

The TB-Per-Core licensing model is based on both capacity per-core and time period. You can license Veritas Access as per your requirement for the raw capacity. You can use the Veritas Access software to manage the license model. Veritas provides support of 12 months, 24 months, and 36 months for the licensing model.

Note: The TB-Per-Core licensing model is supported in this release only if you upgrade an earlier version of Veritas Access that uses this license model to Veritas Access 7.4.2.400. However, Veritas recommends that you use the Per-TB licensing model.

For more information on the Per-TB licensing model:

See "Per-TB licensing model" on page 9.

When you exceed the licensed storage capacity, the product usage is not affected. However, Veritas recommends that in such cases, you must procure or renew your license to a higher capacity.

Veritas recommends the tier that is best suited for your needs based on your current system configuration across the clusters. The new metering and recommended tier is based on the capacity utilization to the core ratio. Capacity utilization is the raw capacity that is used while the core refers to the physical cores present across the cluster. This information is also available in the Veritas Access Management Console in the **Recommended Tier**.

Table 1-2 Licensing methods

Tiering model	TB-Per-Core meter capacity	Capacity tier range	Time-based licensing
Premium	TB to core ratio <= 4 TB/core	2001 TB - Unlimited	Subscription - 1 year, 2 years, and 3 years Perpetual - Unlimited for a product version Trialware- 60 days
Standard	TB to core ratio Between 4 TB/core and 25 TB/core	2001 TB - Unlimited	Subscription - 1 year, 2 years, and 3 years Perpetual - Unlimited for a product version
Basic	TB to core ratio > 25 TB/core	2001 TB - Unlimited	Subscription - 1 year, 2 years, and 3 years Perpetual - Unlimited for a product version

The trialware has the premium tier licensing model with a storage capacity range of 2001 TB – Unlimited. You can upgrade to any valid Per-Core license from the trialware.

For more information about functional enforcements and notes related to licensing:

See "Notes and functional enforcements for licensing" on page 12.

Per-Core licensing model

The Per-Core licensing model is based on time period. You can license Veritas Access as per your requirement for the raw capacity. Veritas provides support of 12 months, 24 months, and 36 months for the licensing model.

Note: The Per-Core licensing model is supported in this release only if you upgrade to Veritas Access 7.4.2.400 from an earlier version of Veritas Access that uses this license model. However, Veritas recommends that you use the Per-TB licensing model.

For more information about the Per-TB licensing model:

See "Per-TB licensing model" on page 9.

The time-based license category includes the following licenses:

- Perpetual license: You can procure a perpetual license to use the Veritas Access software for an unlimited period.
- Subscription license: You can procure a subscription license to use the Veritas Access software for a specified period. After the subscription period is over, you need to renew the subscription license. The subscription period can be of 1 year, 2 years, 3 years, and so on.
- Trialware license: You can download a trial version of the Veritas Access software from the Veritas website to evaluate it for 60 days.

For more information about functional enforcements and notes related to licensing:

See "Notes and functional enforcements for licensing" on page 12.

Notes and functional enforcements for licensing

This section provides details about functional enforcements and notes for licensing.

- You must provide a valid license during the product installation. If you do not provide a valid license, a 60 days of trialware license is installed.
- If you fail to procure or renew your license before the expiry date, a grace period of 60 days is provided without any effect on the product usage.
- If you fail to procure or renew your license after the grace period, the services fail to start after a system restart or when the services such as, NFS, CIFS, FTP, S3, and Veritas Data Deduplication are restarted.

- Veritas reserves the right to ensure entitlement and compliance through auditing.
- If you encounter problems while licensing this product, visit the Veritas Licensing Support website.
  https://www.veritas.com/licensing/process

Table 1-3 Functional enforcements of Veritas Access licensing

Enforcement	Action
During Validity	None
During Grace period	Persistent message (in the Veritas Access Management Console only)
Post Grace Period	Before you restart the node, you can stop the NFS, CIFS, FTP, S3, and Veritas Data Deduplication services, but you cannot start the services again (even if you have not restarted the node). After you restart the node, the NFS, CIFS, FTP, S3, and Veritas Data Deduplication do not come online on the restarted node.

If you add the Veritas Access license using the command line interface:

- When you restart a node after the license is expired, the NFS, CIFS, FTP, S3, and Veritas Data Deduplication services are stopped on that node. You can use the `support services show` command to display the node-wise status of the service.
- You can start, stop, and check the status of NFS, CIFS, FTP, S3, and Veritas Data Deduplication services.
- You can add the license by using the `System> license add` command. The `license add` command provides support for the `scp` path as well.
- The `System> license list` and `System> license list details` commands provide details for the license that is installed on each node of the cluster.

Chapter 2

System requirements

This chapter includes the following topics:

Important release information

Review the *Veritas Access Release Notes* for the latest information before you install the product.

The hardware compatibility list contains information about supported hardware and is updated regularly. You can use any commodity hardware that is certified and mentioned in the hardware compatibility list.

For the latest information on supported hardware, see the Hardware Compatibility List (HCL) at:

https://sort.veritas.com/documents/doc_details/isa/7.4.2.400/Linux/CompatibilityLists/

The 7.4.2.400 update can be installed only on Veritas Access 7.4.2 release with Red Hat Enterprise Linux (RHEL) version 7.7 or 7.9. If you are on an earlier version of RHEL, you must upgrade to version 7.7 or 7.9 before installing the update.

System requirements

Table 2-1 lists the per-node system requirements for running the Veritas Access system software.

Table 2-1 System requirements for Veritas Access

Minimum	Recommended
Each Veritas Access node using a 64-bit Intel-based server architecture that is compatible with Red Hat Enterprise Linux (RHEL) 7 Update 7 and 9, or AMD64, or Intel EMT. Itanium is not supported.	Two nodes of dual or quad core processors at 2.0 GHz or later for optimal performance.
32 GB Error Correcting Code (ECC) RAM	The recommended values depend on the expected workload.
One internal drive with size equal to size of RAM + 60 GB	Dual boot drives each of size RAM + 60 GB or more capacity. In an FSS-based environment, additional internal drives (SSD + HDD) are required.
Four 1G Ethernet interfaces (Two ethernet interface are used for public and two for private network.)	Four 10G ethernet interfaces (Two ethernet interface are used for public and two for private network.).
One Fibre Channel Host Bus Adapters (HBA)	Two Fibre Channel Host Bus Adapters (HBAs) for high availability (HA) if you use shared LUNs that need to be mapped over a Fibre Channel protocol. If the environment has only DAS or iSCSI disks, the HBA is not required.
Internal/external USB DVD-ROM DVD drive	N/A
Redundant power supply	Veritas recommends it but it is not required.
SmartIO caching feature	A PCI-based SSD card if you want to use the SmartIO caching feature.
Minimum number of servers required is 1	N/A

Table 2-2 lists the operating system (OS) partition requirements for running the Veritas Access system software.

Table 2-2 Operating system partition requirements for Veritas Access

Partition	Recommended size (Minimum)	Details
/opt	100 GB	To store the Veritas Access software, logs, and core dumps.
/usr	3 GB	To install the dependent OS rpms.
swap	8 GB	To swap space when physical memory is full.
/	30 GB	Used for operating system.

Note: The aforementioned operating system (OS) partition requirements are only for Veritas Access, additional space is required for OS specific packages, which needs to be accounted and allocated as required.

Linux requirements

The Veritas Access 7.4.2.400 release requires the following OS version:

- RHEL 7 Update 7 and 9

The certification of the RHEL OS updates requires a new minor version of Veritas Access. You need an agreement with Veritas to install the RHEL OS updates.

The minimum OS requirements are enforced during the Veritas Access installation. A Kickstart file is also available on request for Veritas Access 7.4.2.400 to assist partners with the OS installation requirements.

You can install OS patches, including security vulnerability patches, without requiring certification from Veritas. However, OS Kernel RPMs should not be patched without specific approval from Veritas.

The following table lists the OS requirements for Veritas Access:

Table 2-3 OS requirements

RHEL OS version	Kernel version
RHEL 7 update 7	3.10.0-1062.el7
RHEL 7 update 9	3.10.0-1160.el7

Required operating system RPMs and patches

Veritas has categorized the operating system (OS) RPMs into four groups. These RPMs are installed automatically during the product installation.

Category 1

- This set of RPMs are kernel RPMs that are required to be installed with exact predefined RPM versions only.
- The required RPM versions are different for:
  - RHEL 7 Update 7 and Update 9

Category 2

- This set of RPMs include the OS libs and OS packages that must be installed with minimum predefined RPM versions.
- The required RPM versions are different for:
  - RHEL 7 Update 7 and Update 9

Category 3

- This set of RPMs are required by Category 2 RPMs as dependencies, their installation is enforced by Red Hat.
- Veritas Access does not require any specific versions of these RPMs to be installed.
- The versions of these RPMs are determined by Red Hat.
- The RPMs in this category can be patched using official Red Hat patches.
- Veritas does not document these RPMs as required RPMs for Veritas Access.

Category 4

- These are third-party RPMs that are included in the Veritas Access ISO.
- These RPMs are not operating system RPMs. It includes Samba and other third-party products.
- Veritas installs these RPMs as they are included in the Veritas Access ISO.

Software requirements for installing Veritas Access in a VMware ESXi environment

Table 2-4 Software requirements for installing Veritas Access in a VMware ESXi environment

Operating system (OS)	VMware Versions	IP addresses
RHEL 7.7 and 7.9	VMware ESXi 6.5, 6.7	Nine IPs are required for a two-node cluster with two public NICs: ■ Four IP addresses are used to configure physical IPs. ■ Four IP addresses are used to configure virtual IPs. ■ One IP address is used for the management console. ■ One IP address is used for replication.

Hardware requirements for installing Veritas Access virtual machines

Table 2-5 Hardware requirements for installing Veritas Access virtual machines

Item	Description
CPU	1 CPU – 64 bit, dual, or quad core, 2.0 GHz or later
RAM	■ 32 GB of RAM for physical servers ■ 60 GB (or more) RAM size internally available storage capacity for boot disk
Network interface card (NIC)	Four NIC cards ■ Two NIC cards for public network (minimum) ■ Two NIC cards for private network
Fibre Channel HBA	Two-port Fibre Channel HBAs are required if you want to use shared LUNs. If the environment has only DAS disks, then the HBA requirement is optional.

Management Server Web browser support

The following are the supported Web browsers for Veritas Access:

Table 2-6

Browser	Version	Comments
Microsoft Edge	40.x	JavaScript: Enabled Cookies: Enabled
Firefox	4.x and later	JavaScript: Enabled Cookies: Enabled
Google Chrome	10 and later	JavaScript: Enabled Cookies: Enabled

Additional considerations for supported Web browsers:

- Your browser must support JavaScript 1.2 or later.
- If you use pop-up blockers (including Yahoo Toolbar or Google Toolbar), either disable them or configure them to accept pop-ups from the Veritas Access node to which you connect.
- If you cannot add the site to the list of trusted sites, enable the Binary and script Behaviors option in security settings.
- You must install Adobe Flash plug-in version 10, or later.

Required NetBackup versions

See "Supported NetBackup client versions" section in the *Veritas Access Release Notes*.

Required OpenStack versions

The OpenStack drivers, Cinder and Manila, are supported on the RHEL 7 OS and the OpenStack Kilo, Mitaka, Newton, or Ocata releases.

The Cinder and Manila drivers were tested with the following:

- OpenStack Kilo, Mitaka, Newton, or Ocata versions from the DevStack repository
- OpenStack RDO

Note: The Manila driver works only with kernel NFS. It does not work with NFS-Ganesha.

Required IP version 6 Internet standard protocol

Table 2-7 describes the IP version 6 (IPv6) Internet standard protocol.

Table 2-7 IPv6 Internet standard protocol

Description	Example format
Preferred form	ABCD:EF01:2345:6789:ABCD:EF01:2345:6789
Compressed form	FF01::101
Mixed form	0:0:0:0:0:FFFF:129.144.52.38

Network and firewall requirements

Table 2-8 displays the default ports that Veritas Access uses to transfer information.

Table 2-8 Default Veritas Access ports

Port	Protocol or Service	Purpose	Impact if blocked
21	FTP	Port where the FTP server listens for connections. **Note:** Users can configure another port if desired.	FTP features are blocked.
22	SSH	Secure access to the Veritas Access server	Veritas Access is not accessible.
25	SMTP	Sending SMTP messages.	The SMTP messages that are sent from Veritas Access are blocked.
53	DNS queries	Communication with the DNS server	Domain name mapping fails.
111	rpcbind	RPC portmapper services	RPC services fail.

Table 2-8 Default Veritas Access ports *(continued)*

Port	Protocol or Service	Purpose	Impact if blocked
123	NTP	Communication with the NTP server	Server clocks are not synchronized across the cluster. NTP-reliant features (such as DAR) are not available.
139	CIFS	CIFS client to server communication	CIFS clients cannot access the Veritas Access cluster
161	SNMP	Sending SNMP alerts	SNMP alerts cannot be broadcast.
445	CIFS	CIFS client to server communication	CIFS clients cannot access the Veritas Access cluster.
514	syslog	Logging program messages	Syslog messages are not recorded.
756, 757, 755	statd	NFS statd port	NFS v3 protocol cannot function correctly.
2049	NFS	NFS client to server communication	NFS clients cannot access the Veritas Access cluster.
3172, 3173	ServerView	ServerView port	ServerView cannot work.
3260	iSCSI	iSCSI target and initiator communication	Initiator cannot communicate with the target.
4001	mountd	NFS mount protocol	NFS clients cannot mount file systems in the Veritas Access cluster.
4045	lockd	Processes the lock requests	File locking services are not available.

Table 2-8 Default Veritas Access ports *(continued)*

Port	Protocol or Service	Purpose	Impact if blocked
5634	HTTPS	Management Server connectivity	Web GUI may not be accessible.
56987	Replication	File synchronization, Veritas Access replication	Veritas Access replication daemon is blocked. Replication cannot work.
8088	REST server	REST client to server communication	REST client cannot access REST API of Veritas Access.
8143	S3	Data port for Veritas Access S3 server	User will not able to use Veritas Access object server.
8144	ObjectAccess service	Administration port for Veritas Access S3 server.	User cannot create access or secret keys for using Objectaccess service.
10082	spoold	Veritas Data Deduplication engine	Cannot use the Veritas Data Deduplication service for deduplicating data
10102	spad	Veritas Data Deduplication manager	Cannot use the Veritas Data Deduplication service for deduplicating data
11211	Memcached port	Veritas Access command-line interface framework	Veritas Access command-line interface cannot function correctly, and cluster configuration may get corrupted.
30000:40000	FTP	FTP passive port	FTP passive mode fails.
14161	HTTPS	Veritas Access GUI	User is unable to access Veritas Access GUI

Table 2-8 Default Veritas Access ports *(continued)*

Port	Protocol or Service	Purpose	Impact if blocked
51001	UDP	LLT over RDMA	LLT is not working.
51002	UDP	LLT over RDMA	LLT is not working.

NetBackup ports

NetBackup uses TCP/IP connections to communicate between one or more TCP/IP ports. Depending on the type of operation and configuration on the environment, different ports are required to enable the connections. NetBackup has different requirements for operations such as backup, restore, and administration.

Table 2-9 shows some of the most-common TCP and UDP ports that Veritas Access NetBackup uses to transfer information. For more information, see the *Veritas NetBackup Security and Encryption Guide*.

Table 2-9 Default NetBackup TCP and UDP ports

Port Range	Protocol
1556	TCP, UDP
13701-13702, 13705-13706	TCP
13711, 13713, 13715-13717, 13719	TCP
13720-13722	TCP, UDP
13723	TCP
13724	TCP, UDP
13782-13783	TCP, UDP
13785	TCP

CIFS protocols and firewall ports

For the CIFS service to work properly in an Active Directory (AD) domain environment, the following protocols and firewall ports need be allowed or opened to enable the CIFS server to communicate smoothly with Active Directory Domain Controllers and Windows/CIFS clients.

Internet Control Message Protocol (ICMP) protocol must be allowed through the firewall from the CIFS server to the domain controllers. Enable "Allow incoming echo request" is required for running the CIFS service.

Table 2-10 lists additional CIFS ports and protocols.

Table 2-10 Additional CIFS ports and protocols

Port	Protocol	Purpose
53	TCP, UDP	DNS
88	TCP, UDP	Kerberos
139	TCP	DFSN, NetBIOS Session Service, NetLog
445	TCP, UDP	SMB, CIFS, SMB2, DFSN, LSARPC, NbtSS, NetLogonR, SamR, SrvSvc
464	TCP, UDP	Kerberos change or set a password
3268	TCP	LDAP GC
4379	TCP	CTDB in CIFS

Table 2-11 lists the ports that are required for LDAP with SSL.

Table 2-11 LDAP with SSL ports

Port	Protocol	Purpose
636	TCP	LDAP SSL
3269	TCP	LDAP GC SSL

Maximum configuration limits

The maximum configuration limits for configuring the Veritas Access system software are as follows:

Table 2-12 Maximum configuration limits

Veritas Access system software	Configuration limit
File system size	5 PB for non-scale-out file system without cloud tiering support
Veritas Access nodes	20

Table 2-12 Maximum configuration limits *(continued)*

Veritas Access system software	Configuration limit
Supported LUNs	The maximum number of disks is theoretically limited to the number that can be attached to the operating system. However, it has only be tested in the thousands.
Supported file systems	500
Tiers within a file system	2 (primary tier and secondary tier)

Chapter 3

Preparing to install Veritas Access

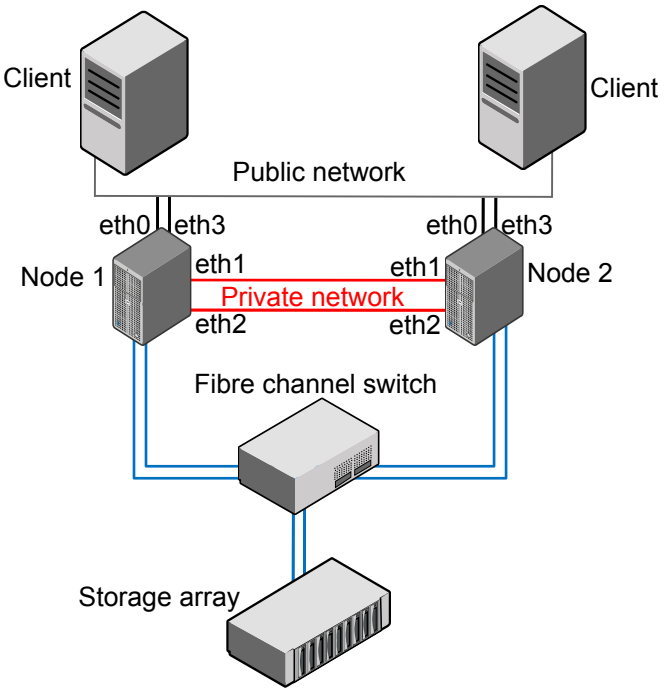
This chapter includes the following topics:

Overview of the installation process

The Veritas Access cluster is a set of connected servers called "nodes." Together these nodes form a unified entity called a cluster.

Figure 3-1 shows an example of an Veritas Access cluster.

Figure 3-1 Sample of Veritas Access cluster overview



Note: The NIC names mentioned in Figure 3-1 are only for examples. You need to determine the actual names of your NICs during the installation.

An overview of the Veritas Access software installation includes the following steps:

- Gather network information from your network administrator.
- Connect your network hardware.
- Install the operating system on each of the nodes.
- Download the release update from the Download Center on the Veritas Support website. Install Veritas Access on the node. If the driver node is one of the nodes of the cluster, you must start the installer from the console of the node. If the driver node is not part of the cluster, the installer can be run from the driver node to install and configure the cluster over an ssh connection.
  From the Veritas Access 7.2 release, the installer can be run from any node of the cluster.
  See "Installing and configuring the Veritas Access software on the cluster" on page 53.

See "About the driver node" on page 50.

- Run the installation and configuration on the node to configure the entire cluster. Installation times vary depending on your configuration.

Hardware requirements for the nodes

The following table summarizes the hardware requirements for each node.

Table 3-1 Hardware requirements for the nodes

Item	Requirements
Network interface card (NIC)	At least four NICs are required for each node. Two NICs connected to a private network. ■ For a two-node cluster, either cross connect two private NICs on each node or use a switch. ■ If there are more than two nodes in the cluster, make sure that you have a dedicated switch (or a public or private switch with a dedicated VLAN) and that all the private NICs are connected to the switch. Connect two public NICs from each node to the public network. The gateway must be reachable to each public NIC.
IP address	For a two-node cluster, make sure that you have nine IP addresses available. ■ Four IP addresses are used to configure physical IPs. ■ Four IP addresses are used to configure virtual IPs. ■ One IP address is used to configure the Operations Manager console. ■ One IP address is used for replication, which is optional. Make sure that these nine IP addresses are different from the IP addresses that are already assigned to the target cluster nodes to install Veritas Access over Secure Shell (ssh).

About using LLT over the RDMA network for Veritas Access

Remote direct memory access (RDMA) allows server-to-server data movement directly between application memories with minimal CPU involvement. RDMA provides fast interconnection between user-space applications or file systems between nodes over InfiniBand networks with RDMA-enabled network cards and

switches. In a clustering environment, RDMA allows applications on separate nodes to transfer data at a faster rate with low latency and less CPU usage.

RDMA over InfiniBand networks in the Veritas Access clustering environment

Veritas Access uses Low Latency Transport (LLT) for data transfer between applications on nodes. LLT functions as a high-performance, low-latency replacement for the IP stack, and is used for all cluster communications. It distributes (load balances) internode communication across all available private network links. This distribution means that all cluster communications are evenly distributed across all private network links (maximum eight) for performance and fault resilience. If a link fails, traffic is redirected to the remaining links. LLT is also responsible for sending and receiving heartbeat traffic over network links. Using LLT data transfer over an RDMA network boosts performance of both file system data transfer and I/O transfer between nodes.

Network interface cards (NICs) and network switches that support RDMA are required to enable the faster application data transfer between nodes. You also need to configure the operating system and LLT for RDMA.

See "Configuring LLT over RDMA for Veritas Access" on page 30.

How LLT supports RDMA for faster interconnections between applications

Low Latency Transport (LLT) maintains two channels (RDMA and non-RDMA) for each of the configured RDMA links. Both RDMA and non-RDMA channels can transfer data between the nodes. LLT provides separate Application Program Interfaces (APIs) to the clients (such as CFS and CVM) to use these channels. The RDMA channel is mainly used for data transfer by the client; while the non-RDMA channel is created over the UDP layer, and LLT uses it mainly for sending and receiving heartbeats. Group Membership Services/Atomic Broadcast (GAB) decides cluster membership for the cluster according to the health of the non-RDMA channel. The connections of the RDMA and non-RDMA channels are under separate management, while the connect and disconnect operations for the RDMA channel are triggered based on the status of the non-RDMA channel.

If the non-RDMA channel is up while the RDMA channel is down, the data is transferred over the non-RDMA channel with lower performance until the RDMA channel is fixed. The system logs display a message when the RDMA channel is up or down.

LLT uses the Open Fabrics Enterprise Distribution (OFED) layer and the drivers on the operating system to communicate with the hardware. LLT over RDMA allows

applications running on one node to directly access the memory of an application running on another node over an RDMA-enabled network. While over a non-RDMA network, LLT clients have to create intermediate data copies to complete the read or write operation on the application. The RDMA network brings low latency, higher throughput, and minimized CPU host usage, and boosts application performance. LLT and GAB clients CFS and CVM can use LLT over RDMA.

Configuring LLT over RDMA for Veritas Access

During the Veritas Access installation, the installer automatically configures LLT over RDMA if there are InfiniBand NICs on the cluster nodes, unless the InfiniBand NICs are excluded.

This section describes the required hardware and configuration for LLT to support RDMA for Veritas Access. The high-level steps to configure LLT over RDMA are as follows:

1. Choose NICs, network switches, and cables that support RDMA.

Table 3-2 RDMA-enabled hardware

Hardware	Supported types	Reference
Network card	Mellanox-based Host Channel Adapters (HCAs) (VPI, ConnectX, ConnectX-2 and 3)	For detailed installation information, refer to the hardware vendor documentation.
Network switch	Mellanox, InfiniBand switches Ethernet switches must be Data Center Bridging (DCB) capable	For detailed installation information, refer to the hardware vendor documentation.
Cables	Copper and Optical Cables, InfiniBand cables	For detailed installation information, refer to the hardware vendor documentation.

2. Connect the first two non-excluded InfiniBand NICs as private NICs.

Note: Cross-links connection is not supported for private NICs in an RDMA environment.

3. Make sure that the required packages to enable RDMA, InfiniBand drivers, and utilities are installed with the base operating system. Or they can be installed from the yum repository.

Table 3-3 Drivers and utilities required for RDMA, InfiniBand, or an Ethernet network

Packages	Drivers and utilities
Device drivers for RDMA operations	■ `libmthca` ■ `libmlx4` ■ `rdma` ■ `librdmacm-utils`
OpenSM-related package	■ `opensm` ■ `opensm-libs` ■ `libibumad`
InfiniBand troubleshooting and performance tests	■ `ibutils` ■ `infiniband-diags` ■ `perftest`
libibverbs packages for InfiniBand operations	■ `libibverbs-devel` ■ `libibverbs-utils`

How the Veritas Access installer configures LLT over RDMA

At a high level, the Veritas Access installer configures the InfiniBand NICs as LLT over RDMA for Veritas Access by the following steps:

1. After the InfiniBand NICs are detected, the installer installs the required operating system packages.
2. Choose InfiniBand NICs as private NICs, if the NIC is not excluded.
3. Assign static private IPs and configure LLT to use InfiniBand NICs.

LLT over RDMA sample /etc/llttab

The following is a sample of LLT over RDMA in the `etc/llttab` file.

```
rdma-01:~ # cat /etc/llttab
set-node rdma-01
set-cluster 54791
link priveth0 udp - rdma 51001 - 172.16.0.3 172.16.0.255
link priveth1 udp - rdma 51002 - 172.16.1.3 172.16.1.255
set-flow highwater:1000
set-flow lowwater:800
```

Connecting the network hardware

Before you install the Veritas Access software, you must assemble a cluster by configuring all the nodes with the required network hardware, and connecting the Ethernet interfaces to the private and the public networks.

To assemble the cluster, do the following:

- Determine a preferred location for the cluster.
- Make sure that each node has at least two redundant Ethernet interfaces (gigabit Ethernet) to connect to a private network for cluster internal control.
- Make sure that each node has at least two additional Ethernet interfaces (gigabit Ethernet) to connect to the public network. You can use the public Ethernet interfaces from the embedded interfaces on the motherboard or from the add-on (PCI) network adapter interfaces.
- To connect the public NICs, connect one end of the Ethernet cables to the Ethernet interfaces on the back of the nodes. Connect the other end of the Ethernet cables to your corporate network so that they can reach the gateway. At least two public interfaces are required for each node.
- To connect the private NICs, use the first two available NICs when sorted by NIC name. Available NICs are those not connected to the public network or excluded from the node.
  For example, if your NICs are eth1, eth2, eth3, and eth4, and none of the NICs are connected to the public network or excluded, then use eth1 and eth2 as the private NICs.
  Connect one end of the Ethernet cables to Ethernet interface 1 and 2 on the back of the nodes. For a 2-node cluster, connect the other end of the Ethernet cables to the corresponding Ethernet interfaces on the second node. For a cluster with more than 2 nodes, connect the other end of the Ethernet cables to a dedicated switch or VLAN.

  Note: Veritas recommends to use InfiniBand NICs to configure LLT over RDMA for Veritas Access. Connect InfiniBand NICs as private or exclude the NICs when you install Veritas Access.

  See “About using LLT over the RDMA network for Veritas Access” on page 28.

  See “Excluding a NIC” on page 67.

- Ask your network administrator for the IP addresses to use in the Veritas Access installation. The number of IP addresses you need depends on the number of nodes and number of network interface cards in your cluster.

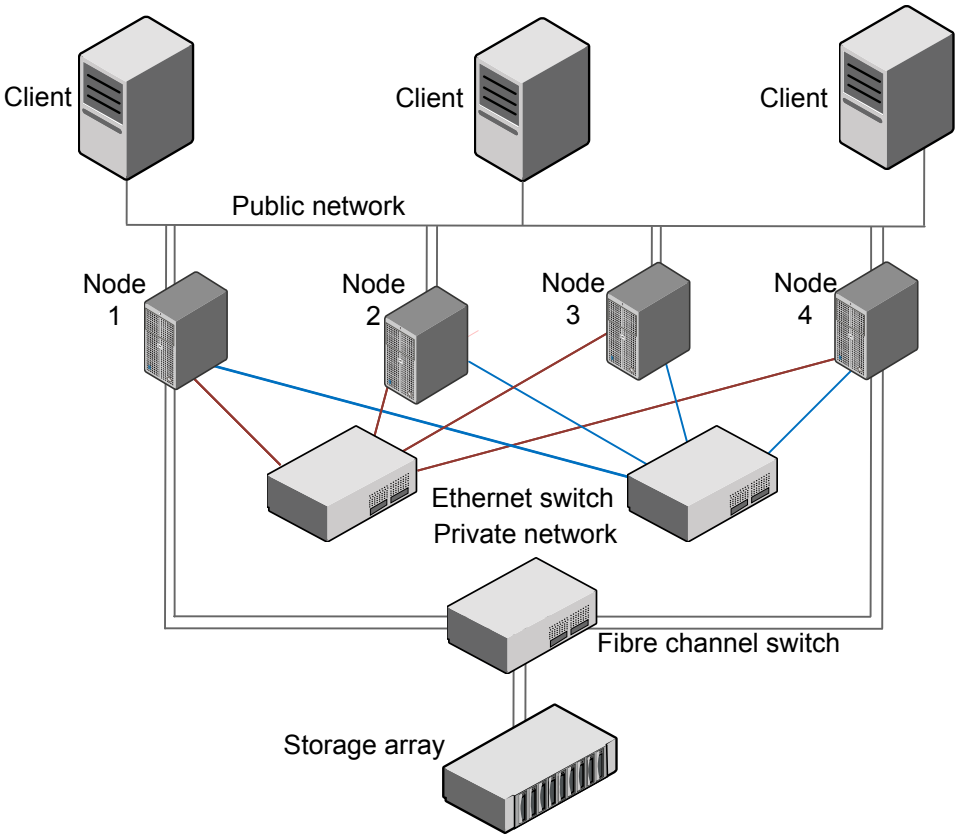
You need at least one IP address per node per public interface. For virtual IP addresses, you can configure the virtual IP addresses later using the Veritas Access command-line interface if you input 0 for the number of virtual IP addresses per NIC during installation time.

Veritas Access supports both Internet Protocol version 4 (IPv4) or Internet Protocol version 6 (IPv6), but they cannot be mixed.

Physical IP address	An IP address that is associated with a specific Ethernet interface address and cannot automatically be failed over.
Virtual IP address (VIP)	An IP address whose association to a specific Ethernet interface (VIP) can be failed over to other interfaces on other nodes by the Veritas Access software.
Console IP address	A dedicated virtual IP address that is used to communicate with the Veritas Access cluster Management Console. This virtual IP address is assigned to the master node. If the master node fails, the Veritas Access software automatically selects a new master node from the cluster and fails the console IP address over to it.

Figure 3-2 shows a diagram of a four-node cluster.

Figure 3-2 Private network configurations: four-node cluster



Note: Two or more Veritas Access private networks cannot be configured on the same IPv4 network.

About obtaining IP addresses

The Veritas Access installation process lets you configure IP addresses for 1 to 20 nodes. The default is two nodes.

Note: You can configure either IPv4 addresses or IPv6 addresses (depending on what you use when installing Veritas Access), but not both. Do not use IP addresses starting with 172.16.X.X either as physical IP addresses or virtual IP addresses since this range of IP addresses are used for the private network.

You need to obtain physical IP addresses, virtual IP addresses, and a netmask for the chosen public network from the network administrator in charge of the facility where the cluster is located. All IP addresses (both physical and virtual) must be part of the same subnet and use the same netmask as the node's access IP.

By design, the installer does not support the use of the localhost (127.0.0.1) IP address during installation

---

Note: Netmask is used for IPv4 addresses. Prefix is used for IPv6 addresses. Accepted ranges for prefixes are 0-128 (integers) for IPv6 addresses.

The information you obtained from the network administrator is used to configure the following:

- Physical IP addresses
- Virtual IP addresses
- Console IP address
- Replication IP address (optional)
- IP address for the default gateway
- IP address for the Domain Name System (DNS) server
- DNS domain name
- IP address for the Network Time Protocol (NTP) server (optional)
- Virtual IP address for Veritas NetBackup (optional)

About calculating IP address requirements

This section provides an example of how to calculate IP addresses for a two-node cluster. In this example, all the nodes in the cluster have the same hardware configuration. Therefore, the number of network interface cards (NICs) is the same for all the nodes in the cluster.

- Two private NICs and two public NICs should be connected to respective networks.
- One public IP address should be assigned to one of the public interface for installation over ssh. None of the private interfaces should have the IP address in the same network segment.
- The public IP address must be made permanent by writing it to the network configuration file `/etc/sysconfig/network-scripts/ifcfg-ethX`.

Table 3-4 Example calculation of required IPs for a standard configuration

Number of IPs	Item
2	Number of nodes in the cluster
4	Number of interfaces on each node
2	Number of the private interfaces that are required for each node

After two private interfaces on each node are selected, all remaining interfaces act as public interfaces.

To calculate the number of public interfaces per node

- The total number of interfaces on the node, minus the number of private interfaces that are required on a node, is equal to the remaining number of public interfaces on the node.

```
Total number of interfaces (4)
- Number of private interfaces (2)
= Number of public interfaces

4 - 2 = 2
```

To calculate the physical and the virtual IP addresses for the cluster

1 The total number of physical IP addresses that are required for the cluster installation is equal to the number of nodes in the cluster multiplied by the number of public interfaces on each node:

```
Total number of nodes (2)
x Number of public interfaces per node (2)
= Total number of physical IP addresses

= 2 x 2 = 4
```

2 The number of nodes in the cluster multiplied by the number of public interfaces on each node is equal to the total number of virtual IP addresses that are required for the cluster installation:

```
Total number of nodes (2)
x Number of public interfaces per node (2)
= Total number of virtual IP addresses

= 2 x 2 = 4
```

3 The number of IP addresses required for the Veritas Access Operations Manager is equal to one (1).

To calculate the total number of public IP addresses for the cluster

- The number of physical IP addresses for the cluster, plus the number of virtual IP addresses for the cluster, plus the number of IP addresses for the Operations Manager is equal to the total number of public IP addresses that are required for the cluster.

```
Total number of physical IP addresses/cluster (4)
+ Total number of virtual IP addresses/cluster (4)
+ Number of IP addresses for the Management Console (1)
= Total number of public IP addresses required for the cluster

= 4 + 4 + 1 = 9
```

To request and specify IP addresses

1. Request the public IP addresses that you need from your Network Administrator.
2. For example, if the Network Administrator provides you with IP addresses `10.209.105.120` through `10.209.105.128`, you can allocate the resources in the following manner:

   ```
   Start of Physical IP address: 10.209.105.120
   Start of Virtual IP address: 10.209.105.124
   Management Console IP:"10.209.105.128"
   ```

   This entry gives you four physical IP addresses (`10.209.105.120` to `10.209.105.123`), four virtual IP addresses (`10.209.105.124` to `10.209.105.127`), and one IP address for the Operations Manager (`10.209.105.128`).

   `10.209.105.120` and `10.209.105.121` are assigned to pubeth0 and pubeth1 as physical IP addresses on the first node.

   `10.209.105.122` and `10.209.105.123` are assigned to pubeth0 and pubeth1 as physical IP addresses on the second node.

   `10.209.105.124` to `10.209.105.127` are assigned to pubeth0 and pubeth1 as virtual IP addresses on the two nodes.

Reducing the number of IP addresses required at installation time

You can reduce the number of IP addresses required at installation time by not configuring any virtual IP addresses. During the Veritas Access installation, input 0 for the number of virtual IP addresses per NIC.

Virtual IP addresses are not required at installation time. You can configure the virtual IP addresses later using the `Network> ip addr add` command in the Veritas Access command-line interface.

See the `network(1)` manual page for more information on adding NICs.

You need at least one IP address per node per public interface at installation time.

Table 3-5 Example configuration of required IP addresses at installation time for a two-node cluster with two public NICs per node

Number of IP addresses	Item
4	Number of physical IP addresses. The four IP addresses include the original physical IP addresses.

Table 3-5 Example configuration of required IP addresses at installation time for a two-node cluster with two public NICs per node *(continued)*

Number of IP addresses	Item
1	One IP address for the management console.

About checking the storage configuration

Warning: Do not connect the Fibre Channel HBAs until you finish installing the operating system. If the local disks are bad, connecting the Fibre Channel HBAs prevents the operating system from being installed on the local disks. Because the disk is scanned, it takes longer to install the software on a local disk.

Veritas Access supports Flexible Storage Sharing (FSS), which allows the users to configure and manage direct-attached storage on the Veritas Access appliance.

After you install the operating system, check the storage configuration. If you don't want to use FSS, make sure that each node has the following:

- One or two Fibre Channel Host Bus Adapters (HBAs) for connection to the Storage Area Network (SAN) switch.

  Two Fibre Channel HBAs are recommended, but only one is required. Having only one Fibre Channel HBA enables all the operations of the Fibre Channel (except high availability).
- An internal boot disk. Make sure that one is in place before you install the Veritas Access software.

If you want to use FSS, make sure that each node has attached at least two extra local data disks besides the internal boot disk.

Chapter 4

Deploying virtual machines in VMware ESXi for Veritas Access installation

This chapter includes the following topics:

Setting up networking in VMware ESXi

Before you start, install the ESXi server. You can deploy the first virtual machine on your ESXi host by using the vSphere Client.

To set up a network in VMware ESXi

1. Start the vSphere Client and type the logon details for your host.

   In the **IP address/Hostname** box, enter the **ESXi server IP/hostname**.

   In the **User name** box, type **root**.

   In the **Password** box, type ***my_esxi_password***.
2. Set up the networking requirements for Veritas Access.
3. To set up the public network virtual switch:
   - In the **Configuration** tab of the ESXi host, go to **Hardware > Networking**.
   - On the upper right corner, click **Add Networking**.

- Select the connection type as **Virtual Machine** and click **Next**.
- Under the **Create a virtual switch**, select a NIC that is connected to the public network.
- Enter the appropriate network label for the public virtual switch.
- Verify the summary and click **Finish**.

Note: If you want to create multiple public network switches, repeat the preceding steps.

---

4 To set up the private network virtual switch:

- In the **Configuration** tab of the ESXi host, go to **Hardware > Networking**.
- Click **Add Networking** on the top right corner.
- Select the connection type as **Virtual Machine** and click **Next**.
- Deselect any NIC that is selected by default for creating the virtual switch.
- Enter a label for the private virtual switch.
- Verify that the summary shows no-adapters under the physical adapters, and click **Finish** to create the first private network virtual switch.

Note: If you want to create the second private network virtual switch, repeat the preceding steps.

Creating a datastore for the boot disk and LUNs

To create a datastore for the boot disk and LUNs

1 Create a datastore for `vmdk` files for virtual machines.

2 In the **Configuration** tab of the ESx host, navigate to **Hardware > Storage**.

3 Click **Add Storage** on the top right corner.

4 Select the storage type as **Disk/LUN** and click **Next**.

5 Select the disk that you want to use to create the virtual machine vmdk files.

6 Review the current disk layout and click **Next**.

7 Enter the datastore name of your choice and click **Next**.

8 Select the disk space that you want to dedicate for the datastore. The default option is to use the complete list.

9 Review the details and click **Finish**.

Creating a virtual machine for Veritas Access installation

To create a virtual machine for Veritas Access installation

1 After the networking configuration is complete and the datastore is defined, create the virtual machines.

- Select the **ESXi host IP/hostname** in the top of the tree structure in the upper left frame.
- From the **File** menu, select **New Virtual Machine**.
  The dialog box for creating the virtual machine is displayed.
- Select the configuration as **Custom** and click **Next** to decide on the exact configuration of the virtual machine.
- Enter the virtual machine name of your choice and click **Next**.
- Select the datastore that stores the virtual machine `vmdk` file and click **Next**.
- Select the virtual machine version that you want to use and click **Next**. Veritas recommends version 8.
- Select the guest operating system as **Linux** and version as **Red Hat Enterprise Linux 6 or 7 (64-bit)** and click **Next**.

Select the number of CPUs. Veritas recommends eight cores that can be:

- Two virtual sockets and four cores per virtual socket.
- One virtual socket and eight cores per virtual socket.
- Any higher number of cores as per your workload.
  Select the memory configuration. Veritas recommends 32 GB of size for the memory configuration.
- In the network configuration, Veritas recommends to select the number of NICs as four.
  For NIC 1 and NIC 2, select the public network virtual switch and validate that the adapter is correct.
  For NIC 3, select the private network virtual switch 1 and validate that the adapter is correct.

For NIC 4, select the private network virtual switch 2 and validate that the adapter is correct.

- Select the SCSI controller as **VMware Paravirtual**.
- On the **Disk Configuration** page, select **Create a new virtual disk** and click **Next**.
- Select the boot disk size. Veritas recommends 100 GB of size for the boot disk.
- Select the disk provisioning type as **Thick Provision Eager Zeroed**.
- Select the datastore as **Specify a data store or data store cluster** and click **Next**.
  After selecting the datastore, click **Next**.
- Select the **Virtual device node** as default (SCSI (0:0) for the boot disk) and click **Next**.
- Review the virtual machine configuration and click **Finish** to create the virtual machine.
  The virtual machine creation task is complete.

2 Select the virtual machine and click **Edit virtual machine settings** to validate the following:

- There should be four network adapters, that is, two for the public network and two for the private network.
- Verify that the memory and CPU configuration is correct.

3 Repeat the step 1 and 2 to create the second virtual machine, which is used to form the two-node Veritas Access cluster.

4 Add LUNs/DAS disks to the virtual machines.

To add local DAS disks:

- Select the virtual machine and click **Edit virtual machine** settings.
- Click **Add**.
- Select **Hard Disk** as device type and click **Next**.
- In the type of disk list, click **Create a new virtual disk** and click **Next**.
- Select a size for the DAS disk. Veritas recommends 100 GB of size for the DAS disk.
- Select **Thick Provision Eager Zeroed** as disk provisioning type.
- Select the datastore as **Specify a data store or data store cluster** and click **Next**.

- Select the **Virtual device node** as SCSI (1:0) for the first SAS disk and click **Next**.

  After all the required DAS disk are created, complete the following:

  - Select the SCSI controller 1 that is used for DAS disks.
  - Set the **SCSI Bus sharing** mode as **Virtual.**
    This mode is required so that DAS disks are claimed in VxVM enclosure-based naming (EBN) mode and host name is only prefixed by VxVM when disks are in EBN mode, which distinguishes it from the shared LUNs present in the arrays.

- Click **OK** to create the DAS disk.
  Repeat step 4 for creating the DAS disk for other Veritas Access nodes.

5 Map the shared disks to the LUNs.

Mapping of LUNs from an array is only supported by using Raw Device Mapping (RDM) mode.

To map the shared LUNs to the first virtual machine:

- Select the first virtual machine and click **Edit virtual machine settings**.
- Click **Add**.
- Select **Hard Disk** as device type and click **Next**.
- Select the LUN that you want to map and click **Next**.
- Select the datastore that stores the LUN mapping or select **Store with virtual machine**.
- Select **Physical** as compatibility mode to access the array LUN hardware directly.
- Select the **Virtual device node** as SCSI (2:0) for the shared disk and click **Next**.
- Review the mapping of the disk and click **Finish** to map the array LUN disk to the virtual machine.
  Repeat step 5 for the number of LUNs that you want to map and update the **Virtual device node** to the next free SCSI controller port.

After all the required LUNs are mapped, do the following:

- Select the SCSI controller 2 that is used for shared LUNs.
- Set the **SCSI Bus sharing** mode as **Virtual**.
  This mode is required so that the shared LUNs are claimed in VxVM EBN mode. This distinguishes it from the shared LUNs present in the arrays.
- Click **OK** to complete the mapping of LUNs in RDM mode.

Mapping shared LUNs to the second virtual machine:

- Select the first virtual machine and click **Edit virtual machine settings**.
- Click **Add**.
- Select **Hard Disk** as device type and click **Next**.
- Select **Use an existing Virtual Disk** in the type of disk and click **Next**.
- Navigate to the corresponding disk path in the datastore where the shared disk was stored when they were mapped to the first virtual machine.
- Select the **Virtual device node** as SCSI (2:0) for the shared disk and click **Next**. Ensure that the sequence of disk mapping is the same as that of the first virtual machine and mapping has been done to the same SCSI controller to achieve a shared disk configuration.
- Review the mapping of the disk and click **Finish** to map the array LUN disk to the virtual machine.
  Repeat this Step for the number of shared LUNs that you have mapped to other virtual machines and update the **Virtual device node** to the next free SCSI controller port.

After all the required LUNs are mapped, complete the following:

- Select the SCSI controller 2, which is used for the shared LUNs.
- Set the **SCSI Bus sharing** mode as **Virtual**.
  This mode is required so that the shared LUNs are claimed in VxVM EBN mode. This distinguishes it from the shared LUNs present in the arrays.
- Click **OK** to complete the mapping of LUNs in RDM mode.
  The networking and storage configuration is complete for the virtual machines.

6 Install the RHEL 7 Update 3 or 4 (64-bit) operating system that is supported by the Veritas Access installer.

See "Installing the RHEL operating system on the target Veritas Access cluster" on page 51.

Note: The virtual machine can have DAS disks, shared LUNs, or both of them. For the erasure coded file system, the disks should be DAS only.

Chapter 5

Installing and configuring a cluster

This chapter includes the following topics:

Installation overview

You can install the Veritas Access on a cluster. You can add a minimum of one-node and a maximum of 20 nodes to the cluster. By adding a single node or multiple nodes to the cluster, you can make the system fault-tolerant and scale it up as required.

Summary of the installation steps

The Veritas Access software installation consists of two main pieces:

- Operating system installation.
  Veritas Access requires Red Hat Enterprise Linux.
  See "System requirements" on page 14.
- Veritas Access software installation.

Table 5-1 provides a brief summary of the installation steps. The summary includes cross references to where you can find more information about each task.

Table 5-1 Summary of installation steps

Task	Steps	For more information
Task 1: Install the operating system on each node of the cluster.	Steps include: ■ Automatic system discovery of USB devices, hard disk controllers, and so on. ■ Select the installation device. ■ Set the clock and the time zone. ■ System preparation for automated installation. ■ Manual disk partitioning. ■ Minimal package installation. ■ Install the Red Hat Enterprise Linux kernel update.	See "Installing and configuring the Veritas Access software on the cluster" on page 53.

Table 5-1 Summary of installation steps *(continued)*

Task	Steps	For more information
Task 2: Install the Veritas Access software on the cluster.	Steps include: ■ Install the required Red Hat Enterprise Linux operating system RPMs. If yum is configured, then the installer helps to install the required RPMs during the precheck. ■ Download the release update from the Download Center on the Veritas Support website ■ Extract the Veritas Access tar file and run the installer. ■ Enter network configuration information (cluster name, IP addresses, DNS information, and so on). ■ Verify installation on the node.	See "Installing and configuring the Veritas Access software on the cluster" on page 53.

Before you install

Before you install the Veritas Access software:

- Download the release update from the Veritas support website:
  - Open a web browser and type support.veritas.com. Click **Sign In** next to the **User** icon to log into your Veritas Account.
  - Enter your Veritas Account credentials and click **Sign In**.
  - Click **Downloads**.
  - In the **Product** and the **Sub-product** list, select **Access**. In the **Version** list, select 7.4.2.400.
  - Click **Updates** and download the release update.
- Make sure that the names of all the public and private interface which are targeted to be part of access configuration during installation should be same

across all the Veritas Access cluster nodes on which you want to install Veritas Access.

- Before you install Veritas Access, if the system has a network interface bond , you need to specify the bond names as bond0, bond1, bond2 and so on, and then you can start the Veritas Access installation.
- Before you install Veritas Access, if the system has VLAN configured network interface on it and if you want to configure it on the Veritas Access nodes, the interface name on which VLAN is configured must follow the naming format as follows:
  <interface_name>.<vlan_id>. For example, eth0.100 or ens168.101.
- Make sure that no DHCP servers are running in the private network.
- Disable the USB Ethernet interface in BIOS for all nodes in the cluster.
- Make sure that there are at least two private connection between the cluster and the public connection. The public connected interface must be reachable to gateway.
- Connect DAS or SAN storage to the cluster node. If you are configuring the fencing feature to avoid the split brain condition. Make sure that the SAN disks are SCSI3 compliant.
- Assign one public IP address to each node in a cluster. This IP address is used by the installer, and it is reused as one of the public interface physical IP address after configuration.
- Make sure that the assigned IP is persistent after reboot. To make the IP persistent, do following changes in
  `/etc/sysconfig/network-scripts/ifcfg-XX`
  For example:

```
TYPE=Ethernet
HWADDR=00:50:56:3d:f1:3e
DEVICE=eth2
BOOTPROTO=none
IPADDR=10.200.56.214
NETMASK=255.255.252.0
NM_CONTROLLED=no
ONBOOT=yes
```

Note: Make sure that you have enough public IPs that are required to install the Veritas Access software.

Installing the operating system on each node of the cluster

Before you install the Veritas Access software, you must install the RHEL OS and kernel version. The following procedure includes the instructions and download links.

To install the RHEL OS on each node of the cluster

1. Meet the requisite system requirements. Ensure that you have the required version of the RHEL OS and the kernel version.
2. Use the following information to install RHEL OS:
   - Refer to the *Chapter 1. Obtaining Red Hat Enterprise Linux* in the *Red Hat Enterprise Linux 7 Install guide*:
     https://access.redhat.com/downloads/
     https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/pdf/installation_guide/Red_Hat_Enterprise_Linux-7-Installation_Guide-en-US.pdf

About the driver node

If you do not plan to install Veritas Access from the console of the nodes in the cluster (the local management console of your nodes), you need another server that is not a target node in the Veritas Access cluster to use in the Veritas Access installation. This server is called the driver node.

When you run the Veritas Access installation script, the Veritas Access installer helps set up the ssh connection between the driver node and the target Veritas Access cluster nodes.

The driver node platform can be: RHEL 7, SLES 11 SP2, or SLES 11 SP3.

Table 5-2 provides the information about Veritas Access installation support from the cluster node and the driver node with different type of network devices.

Table 5-2 Installation support from the cluster node and the driver node

Network device type	Driver Node	Cluster Node
Normal network device	Yes	Yes
Create Bond device through installer and add NIC in bond through which installation is started	Yes	No
Create bond device on NIC other than the NIC through which installation is started	Yes	Yes

Table 5-2 Installation support from the cluster node and the driver node *(continued)*

Network device type	Driver Node	Cluster Node
Create VLAN through installer on NIC other than NIC through which installation is started	No	Yes
Create VLAN through installer on NIC through which installation is started	No	No
Exclude NIC from which installation started	No	No
Create bond and VLAN over bond device on NIC other that NIC through which installation is started	No	Yes
Pre-configured bond as public and Installation from other NIC	Yes	Yes
Create a bond through installer and select it as private connection	No	No
Create VLAN through installer and select it as private connection	No	No
Installation with public NIC and pre-existing public bond	Yes	Yes

Installing the RHEL operating system on the target Veritas Access cluster

You need to install the RHEL OS on each node of the cluster.

To install the RHEL OS

1. Insert the RHEL 7 update 7 or 9 operating system installation DVD, and boot the server from the DVD.

   See “Linux requirements” on page 16.

   You can also use an external USB DVD-ROM.
2. Select **Install Red Hat Enterprise Linux 7 Update 7 or Update 9**.
3. After the system loads, select the **language for installation** as **English** and click **Continue**.

   After the installer displays Installation Summary, you can customize the installation process.
4. Click **Date & Time**, choose your system location from the provided map, and then click **Done** to apply configuration.

5. Select **English** language for the Language System Support and the Keyboard language.
6. To select your system software, click **Software Selection** and select a base installation environment from the list.
7. Select **Minimal Install** with **Compatibility Libraries Add-ons**, and then click **Done** to apply this changes to the installation process.
8. Select a disk and perform disk partition manually to configure the system partitions.
9. Click **Network & Hostname** and provide a system hostname to set up your network connection.

   After you set up the hostname, set the Ethernet to On to bring your network interface up. Click **Configure** and provide your static network settings for your appropriate network connection.
10. After you finish editing the Ethernet Interface settings, click **Done**.

    The default installer window is displayed.
11. Verify the installation settings, and then click **Begin Installation** to start the system installation.
12. As the installation begins writing the system components on your hard-disk, you need to provide your root password. Click **Root Password**, enter the password, and the click **Done** when you finish.

    After the installation is finished, the installer displays details of the successful installation.You can follow the same steps to install the RHEL OS on other nodes of the cluster.

    See the *Red Hat Enterprise Linux documentation* for the detailed procedure.

Installing Veritas Access on the target cluster nodes

Before you install Veritas Access on the target cluster nodes, you must allocate enough IP addresses for the cluster. You can install up to a 20-node cluster.

Installing the cluster is a one-time activity. It takes about 40 minutes to install a two-node cluster. Installation times may vary depending on your configuration and the number of nodes.

If you want to install the Veritas Access cluster with IPv6 IP addresses, you need to configure a static IPv6 address on the driver node and on all the nodes of the cluster. You need to make sure that the IPv6 IP auto-assignment is disabled on all

nodes of the cluster. You can then use the IPv6 IPs to install Veritas Access on the cluster nodes.

If you want to configure the cluster in an IPv6 environment or use the cluster in mixed mode by configuring both IPv4 and IPv6 IPs, you have to disable the IPv6 IP auto-assignment.

To configure a static IPv6 address

1. Modify the `vim /etc/sysconfig/network-scripts/ifcfg-ens161` network interface file by using the following:

   ```
   IPV6INIT="yes"
   IPV6_AUTOCONF="no"
   IPV6ADDR=2001:128:f0a2:900a::121/64
   ```

2. Restart the network service by using the`systemctl restart network` command.

Installing and configuring the Veritas Access software on the cluster

To install and configure the cluster

Note: During the installation, the installer log is located at `/var/tmp`.

---

1. Log on as superuser and mount the Veritas Access 7.4.2.400 installation media. From the root, enter the following command to start the installation.

   ```
   # ./installaccess node1_ip node2_ip
   ```

   Where *node1_ip* and *node2_ip* are the public physical IP addresses that are already assigned to the target cluster nodes to install Veritas Access over SSH.

   These are the current IPs assigned to the nodes for the installation communication.

   The example is used to install two nodes. To install another target node cluster, add *node3_ip* to the command line that is used in this step.

2. The installer checks for the operating system dependencies and automatically installs the required OS RPMs. If the OS RPMs' dependencies are not sorted, the Red Hat subscription manager user id and password is required.

3. The installer installs the Veritas Access RPMs.

4 Choose the licensing method. Answer the licensing questions and follow the prompts.

```
1)  Enter a valid perpetual or subscription license key file
2)  Register with evaluation mode and complete system licensing later

How would you like to license the systems? [1-2,q,?] (2)
```

5 The installer displays the firewall ports to be opened after the configuration, and asks if you want to open them:

```
Veritas Access needs to open the following ports:
111 Rpcbind (NFS)
11211 Memcached Port
123 NTP Service
139 CIFS Service
14161 GUI
161 SNMP Service
2049 NFS Service
21 FTP Port
22 SSH Service
25 SMTP Port
30000:40000 FTP Passive Port Range
3172,3173 Server View Ports
4001 Mountd (NFS)
4045 NLM (NFS)
4379 CTDB Port
445 CIFS TCP Service
51001,51002 RDMA Service
514 Syslog Service
53 DNS Service
5634 VIOM
56987 Replication Service
756,757,755 Statd (NFS)
8088 REST Server
8143 Object Access Gateway
8144 Object Access Admin Gateway
Do you want to proceed? [y,n,q] (y)
```

The installer automatically configures the RDMA environment on the cluster nodes if there are InfiniBand NICs.

6 The installer asks the following information to configure the cluster:

```
Enter the cluster name: [q,?]
Do you want to rename the hosts' name like vac-01, vac-02? [y,n,q,b,?] (n)
Enter the public IP starting address or : [b,q,?]
Enter the netmask for the public IP address: [b,q,?] (255.255.255.0)
Enter the number of VIPs per interface: [b,q,?] (0) 1
Enter the virtual IP starting address: [b,q,?]
Enter the default gateway IP address: [b,q,?]
Enter the DNS IP address: [b,q,?] (10.0.2.3)
Enter the DNS domain name: [b,q,?] (community.veritas.com)
Enter the console virtual IP address: [b,q,?]
Do you want to use the separate console port? [y,n,q,b,?] (n)
Enter the private IP starting address: [b,q,?] (172.16.0.3)
```

Note: Cluster names should be DNS-compatible. Cluster name must be at least three character and no more than 63 characters long. Allowed characters in a cluster name are ‘a-z, 0-9, -‘ (lowercase letters, numbers, and hyphens). Any other character is invalid. Also, if a separate console port is chosen, the first public NIC is chosen to work exclusively as a console port.

Starting with 7.4.2.400, underscore (_) is not allowed in a cluster name. You can however upgrade to 7.4.2.400 from an earlier version where the cluster name included an underscore.

7 The installer asks if you want to configure the Network Time Protocol (NTP) server.

```
Do you want to configure the Network Time Protocol(NTP) server to
synchronize the system clocks? [y,n,q]  y
Enter the Network Time Protocol server: [q,?]
```

If you enter **y**, you can type in your NTP server. If you enter **n**, the NTP server is not configured.

8 Installer asks to confirm the entered cluster information.

The installer detects the network devices, checks the network device's connectivity with gateway, and displays information about it.

```
Checking network configuration ..................................... Done
Detecting network devices .......................................... Done
Checking network connection ........................................ Done

Detecting network devices completed successfully.

Common NICs on all systems:

NIC       Type       Properties     Public
=========================================
ens161    Physical   -              N
ens192    Physical   -              Y
ens193    Physical   -              Y
ens224    Physical   -              Y
ens256    Physical   -              N

For the 'Public' field of the NIC:
    Y: means the NIC can connect to the public gateway, and can be selected as
public NIC.
    N: means the NIC cannot connect to the public gateway, and can be selected
as private NIC.
    -: means the NIC was not tested if connect to the public gateway.
    blank: means this NIC is excluded or not selectable.
```

To configure Veritas Access networking, you need to exclude the unused NICs, and to include at least one public NIC and one private NIC. Veritas recommends to have two public NICs and two private NICs, and the selected private NICs on all systems should be interconnected.

If you want to configure NIC bonding or exclude NICs, enter **y**.

If you do not want to configure NIC bonding or exclude NICs, enter **n**. Go to step To install and configure the cluster .

See “Excluding a NIC” on page 67.

See “Creating a NIC bond” on page 75.

See “Creating a VLAN device ” on page 87.

9 You need to select one of the option from the following for the installation:

- Manually select NIC

- Configure NIC bonding
- Configure VLAN through installer

```
Do you want to manually select NICs, or configure NIC bonding or VLAN tagging? [y,n,q] (n)
Enter n : If you want installer to do auto network configuration for the cluster
Enter y : If you want to select public and private NICs manually, configure NIC bonding
          or to create VLAN using installer.
```

The installer performs the public and private NIC detection tests on each node of cluster, if physical or virtual IPs that are entered are less than the required IPs for the cluster configuration, the installer asks you to add the required IPs.

10 Verify that the network configuration details such as the new IP addresses, hostname, and other details are correct.

11 The installer prompts to verify the network configuration.

Verify that the configuration information such as the new IP addresses, host name, and other details are correct.

```
Configuration checklist:

System        Public NIC Physical IP
=======================================
192.168.10.10 ens192     192.168.10.20
192.168.10.10 ens224     192.168.10.21
192.168.10.10 ens193     192.168.10.22
192.168.10.11 ens192     192.168.10.23
192.168.10.11 ens224     192.168.10.24
192.168.10.11 ens193     192.168.10.25

System        Private NIC
=========================
192.168.10.10 ens161
192.168.10.10 ens256
192.168.10.11 ens161
192.168.10.11 ens256

Virtual IP
============================================================================
192.168.10.30 192.168.10.31 192.168.10.32 192.168.10.33 ...(6 in total)

Console IP
============
192.168.10.50

Gateway IP      DNS IP       Domain name
=============================================
192.168.10.1    192.168.10.2 vxindia.veritas.com

Is this information correct? [y,n,q,b,?] (y)
```

12 After you confirm the network configuration details, the installer renames the hostname if you have chosen to rename it and assigns the IPs for the systems. The installer also checks the Low Latency Transport (LLT) link status and automatically selects them.

Note: The installer does not check the LLT link status if the InfiniBand NICs are chosen as private NICs. See "About using LLT over the RDMA network for Veritas Access" on page 28.

13 Installer performs the Veritas Access service group configuration.

14 If the cluster has shared storage (shared disks), you are prompted to configure either disk-based or majority-based I/O fencing. Enter any one of the options to configure I/O fencing.

If the cluster does not have shared storage, majority-based I/O fencing is configured automatically. You are not prompted to specify your option.

15 The installer automatically restarts the cluster nodes to enables the Kdump function for each node.

16 Check the log file to confirm the installation and configuration. Logs can be found in `/opt/VRTS/install/logs/`.

Note: After the installation, connect to the Veritas Access console using the console IP address you assigned earlier, then log on using the default user name `master` and the default password `master`.

Veritas Access Graphical User Interface

Veritas Access has a Graphical User Interface (GUI) that provides a dashboard for a specific Veritas Access cluster, as well as views for shares, storage infrastructure, reports, and settings. The GUI lets the administrator perform tasks for the cluster and monitor the results. In this release, the GUI is part of Veritas Access.

After you complete I/O fencing configuration successfully, the link to the GUI appears on the screen.

```
Open the https://<console IP>:14161 URL
in your browser to start the Veritas Access GUI application.
```

About managing the NICs, bonds, and VLAN devices

When you enter **y**, the installer allows you to perform the following operations:

```
Do you want to manually select NICs, or configure NIC bonding or VLAN tagging? [y,n,q] (n) y
```

- Select the public NICs
  See "Selecting the public NICs" on page 61.
- Select the private NICs
  See "Selecting the private NICs" on page 64.
- Exclude a NIC
  See "Excluding a NIC" on page 67.
- Include a NIC
  See "Including a NIC" on page 71.
- Create a new NIC bond and add a NIC to a bond
  See "Creating a NIC bond" on page 75.
- Remove a bond
  See "Removing a NIC bond" on page 81.
- Remove a NIC from the bond list
  See "Removing a NIC from the bond list" on page 84.
- Add a VLAN device on a particular NIC
  See "Creating a VLAN device " on page 87.
- Remove a VLAN device on a particular NIC
  See "Removing a VLAN device " on page 90.

Note: The NIC bonding and NIC exclusion configuration options support both a single NIC or bond, and multiple NICs or bonds.

When using the NIC exclusion feature, you can exclude any NIC on the first node. But if you want to exclude any NIC on the other nodes, you can choose to exclude NICs per node.

See "Excluding a NIC" on page 67.

Selecting the public NICs

When you install Veritas Access on a cluster, you may want to configure the specific network devices as public interface even though they are not reachable to gateway and specific network devices as a private interface.

To select the public NICs

1 In the manual selection mode, enter **1** to select public NICs.

```
NIC selection/NIC bonding/NIC VLAN configuration

Common NICs on all systems:

NIC         Type       Properties       Public
=================================================
ens161      Physical   -                N
ens192      Physical   -                Y
ens193      Physical   -                Y
ens224      Physical   Slave of bond0
ens225      Physical   Slave of bond0
ens256      Physical   -                N
ens257      Physical   -
bond0       Virtual    Bond balance-rr  Y
ens192.100 Virtual    VLAN 100         -

     1)  Select public NICs
     2)  Select private NICs
     3)  Select a NIC to host the Management Console IP
     4)  Select system administration management NICs
     5)  Exclude NICs
     6)  Include NICs
     7)  Create a new bond
     8)  Add NICs to a bond
     9)  Remove bonds
    10)  Remove NICs from the bond list
    11)  Create VLAN device
    12)  Delete VLAN device
    13)  Reset selections for all NICs
    14)  Save and continue

Select the NIC option to be configured in this cluster: [1-14,q] 1
```

2 Select the NIC that you want to choose as public NICs.

```
Choose NICs as public

     1)  ens161
     2)  ens192
     3)  ens193
     4)  ens256
     5)  bond0
     6)  ens192.100
     7)  Unselect previous public NICs
     b)  Back to previous menu

Choose items, separated by spaces: [1-7,b,q] 2 4 5 6
NIC selection/NIC bonding/NIC VLAN configuration

Common NICs on all systems:

NIC         Type       Properties       Public
=================================================
ens161      Physical   -                N
ens192      Physical   -                Y (Selected)
ens193      Physical   -                Y
ens224      Physical   Slave of bond0
ens225      Physical   Slave of bond0
ens256      Physical   -                Y (Selected)
ens257      Physical   -
bond0       Virtual    Bond balance-rr  Y (Selected)
ens192.100 Virtual    VLAN 100         Y (Selected)

     1)  Select public NICs
     2)  Select private NICs
     3)  Select a NIC to host the Management Console IP
     4)  Select system administration management NICs
     5)  Exclude NICs
     6)  Include NICs
     7)  Create a new bond
     8)  Add NICs to a bond
     9)  Remove bonds
    10)  Remove NICs from the bond list
    11)  Create VLAN device
    12)  Delete VLAN device
    13)  Reset selections for all NICs
```

```
    14)  Save and continue

Select the NIC option to be configured in this cluster: [1-14,q]
```

Note: To start the cluster configuration, after the manual public and private NIC selection or configuration, enter **14** to select the `Save and continue` option.

Selecting the private NICs

When you install Veritas Access on a cluster, you may want to configure the specific network devices as private interface.

To select the private NICs

1 In the manual selection mode, enter **2** to select private NICs.

```
NIC selection/NIC bonding/NIC VLAN configuration

Common NICs on all systems:

NIC         Type       Properties       Public
=================================================
ens161      Physical   -                 N
ens192      Physical   -                 Y (Selected)
ens193      Physical   -                 Y (Selected)
ens224      Physical   Slave of bond0
ens225      Physical   Slave of bond0
ens256      Physical   -                 N
ens257      Physical   -
bond0       Virtual      Bond balance-rr  Y (Selected)
ens192.100 Virtual   VLAN 100         Y (Selected)

     1)  Select public NICs
     2)  Select private NICs
     3)  Exclude NICs
     4)  Include NICs
     5)  Create a new bond
     6)  Add NICs to a bond
     7)  Remove bonds
     8)  Remove NICs from the bond list
     9)  Create VLAN device
    10)  Delete VLAN device
    11)  Save and continue

Select the NIC option to be configured in this cluster: [1-11,q]  2
```

2 Select the NIC that you want to choose as private NICs.

```
Choose NICs as private

     1)  ens161
     2)  ens192
     3)  ens193
     4)  ens256
     5)  Unselect previous private NICs
     b)  Back to previous menu

Choose items, separated by spaces: [1-5,b,q] 1 4

NIC selection/NIC bonding/NIC VLAN configuration

Common NICs on all systems:

NIC         Type       Properties       Public
=====================================================
ens161      Physical   -                N (Selected)
ens192      Physical   -                Y (Selected)
ens193      Physical   -                Y (Selected)
ens224      Physical   Slave of bond0
ens225      Physical   Slave of bond0
ens256      Physical   -                N (Selected)
ens257      Physical   -
bond0       Virtual    Bond balance-rr  Y (Selected)
ens192.100 Virtual    VLAN 100         Y (Selected)

     1)  Select public NICs
     2)  Select private NICs
     3)  Exclude NICs
     4)  Include NICs
     5)  Create a new bond
     6)  Add NICs to a bond
     7)  Remove bonds
     8)  Remove NICs from the bond list
     9)  Create VLAN device
    10)  Delete VLAN device
    11)  Save and continue

Select the NIC option to be configured in this cluster: [1-11,q]
```

Note: To start the cluster configuration, after the manual public and private NIC selection or configuration, enter **11** to select the `Save and continue` option.

Excluding a NIC

When you install Veritas Access on a cluster, you may want to use some of the NICs for other storage purposes. You can exclude a NIC that you do not want to use for Veritas Access.

Note: The NIC bonding/NIC exclusion configuration options support both a single NIC or bond, and multiple NICs or bonds.

To exclude a NIC

1 During Veritas Access installation, the installer asks if you want to configure NIC bonding or exclude NICs. Enter **y** if you want to exclude a NIC.

```
Do you want to configure NIC bonding or exclude NICs or configure VLAN
tagging? [y,n,q] (n)
```

2 The installer prompts you to enter your selection. Enter **1** to exclude a NIC.

```
Veritas Access 7.4.2.400 Configure Program
10.200.114.45 10.200.114.46

NIC bonding/NIC exclusion configuration

NIC bonding supports only public NICs. Make sure the NICs you choose
are connected to public network.

NIC  PCI ID       bond status    If excluded
=============================================
eth2 0000:02:03.0 (physical NIC)  N
eth3 0000:02:04.0 (physical NIC)  N
eth4 0000:02:05.0 (physical NIC)  N
eth5 0000:02:06.0 (physical NIC)  N
eth6 0000:02:07.0 (physical NIC)  N
eth7 0000:02:08.0 (physical NIC)  N

     1)  Exclude NICs
     2)  Include NICs
     3)  Create a new bond
     4)  Add NICs to a bond
     5)  Remove bonds
     6)  Remove NICs from the bond list
     7)  Create VLAN device
     8)  Delete VLAN device
     9)  Save and Continue

Select the NIC option to be configured in this cluster: [1-9,q] 1
```

3 The installer prompts you to select the NIC that you want to exclude. Enter your choice.

```
Choose NICs for exclusion

     1)  eth2 0000:02:03.0 (physical NIC)
     2)  eth3 0000:02:04.0 (physical NIC)
     3)  eth4 0000:02:05.0 (physical NIC)
     4)  eth5 0000:02:06.0 (physical NIC)
     5)  eth6 0000:02:07.0 (physical NIC)
     6)  eth7 0000:02:08.0 (physical NIC)
     7)  Exclude NICs per node
     b)  Back to previous menu

Choose NICs: [1-7,b,q] 1 2(1,2)
```

4 The installer goes back to the previous menu. You can choose another NIC for exclusion. Enter **1** to exclude another NIC. Or you can save your configurations and continue with the installation of Veritas Access.

If you want to save your configurations, enter **9** :

```
Veritas Access 7.4.2.400 Configure Program
10.200.114.45 10.200.114.46

NIC bonding/NIC exclusion configuration

NIC bonding supports only public NICs. Make sure the NICs you choose
are connected to public network.

NIC  PCI ID       bond status    If excluded
=============================================
eth2 0000:02:03.0 (physical NIC)  Y
eth3 0000:02:04.0 (physical NIC)  Y
eth4 0000:02:05.0 (physical NIC)  N
eth5 0000:02:06.0 (physical NIC)  N
eth6 0000:02:07.0 (physical NIC)  N
eth7 0000:02:08.0 (physical NIC)  N

     1)  Exclude NICs
     2)  Include NICs
     3)  Create a new bond
     4)  Add NICs to a bond
     5)  Remove bonds
     6)  Remove NICs from the bond list
     7)  Create VLAN device
     8)  Delete VLAN device
     9)  Save and Continue

Select the NIC option to be configured in this cluster: [1-9,q] 9
```

5 If you want to exclude NICs per node, in Step 2 enter **7**. The NICs with inconsistent PCI IDs are listed:

```
Choose NICs for exclusion

     1)  eth2 0000:02:03.0 (physical NIC)
     2)  eth3 0000:02:04.0 (physical NIC)
     3)  eth4 0000:02:05.0 (physical NIC)
     4)  eth5 0000:02:06.0 (physical NIC)
     5)  eth6 0000:02:07.0 (physical NIC)
     6)  eth7 0000:02:08.0 (physical NIC)
     7)  Exclude NICs per node
     b)  Back to previous menu

Choose NICs: [1-7,b,q] 7

Choose items: [1-1,b,q] 1
     1    0000:02:00.0 (10.198.95.214)
     2    0000:02:01.0 (10.198.95.214)
     3    0000:02:06.0 (10.198.95.212)
     4    0000:02:09.0 (10.198.95.214)
     5    0000:02:14.0 (10.198.95.212)
     6    0000:02:15.0 (10.198.95.212)
     b)  Back to previous menu
Choose NICs: [1-6,b,q] 1 2 3 4 5 6
```

Note: NIC exclusion function is supported on InfiniBand NICs, but all the NICs with the same PCI ID are excluded during the exclusion operation. See “About using LLT over the RDMA network for Veritas Access” on page 28.

Including a NIC

When you install Veritas Access on a cluster, you may want to include one or more NICs that you had previously excluded. You can include the NICs that you want to use for Veritas Access.

To include a NIC

1 If you have excluded some NICs and not saved your configuration, it is possible to include a NIC again. When the installer asks you to select the NIC option that you want to configure in the cluster, enter **2** if you want to include a NIC.

```
Veritas Access 7.4.2.400 Configure Program
10.200.114.45 10.200.114.46

NIC bonding/NIC exclusion configuration

NIC bonding supports only public NICs. Make sure the NICs you choose
are connected to public network.

NIC  PCI ID       bond status    If excluded
=============================================
eth2 0000:02:03.0 (physical NIC)  Y
eth3 0000:02:04.0 (physical NIC)  Y
eth4 0000:02:05.0 (physical NIC)  N
eth5 0000:02:06.0 (physical NIC)  N
eth6 0000:02:07.0 (physical NIC)  N
eth7 0000:02:08.0 (physical NIC)  N

     1)  Exclude NICs
     2)  Include NICs
     3)  Create a new bond
     4)  Add NICs to a bond
     5)  Remove bonds
     6)  Remove NICs from the bond list
     7)  Create VLAN device
     8)  Delete VLAN device
     9)  Save and continue

Select the NIC option to be configured in this cluster: [1-9,q] 2
```

2 The installer prompts you to select the NIC that you want to include. Enter your choice.

```
Choose NICs for inclusion

     1)  eth2 0000:02:03.0 (excluded NIC)
     2)  eth3 0000:02:04.0 (excluded NIC)
     3)  Include NICs per node
     b)  Back to previous menu

Choose NICs: [1-6,b,q] 1
```

3 The installer goes back to the previous menu. You can choose another NIC for inclusion. Enter **2** to include another NIC. Or you can save your configurations and continue with the installation of Veritas Access.

If you want to save your configurations, enter **9**.

```
Veritas Access 7.4.2.400 Configure Program
10.200.114.45 10.200.114.46

NIC bonding/NIC exclusion configuration

NIC bonding supports only public NICs. Make sure the NICs you choose
are connected to public network.

NIC  PCI ID       bond status    If excluded
=============================================
eth2 0000:02:03.0 (physical NIC) N
eth3 0000:02:04.0 (physical NIC) Y
eth4 0000:02:05.0 (physical NIC) N
eth5 0000:02:06.0 (physical NIC) N
eth6 0000:02:07.0 (physical NIC) N
eth7 0000:02:08.0 (physical NIC) N

     1)  Exclude NICs
     2)  Include NICs
     3)  Create a new bond
     4)  Add NICs to a bond
     5)  Remove bonds
     6)  Remove NICs from the bond list
     7)  Create VLAN device
     8)  Delete VLAN device
     9)  Save and continue

Select the NIC option to be configured in this cluster: [1-9,q]
```

4 If you want to include NICs per node, in Step 2 enter **3**.

Note: NIC exclusion function is supported on InfiniBand NICs, but all the NICs with the same PCI ID are excluded during the exclusion operation. See "About using LLT over the RDMA network for Veritas Access" on page 28.

Creating a NIC bond

An administrator can create a bond NIC interface from a given list of public NIC interfaces during Veritas Access installation. This feature allows an administrator to save a number of physical IP addresses that are used for installation and post-installation bond creation.

- You cannot bond InfiniBand NICs because the PCI IDs are identical. See “About using LLT over the RDMA network for Veritas Access” on page 28.

If you do not want to create a bond interface, continue with the installation.

See “About obtaining IP addresses” on page 34.

See “About calculating IP address requirements” on page 35.

To create a bond

1 After you choose manual selection mode, the installer prompts you to enter your selection. Enter **5** to create a new bond.

```
NIC selection/NIC bonding/NIC VLAN configuration

Common NICs on all systems:

NIC        Type       Properties       Public
================================================
ens161     Physical   -                N
ens192     Physical   -                Y
ens193     Physical   -                Y
ens224     Physical   -                Y
ens225     Physical   -                Y
ens256     Physical   -                N
ens257     Physical   -                Y

     1)  Select public NICs
     2)  Select private NICs
     3)  Exclude NICs
     4)  Include NICs
     5)  Create a new bond
     6)  Add NICs to a bond
     7)  Remove bonds
     8)  Remove NICs from the bond list
     9)  Create VLAN device
    10)  Delete VLAN device
    11)  Save and continue

Select the NIC option to be configured in this cluster: [1-11,q] 5
```

2 Select a bond mode for the new bond.

```
Configure the mode of the NIC bonding:

     1)  balance-rr
     2)  active-backup
     3)  balance-xor
     4)  broadcast
     5)  802.3ad
     6)  balance-tlb
     7)  balance-alb
     b)  Back to previous menu

Select the bonding mode: [1-7,b,q] 1

bond0 is created. Please add NICs to bond0 to enable it.

Press [Enter] to continue:
```

If you choose **3** or **5**, you need to choose the bond option for the bond mode:

```
1) layer2
2) layer3+4
3) default

Select the bonding option: [1-3,b,q] 1
```

The installer prompts you to select the NIC option that you want to configure for the cluster.

3 Enter **6** to add NICs to bond.

Note: You need to have NIC in bond.

```
NIC selection/NIC bonding/NIC VLAN configuration

Common NICs on all systems:

NIC        Type       Properties       Public
=================================================
ens161     Physical   -                N
ens192     Physical   -                Y
ens193     Physical   -                Y
ens224     Physical   -                Y
ens225     Physical   -                Y
ens256     Physical   -                N
ens257     Physical   -                Y
bond0      Virtual    Bond balance-rr  -

     1)  Select public NICs
     2)  Select private NICs
     3)  Exclude NICs
     4)  Include NICs
     5)  Create a new bond
     6)  Add NICs to a bond
     7)  Remove bonds
     8)  Remove NICs from the bond list
     9)  Create VLAN device
    10)  Delete VLAN device
    11)  Save and continue

Select the NIC option to be configured in this cluster: [1-11,q] 6
```

4 Enter **6** to select a NIC that you want to add in a bond.

```
Choose NICs for bonding

     1)  ens161
     2)  ens192
     3)  ens193
     4)  ens224
     5)  ens225
     6)  ens256
     7)  ens257
     b)  Back to previous menu

Choose NICs, separated by spaces: [1-7,b,q,?] 4 5
```

5 Select a bond name for which you want to add the NIC

```
Choose a bond name to add NICs

     1)  bond0
     b)  Back to previous menu

Choose bonds, separated by spaces: [1-1,b,q] 1

Adding ens224 ens225 to bond0 was successful

Press [Enter] to continue:

NIC selection/NIC bonding/NIC VLAN configuration

Common NICs on all systems:

NIC        Type       Properties       Public
=====================================================
ens161     Physical   -                N
ens192     Physical   -                Y
ens193     Physical   -                Y
ens224     Physical   Slave of bond0
ens225     Physical   Slave of bond0
ens256     Physical   -                N
ens257     Physical   -                Y
bond0      Virtual    Bond balance-rr  Y

     1)  Select public NICs
     2)  Select private NICs
     3)  Exclude NICs
     4)  Include NICs
     5)  Create a new bond
     6)  Add NICs to a bond
     7)  Remove bonds
     8)  Remove NICs from the bond list
     9)  Create VLAN device
    10)  Delete VLAN device
    11)  Save and continue

Select the NIC option to be configured in this cluster: [1-11,q]
```

Removing a NIC bond

An administrator can remove a bond.

To remove a NIC bond

1 Enter **7** to remove an existing bond.

```
Common NICs on all systems:

NIC        Type       Properties         Public
=========================================================
ens161     Physical   -                  N
ens192     Physical   -                  Y
ens193     Physical   Slave of bond1
ens224     Physical   Slave of bond0
ens225     Physical   Slave of bond0
ens256     Physical   -                  N
ens257     Physical   -                  Y
bond0      Virtual    Bond balance-rr    Y
bond1      Virtual    Bond active-backup Y

     1)  Select public NICs
     2)  Select private NICs
     3)  Exclude NICs
     4)  Include NICs
     5)  Create a new bond
     6)  Add NICs to a bond
     7)  Remove bonds
     8)  Remove NICs from the bond list
     9)  Create VLAN device
    10)  Delete VLAN device
    11)  Save and continue

Select the NIC option to be configured in this cluster: [1-11,q] 7
```

2 Select the bond that you want to remove.

```
Choose bonds to be removed

     1)  bond0
     2)  bond1
     b)  Back to previous menu

Choose bonds, separated by spaces: [1-2,b,q] 2

Deleting NIC bonding bond1 succeeded

Press [Enter] to continue:
NIC selection/NIC bonding/NIC VLAN configuration

Common NICs on all systems:

NIC        Type       Properties       Public
=====================================================
ens161     Physical   -                N
ens192     Physical   -                Y
ens193     Physical   -                Y
ens224     Physical   Slave of bond0
ens225     Physical   Slave of bond0
ens256     Physical   -                N
ens257     Physical   -                Y
bond0      Virtual    Bond balance-rr  Y

     1)  Select public NICs
     2)  Select private NICs
     3)  Exclude NICs
     4)  Include NICs
     5)  Create a new bond
     6)  Add NICs to a bond
     7)  Remove bonds
     8)  Remove NICs from the bond list
     9)  Create VLAN device
    10)  Delete VLAN device
    11)  Save and continue

Select the NIC option to be configured in this cluster: [1-11,q]
```

Removing a NIC from the bond list

During installation, an administrator can remove a NIC which is already a slave of a bond before the configuration is saved.

To remove a NIC from the bond list

1 During the Veritas Access installation, the installer prompts you to enter your selection. Enter **8** to remove a NIC from the bond list.

Note: The NIC bonding or NIC exclusion configuration options support both a single NIC or bond and multiple NICs or bonds.

```
NIC selection/NIC bonding/NIC VLAN configuration

Common NICs on all systems:

NIC        Type       Properties         Public
=======================================================
ens161     Physical   -                  N
ens192     Physical   Slave of bond1
ens193     Physical   Slave of bond1
ens224     Physical   Slave of bond0
ens225     Physical   Slave of bond0
ens256     Physical   -                  N
ens257     Physical   -                  Y
bond0      Virtual    Bond balance-rr    Y
bond1      Virtual    Bond active-backup Y

     1)  Select public NICs
     2)  Select private NICs
     3)  Exclude NICs
     4)  Include NICs
     5)  Create a new bond
     6)  Add NICs to a bond
     7)  Remove bonds
     8)  Remove NICs from the bond list
     9)  Create VLAN device
    10)  Delete VLAN device
    11)  Save and continue

Select the NIC option to be configured in this cluster: [1-11,q] 8
```

2 Select a NIC that you want to remove from the NIC bonding.

```
Choose NICs to be deleted from the NIC bonding

     1)  ens192
     2)  ens193
     3)  ens224
     4)  ens225
     b)  Back to previous menu

Choose NICs, separated by spaces: [1-4,b,q,?] 1

Removing ens192 from bonding was successful

Press [Enter] to continue:
NIC selection/NIC bonding/NIC VLAN configuration

Common NICs on all systems:

NIC        Type       Properties          Public
=======================================================
ens161     Physical   -                   N
ens192     Physical   -                   Y
ens193     Physical   Slave of bond1
ens224     Physical   Slave of bond0
ens225     Physical   Slave of bond0
ens256     Physical   -                   N
ens257     Physical   -                   Y
bond0      Virtual    Bond balance-rr     Y
bond1      Virtual    Bond active-backup  Y

     1)  Select public NICs
     2)  Select private NICs
     3)  Exclude NICs
     4)  Include NICs
     5)  Create a new bond
     6)  Add NICs to a bond
     7)  Remove bonds
     8)  Remove NICs from the bond list
     9)  Create VLAN device
    10)  Delete VLAN device
    11)  Save and continue
```

```
Select the NIC option to be configured in this cluster: [1-11,q]
```

About VLAN tagging

When VLANs (Virtual Local Area Network) span multiple switches, VLAN tagging is required. A VLAN is a way to create independent logical networks within a physical network. VLAN tagging is the practice of inserting a VLAN ID into a packet header to identify which VLAN the packet belongs to.

By using the VLAN tagging feature, you can:

- Create a VLAN device during installation
- Create a VLAN device on the specified bond interface.

Note: You need to create a bond interface first.

See "Creating a VLAN device " on page 87.
See "Removing a VLAN device " on page 90.

Creating a VLAN device

You can create a VLAN device for a public NIC interface or a public bond.

See "About VLAN tagging" on page 87.

Note: If you need to use VLAN interface as public NIC while configuring the Veritas Access network, it is mandatory to add the NIC on which VLAN is created as public NIC during the Veritas Access installation.

For example, if VLAN is `eth0.100`, you should select `eth0.100` and `eth0` as public NIC during the access network configuration when you install Veritas Access.

To create a VLAN device

1 In the manual selection mode, enter **9** to create a VLAN device.

```
NIC selection/NIC bonding/NIC VLAN configuration

Common NICs on all systems:

NIC        Type       Properties       Public
=================================================
ens161     Physical   -                N
ens192     Physical   -                Y
ens193     Physical   -                Y
ens224     Physical   Slave of bond0
ens225     Physical   Slave of bond0
ens256     Physical   -                N
ens257     Physical   -                Y
bond0      Virtual    Bond balance-rr  Y

     1)  Select public NICs
     2)  Select private NICs
     3)  Exclude NICs
     4)  Include NICs
     5)  Create a new bond
     6)  Add NICs to a bond
     7)  Remove bonds
     8)  Remove NICs from the bond list
     9)  Create VLAN device
    10)  Delete VLAN device
    11)  Save and continue

Select the NIC option to be configured in this cluster: [1-11,q] 9
```

2 Select the NICs on which you want to create VLAN devices.

3 Enter the VLAN ID for the device.

```
Choose NICs to create VLAN device on:
     1)  ens161
     2)  ens192
     3)  ens193
     4)  ens256
     5)  ens257
     6)  bond0
     b)  Back to previous menu

Choose VLAN devices, separated by spaces: [1-6,b,q] 2

Enter the VLAN ID for the device (1-4094): [b,q,?] 100


NIC selection/NIC bonding/NIC VLAN configuration

Common NICs on all systems:

NIC         Type       Properties       Public
=====================================================
ens161      Physical   -                N
ens192      Physical   -                Y
ens193      Physical   -                Y
ens224      Physical   Slave of bond0
ens225      Physical   Slave of bond0
ens256      Physical   -                N
ens257      Physical   -                Y
bond0       Virtual    Bond balance-rr  Y
ens192.100 Virtual    VLAN 100         -

     1)  Select public NICs
     2)  Select private NICs
     3)  Exclude NICs
     4)  Include NICs
     5)  Create a new bond
     6)  Add NICs to a bond
     7)  Remove bonds
     8)  Remove NICs from the bond list
     9)  Create VLAN device
    10)  Delete VLAN device
    11)  Save and continue
```

```
Select the NIC option to be configured in this cluster: [1-11,q]
```

Removing a VLAN device

You can remove a VLAN device for a public NIC interface or a public bond.

See “About VLAN tagging” on page 87.

To remove a VLAN device

1 In the manual selection mode, enter **10** to remove a VLAN device.

```
Common NICs on all systems:

NIC         Type       Properties       Public
=====================================================
ens161      Physical   -                N
ens192      Physical   -                Y
ens193      Physical   -                Y
ens224      Physical   Slave of bond0
ens225      Physical   Slave of bond0
ens256      Physical   -                N
ens257      Physical   -                Y
bond0       Virtual    Bond balance-rr  Y
ens192.100 Virtual    VLAN 100         -

     1)  Select public NICs
     2)  Select private NICs
     3)  Exclude NICs
     4)  Include NICs
     5)  Create a new bond
     6)  Add NICs to a bond
     7)  Remove bonds
     8)  Remove NICs from the bond list
     9)  Create VLAN device
    10)  Delete VLAN device
    11)  Save and continue

Select the NIC option to be configured in this cluster: [1-11,q] 10
```

2 Select the VLAN NICs that you want remove.

```
Choose VLAN NICs to be deleted

     1)  ens192.100
     b)  Back to previous menu

Choose VLAN devices, separated by spaces: [1-1,b,q] 1

NIC selection/NIC bonding/NIC VLAN configuration

Common NICs on all systems:

NIC        Type       Properties       Public
=================================================
ens161     Physical   -                N
ens192     Physical   -                Y
ens193     Physical   -                Y
ens224     Physical   Slave of bond0
ens225     Physical   Slave of bond0
ens256     Physical   -                N
ens257     Physical   -                Y
bond0      Virtual    Bond balance-rr  Y

     1)  Select public NICs
     2)  Select private NICs
     3)  Exclude NICs
     4)  Include NICs
     5)  Create a new bond
     6)  Add NICs to a bond
     7)  Remove bonds
     8)  Remove NICs from the bond list
     9)  Create VLAN device
    10)  Delete VLAN device
    11)  Save and continue

Select the NIC option to be configured in this cluster: [1-11,q]
```

Limitations of VLAN Tagging

Note the following limitations for using VLAN Tagging:

- Support only for a fresh installation.

- Support only for creating a VLAN device on a bonded NIC.
- Support only for creating one VLAN device at installation time.

Replacing an Ethernet interface card

In some cases, you may need to replace an Ethernet interface card on a node. This section describes the steps to replace the card.

Note: This procedure works for replacing an existing Ethernet interface card. It does not work for adding an Ethernet interface card to the cluster. If the Ethernet interface card you add needs a new device driver, install the new device driver before installing the Ethernet interface card on the node.

To replace an Ethernet interface card

1. Use the `Cluster> shutdown` command to shut down the node.

   For example:

   ```
   Cluster> shutdown access-03
   Stopping Cluster processes on access-03.......done
   Sent shutdown command to access-03
   ```

2. Use the `Cluster> del` command to delete the node from the cluster.

   For example:

   ```
   Cluster> del access-03
   ```

3. Install the replacement Ethernet interface card on the node.
4. Turn on the node.
5. Make sure that the Ethernet interface card is active and online.
6. Use the `Cluster> add` command to add the node back into the cluster.

   For example:

   ```
   Cluster> add 172.16.113.118
   ```

For details on the `Cluster> add` and `Upgrade>`commands that are described in this section, see the relevant man pages.

Configuring I/O fencing

Veritas Access supports two fencing modes:

- Disk-based fencing
- Majority-based fencing

If the cluster includes shared storage, during installation you are prompted to configure either disk-based or majority-based I/O fencing. You must choose a minimum of three available VxVM disks or initialize three disks as VxVM disks to form the fencing disk group. Ensure that you choose an odd number of disks for configuring fencing.

If the cluster does not have shared storage, majority-based fencing is configured automatically during the installation and you are not prompted to specify any option.

About configuring Veritas NetBackup

If you use Veritas NetBackup, to comply with the NetBackup End-User License Agreement (EULA), you have to purchase and enter valid license keys on the external NetBackup master server before you configure NetBackup to work with Veritas Access. For more information on entering the NetBackup license keys on the NetBackup master server, see the *Veritas NetBackup Installation Guide*.

If you use NetBackup, configure the virtual IP address using the `Backup> virtual-ip` command so that it is different from all of the virtual IP addresses, including the console server IP address and the physical IP addresses that are used to install the Veritas Access software.

About enabling kdump during an Veritas Access configuration

During the Veritas Access configuration, the Veritas Access installer tries to enable kdump on your cluster node. To meet the Veritas Access software requirements, the installer modifies the `/etc/kdump.conf` and `/boot/grub/grub.conf` files by using the following options:

- ```
  /boot/grub/grub.conf
  crashkernel = 512M-2G:64M, 2G-:256M
  ```

- ```
  /etc/kdump.conf
  path /opt/VRTSsnas/core/kernel/
  core_collector makedumpfile -c --message-level 1 -d 31
  ```

Configuring a KMS server on the Veritas Access cluster

You can configure a Key Management Service (KMS) server on the Veritas Access cluster.

To configure a KMS server on the Veritas Access cluster

1. Obtain the KMS server's SSL public key (in base64 format) and its port number. This key is used for communication between the Veritas Access cluster and the KMS server.
2. Generate a self-signed SSL key-pair on the Veritas Access cluster:

   ```
   System> kms certificate generate
   ```

3. Import the KMS server's public key.

   ```
   System> kms certificate import_server_cert
   ```

4. Configure the KMS server. Provide the SSL public key that was obtained in step 1 as input here.

   ```
   System> kms config server <server_ip> <server_port>
   ```

   Where*server_ip* is the KMS server IP

   server_port is the KMS server port number.

5. KMS admin now sets up a trust certificate using its admin GUI to allow communication between the KMS server and Veritas Access cluster.

For more information, see the `system_kms` man page.

Chapter 6

Automating Veritas Access installation and configuration using response files

This chapter includes the following topics:

About response files

The installer script generates a response file during any installation, configuration, upgrade, or uninstall procedure. The response file contains the configuration information that you entered during the procedure. When the procedure completes, the installation script displays the location of the response files.

You can use the response file for future installation procedures by invoking an installation script with the `-responsefile` option. The response file passes arguments to the script to automate an installation or uninstallation.

See “Installation script options” on page 160.

Performing a silent Veritas Access installation

A silent installation and configuration is based on a response file that you prepare so that the Veritas Access software can be installed without prompts. This feature is useful if you want to install the Veritas Access software on a large number of nodes.

Before performing a silent Veritas Access installation and configuration, you have to manually configure a secure shell (ssh) communication between the nodes.

See "Manually configuring passwordless secure shell (ssh)" on page 162.

You can get the Veritas Access example response file from the root directory of the ISO image. Ensure that you specify the absolute path of the response file.

To use the Veritas Access silent installation feature

- Enter the following command:

```
# ./installaccess -responsefile /root/access.responsefile
```

To generate the access.response example file

1. Install and configure the Veritas Access software without any errors.
2. Get the `access.response` example file from the log directory.

To use the access.response example file

1. Rename the Veritas Access example response file to `access.responsefile`.
2. Modify the file by changing the cluster name, IP address ranges, and other parameters, as necessary for your configuration.

Installation times may vary depending on your configuration.

See "Installing and configuring the Veritas Access software on the cluster" on page 53.

Response file variables to install and configure Veritas Access

Table 6-1 lists the response file variables that you can define to install and configure Veritas Access.

Table 6-1 Response file variables for installing Veritas Access

Variable	Description
CFG{bondmode}{bond<n>}	Defines the bond modes for BOND. List or scalar: list Optional or required: optional
CFG{bondname}	List of bond names for BOND. List or scalar: list Optional or required: optional
CFG{bondpool}{bond<n>}	List of the PCI IDs of the slave NICs. List or scalar: list Optional or required: optional
CFG{config_majority_based_fencing}	Enables majority fencing. The value is `1`. It cannot be used with I/O fencing variables 'fencing_scsi3_disk_policy', 'fencing_newdg_disks', and 'fencing_dgname'. List or scalar: scalar Optional or required: required for majority-based fencing
CFG{exclusion}	List of PCI IDs of exlcluded NICs. List or scalar: list Optional or required: optional
CFG{fencing_dgname}	Specifies the disk group for I/O fencing. The value is `sfscoorddg`. List or scalar: scalar Optional or required: required for I/O fencing
CFG{fencing_newdg_disks}	Defines the fencing disks. List or scalar: list Optional or required: required for I/O fencing

Table 6-1 Response file variables for installing Veritas Access *(continued)*

Variable	Description
CFG{fencing_option}	Specifies the I/O fencing configuration mode. The value is `2` for disk-based I/O fencing. List or scalar: scalar Optional or required: required for I/O fencing
CFG{fencing_scsi3_disk_policy}	Specifies the SCSI-3 disk policy to use I/O fencing. The value is `dmp`. List or scalar: scalar Optional or required: required for I/O fencing
CFG{fencingenabled}	Defines whether fencing is enabled. The value is `1` if enabled. List or scalar: scalar Optional or required: required for I/O fencing
CFG{opt}{licensefile}	Specifies the location of the Veritas perpetual or subscription license key file. List or scalar: scalar Optional or required: required
CFG{keys}{"node_ip"}	Specifies the Veritas Access license for each node. List or scalar: scalar Optional or required: required
CFG{newnodes}	Specifies the new access IP for the cluster nodes. The value should be the first public IP address for each node. List or scalar: list Optional or required: required
CFG{opt}{comcleanup}	Cleans up the ssh connection that is added by the installer after the configuration. The value is `1`. List or scalar: scalar Optional or required: required

Table 6-1 Response file variables for installing Veritas Access *(continued)*

Variable	Description
CFG{opt}{configNIC}	Performs the NIC configuration with all the network variable values. The value is `1`. List or scalar: scalar Optional or required: required
CFG{opt}{configure}	Performs the configuration if the packages are already installed. List or scalar: scalar Optional or required: required
CFG{opt}{install}	Installs Veritas Access RPMs. Configuration can be performed at a later time using the `-configure` option. List or scalar: scalar Optional or required: optional
CFG{opt}{installallpkgs}	Instructs the installer to install all the Veritas Access RPMs based on the variable that has the value set to 1. List or scalar: scalar Optional or required: required
CFG{opt}{noipc}	Disables the connection to SORT for updates check. The value is `0`. List or scalar: scalar Optional or required: required
CFG{opt}{ssh}	Determines whether to use ssh for communication between systems. The value is `1` if enabled. List or scalar: scalar Optional or required: required
CFG{prod}	Defines the product to be installed or uninstalled. List or scalar: scalar Optional or required: required

Table 6-1 Response file variables for installing Veritas Access *(continued)*

Variable	Description
CFG{publicbond}	List of PCI IDs of the bonded NICs. List or scalar: list Optional or required: optional
CFG{publicnetmaskarr}	List of netmasks that are assigned to public NICs or bonds. List or scalar: list Optional or required: required
CFG{publicparr}	List of public IPs that are assigned to public NICs or bonds. List or scalar: list Optional or required: required
CFG{redhat_subscription_username}	Specifies the user name to register with Red Hat subscription management. List or scalar: scalar Optional or required: required if some required OS rpms are missing on the systems The user name should be enclosed in single quotes (for example : '1234@abc') if it contains any special character.
CFG{redhat_subscription_password}	Specifies the password to register with Red Hat subscription management. List or scalar: scalar Optional or required: required if some required OS rpms are missing on the systems The password should be enclosed in single quotes (for example : '1234@abc') if it contains any special character.
CFG{snas_clustername}	Defines the cluster name of the product. List or scalar: scalar Optional or required: required

Table 6-1 Response file variables for installing Veritas Access *(continued)*

Variable	Description
CFG{snas_consoleip}	Defines the console IP of the product. List or scalar: scalar Optional or required: required
CFG{snas_defgateway}	Defines the gateway of the product. List or scalar: scalar Optional or required: required
CFG{snas_dnsdomainname}	Defines the DNS domain name of the product. List or scalar: scalar Optional or required: required
CFG{snas_dnsip}	Defines the DNS IP of the product. List or scalar: scalar Optional or required: required
CFG{snas_ntpserver}	Defines the NTP server name of the product. List or scalar: scalar Optional or required: required
CFG{snas_nvip}	Defines the number of VIPs on each NIC. List or scalar: scalar Optional or required: required
CFG{snas_pipprefix}	Defines the prefix of public IPs (only in IPV6 environments). List or scalar: scalar Optional or required: required
CFG{snas_pipstart}	Defines the the initial IP of the public IPs. List or scalar: scalar Optional or required: required

Table 6-1 Response file variables for installing Veritas Access *(continued)*

Variable	Description
CFG{snas_pnmaskstart}	Defines the netmask of public IPs (only in IPV4 environments). List or scalar: scalar Optional or required: required
CFG{snas_sepconsoleport}	Defines if use of separate console port. 1 for yes, 0 for no. List or scalar: scalar Optional or required: required
CFG{snas_vipprefix}	Defines the prefix of virtual IPs (only in IPV6 environments). List or scalar: scalar Optional or required: required
CFG{snas_vipstart}	Defines the the initial IP of the virtual IPs. List or scalar: scalar Optional or required: required
CFG{snas_vnmaskstart}	Defines the netmask of virtual IPs (only in IPV4 environments). List or scalar: scalar Optional or required: required
CFG{systems}	List of systems on which the product is to be installed or uninstalled. List or scalar: list Optional or required: required
CFG{vcs_allowcomms}	Indicates whether to start LLT or GAB when the user wants to set up a single node cluster. List or scalar: scalar Optional or required: required

Table 6-1 Response file variables for installing Veritas Access *(continued)*

Variable	Description
CFG{vcs_clusterid}	Defines the unique cluser ID with a string number. List or scalar: scalar Optional or required: required
CFG{vcs_lltlink<n>}{"new_node_ip"}	Defines the NIC name for the first heartbeat link. List or scalar: scalar Optional or required: required
CFG{vcs_rdmalink1_address}{"new_node_ip"}	Specifies the RDMA private link1 IP address, the value follows that node-01 is `172.16.0.3`, node-02 is `172.16.0.4`, and so on. List or scalar: scalar Optional or required: required for RDMA NICs as private NICs
CFG{vcs_rdmalink1_netmask}{"new_node_ip"}	Specifies the RDMA private link1 IP netmask, the value is `255.255.255.0`. List or scalar: scalar Optional or required: required for RDMA NICs as private NICs
CFG{vcs_rdmalink1_port}{"new_node_ip"}	Specifies the port number for the RDMA private link1, the value is `51001`. List or scalar: scalar Optional or required: required for RDMA NICs as private NICs
CFG{vcs_rdmalink2_address}{"new_node_ip"}	Specifies the RDMA private link2 IP address, the value follows that node-01 is `172.16.1.3`, node-02 is `172.16.1.4`, and so on. List or scalar: scalar Optional or required: required for RDMA NICs as private NICs

Table 6-1 Response file variables for installing Veritas Access *(continued)*

Variable	Description
CFG{vcs_rdmalink2_netmask}{"new_node_ip"}	Specifies the RDMA private link2 IP netmask, the value is `255.255.255.0`. List or scalar: scalar Optional or required: required for RDMA NICs as private NICs
CFG{vcs_rdmalink2_port}{"new_node_ip"}	Specifies the port number for the RDMA private link2, the value is `51002`. List or scalar: scalar Optional or required: required for RDMA NICs as private NICs
CFG{vcs_userenpw}	Defines the encrypted user password. List or scalar: scalar Optional or required: required
CFG{vcs_username}	Defines the added username for VCS. List or scalar: scalar Optional or required: required
CFG{vcs_userpriv}	Defines the user privilege. List or scalar: scalar Optional or required: required
CFG{virtualiparr}	List of virtual IPs that will be assigned to public NICs or bonds. List or scalar: list Optional or required: required
CFG{virtualnetmaskarr}	List of netmasks that will be assigned to public NICs or bonds. List or scalar: list Optional or required: required

Sample response file for Veritas Access installation and configuration

The following example shows a response file for installing and configuring Veritas Access.

```
##################################################
our %CFG;
#Installs Product packages.
$CFG{opt}{install}=1;
$CFG{opt}{installallpkgs}=1;
$CFG{opt}{comsetup}=1;
$CFG{opt}{noipc}=1;
$CFG{opt}{ssh}=1;
$CFG{prod}="SNAS73";
$CFG{opt}{licensefile}="<absolute_path_of_licfile>";

#Performs the configuration if the packages are already installed
$CFG{opt}{configure}=1;

#the PCI IDs of slave NICs
$CFG{bondpool}{bond0}=[ qw(0000:02:09.0 0000:02:07.0) ];
$CFG{bondpool}{bond1}=[ qw(0000:02:04.0 0000:02:08.0) ];

#mode of each bond
$CFG{bondmode}{bond0}=5;
$CFG{bondmode}{bond1}=6;

#names of bond
$CFG{bondname}=[ qw(bond0 bond1) ];

#the PCI IDs of excluded NICs
$CFG{exclusion}=[ qw(0000:02:03.0 0000:02:0a.0) ];

#the PCI IDs of all the bonded NICs
$CFG{publicbond}=[ qw(0000:02:03.0 0000:02:04.0 0000:02:07.0
0000:02:08.0) ];

#public IPs
$CFG{publiciparr}=[ qw(10.200.58.100 10.200.58.101 10.200.58.102
10.200.58.103 10.200.58.104 10.200.58.105 10.200.58.106 10.200.58.107) ];

#netmask for public IPs
```

```
$CFG{publicnetmaskarr}=[ qw(192.168.30.10 192.168.30.11 192.168.30.12
192.168.30.13 192.168.30.14 192.168.30.15 192.168.30.16 192.168.30.17) ];

#the user name to register with Red Hat subscription management
$CFG{redhat_subscription_username}="rhel_user";

#the password to register with Red Hat subscription management
$CFG{redhat_subscription_password}="rhel_password";

#clustername of SNAS
$CFG{snas_clustername}="testsnas";

#console IP of SNAS
$CFG{snas_consoleip}="192.168.30.40";

#default gateway of SNAS
$CFG{snas_defgateway}="192.168.30.1";

#domain name of DNS
$CFG{snas_dnsdomainname}="cdc.veritas.com";

#IP of DNS
$CFG{snas_dnsip}="192.168.30.2";

#NTP server name
$CFG{snas_ntpserver}="ntp.veritas.com";

#number of VIPs on each NIC
$CFG{snas_nvip}=1;

#netmask of public IPs(only ipv4 environment)
$CFG{snas_pnmaskstart}=255.255.255.0;

#the initial IP of public IPs
$CFG{snas_pipstart}="192.168.30.10";

#if use separate console port, 1 for yes, 0 for no
$CFG{snas_sepconsoleport}="0";

#netmask of virutal IPs(only ipv4 environment)
$CFG{snas_vnmaskstart}=255.255.255.0;

#the initial IP of virtual IPs
```

```
$CFG{snas_vipstart}="192.168.30.18";

#virtual IPs
$CFG{virtualiparr}=[ qw(192.168.30.18 192.168.30.19 192.168.30.20
 192.168.30.21 192.168.30.22 192.168.30.23 192.168.30.24 192.168.30.25) ];

#netmask for virual IPs
$CFG{virtualnetmaskarr}=[ qw(255.255.255.0 255.255.255.0 255.255.255.0
255.255.255.0 255.255.255.0 255.255.255.0 255.255.255.0 255.255.255.0) ];

#target systems
$CFG{systems}=[ qw(192.168.30.80 192.168.30.81) ];

#indicates whether to start llt/gab when user wants to setup a single
node cluster
$CFG{vcs_allowcomms}=1;

#define the unique cluser id with a string number
$CFG{vcs_clusterid}=325;

#define the cluster name with a string
$CFG{vcs_clustername}="testsnas";

#define the nic name for the first heartbeat link.
$CFG{vcs_lltlink1}{"192.168.30.10"}="priveth0";
$CFG{vcs_lltlink1}{"192.168.30.13"}="priveth0";
$CFG{vcs_lltlink2}{"192.168.30.10"}="priveth1";
$CFG{vcs_lltlink2}{"192.168.30.13"}="priveth1";

#define the encrypted user password
$CFG{vcs_userenpw}=[ qw(GPQiPKpMQlQQoYQkPN) ];

#define the added username for VCS
$CFG{vcs_username}=[ qw(admin) ];

#define the user privilege
$CFG{vcs_userpriv}=[ qw(Administrators) ];

1;
##################################################
```

Chapter 7

Displaying and adding nodes to a cluster

This chapter includes the following topics:

About the Veritas Access installation states and conditions

Table 7-1 describes the Veritas Access installation states.

Table 7-1 Veritas Access installation states

Installation state	Description
RUNNING	Node is part of the cluster and the Veritas Access processes are running on it.
FAULTED	Node is down and/or the Veritas Access processes are not running on it.

Table 7-1 Veritas Access installation states *(continued)*

Installation state	Description
LEAVING	Node is leaving the cluster gracefully
EXITED	Node has exited the cluster gracefully
UNKNOWN	Exact state of the node cannot be determined

Depending on the cluster condition as described in Table 7-2, output for the `Cluster> show` command changes.

Table 7-2 Cluster conditions and states

Condition	Description
If the node is configured and part of the cluster, but the node is powered off.	State displays as FAULTED, and there is no installation state or network statistics.
If the node is configured and part of the cluster, but the node is physically removed from the cluster.	State displays as FAULTED, and there is no installation state or network statistics.
If the node is configured and part of the cluster, but the node is shutdown using the `Cluster> shutdown` command.	State changes from LEAVING to EXITED.
If the node is configured and part of the cluster, and you use the `Cluster> del` command.	Node is deleted from the cluster, and information about the deleted node is no longer available.

Displaying the nodes in the cluster

You can display all the nodes in the cluster, their states, CPU load, and network load during the past 15 minutes.

If you use the `Cluster> show currentload` option, you can display the CPU and network loads collected from now to the next five seconds.

To display a list of nodes in the cluster

1 To display a list of nodes that are part of a cluster, and the systems that are available to add to the cluster, enter the following:

```
Cluster> show
```

Command output includes the following information. See examples below.

Node	Displays the node name if the node has already been added to the cluster. Displays the IP address of the node if it is still in the process of being added to the cluster.
	Example:
	`node-01`
	or
	`192.168.30.10`

State	Displays the state of the node or the installation state of the system along with an IP address of the system if it is installed. See "About the Veritas Access installation states and conditions" on page 109.
CPU	Indicates the CPU load.
pubethX	Indicates the network load for the Public Interface X.
bondX	Indicates the network load for bond NIC X.

2 For nodes already in the cluster, the following is displayed:

```
Node      State     CPU(15 min) pubeth0(15 min)     pubeth1(15 min)
                        %       rx(MB/s) tx(MB/s)   rx(MB/s) tx(MB/s)
------    -------   ----------- -------- --------   -------- --------
snas-01   RUNNING     1.35        0.00     0.00       0.00     0.00
snas-02   RUNNING     1.96        0.00     0.00       0.00     0.00
```

3 For the nodes that are being added to the cluster, for the nodes that are being deleted from the cluster, and for the nodes that is getting upgraded, the following progress is displayed:

```
Nodes in Transition

Node/IP          Operation       State      Description
------           ---------       -----      -----------
192.168.30.11    Add node        FAILED     Installing packages
snas-03          Delete node     ONGOING    Removing node
snas-01,snas-02  Rolling upgrade ONGOING    Rolling upgrade phase 2
```

Note: The `add node` and `delete node` operations cannot be performed at the same time.

4 To display the CPU and network loads collected from now to the next five seconds, enter the following:

```
Cluster> show currentload
```

Example output:

```
Node       State    CPU(5 sec)   pubeth0(5 sec)    pubeth1(5 sec)
                        %       rx(MB/s) tx(MB/s) rx(MB/s) tx(MB/s)
----       -----    ----------  -------- -------- -------- --------
snas-01    RUNNING    0.26       0.01     0.00      0.01     0.00
snas-02    RUNNING    0.87       0.01     0.00      0.01     0.00
snas-03    RUNNING    10.78      27.83    12.54     0.01     0.00
```

Statistics for network interfaces are shown for each public interface available on the cluster nodes.

Before adding new nodes in the cluster

After you have installed the operating system, you can install and configure a multiple node Veritas Access cluster at one time. If you want to add additional nodes to the cluster after that, you need to complete the following procedures:

- Install the appropriate operating system software on the additional nodes.
- You do not need to install the Veritas Access software on the additional node before you add the node. The Veritas Access software is installed when you

add the nodes. If the Veritas Access software is already installed, it is uninstalled and the product (same version as the cluster) is installed after that. The reason to uninstall and then install the product is to make sure that the new node is installed with exactly the same version, and patch level (if any) as the other cluster nodes. The packages are stored in the cluster nodes so the product image is not needed during the addition of the new node.

- Verify that the existing cluster has sufficient physical IP addresses for the new nodes. You can add additional IP addresses using the following command: .

```
Network> ip addr add command
```

For example:

```
Network> ip addr add 192.168.30.107 255.255.252.0 physical
ACCESS ip addr SUCCESS V-288-1031 ip addr add successful.

Network> ip addr show
IP             Netmask/Prefix  Device  Node       Type     Status
--             --------------  ------  ----       ----     ------
192.168.30.10 255.255.252.0   pubeth0 snas-01    Physical
192.168.30.11 255.255.252.0   pubeth1 snas-01    Physical
192.168.30.12 255.255.252.0   pubeth0 snas-02    Physical
192.168.30.13 255.255.252.0   pubeth1 snas-02    Physical
192.168.30.14 255.255.252.0           ( unused ) Physical
192.168.30.15 255.255.252.0           ( unused ) Physical
192.168.30.16 255.255.252.0   pubeth0 snas-01    Virtual  ONLINE (Con IP)
192.168.30.17 255.255.252.0   pubeth1 snas-01    Virtual  ONLINE
192.168.30.18 255.255.252.0   pubeth1 snas-01    Virtual  ONLINE
192.168.30.19 255.255.252.0   pubeth1 snas-01    Virtual
```

In the example, the unused IP addresses 192.168.30.14 and 192.168.30.15 can be used by the new node as physical IP addresses.

Note: The network configuration on the new nodes should be the same as that of the cluster nodes, that is, NICs should have same names and connectivity.

Bonds and vLANs are created automatically to match the cluster configuration if they do not exist already.

- Add the node to your existing cluster.

Adding a node to the cluster

You must install the operating system on the nodes before you add nodes to a cluster.

If you use the disk-based fencing, the coordinator disks must be visible on the newly added node as a prerequisite for I/O fencing to be configured successfully. Without the coordinator disks, I/O fencing does not load properly and the node cannot obtain the cluster membership.

If you use majority-based fencing, the newly added node does not have to have shared disks.

If you want to add a new node and want to exclude some unique PCI IDs, add the unique PCI IDs to the `/opt/VRTSsnas/conf/net_exclusion_dev.conf` file on each cluster node manually. For example:

```
[root@bob-01 ~]# cat /opt/VRTSsnas/conf/net_exclusion_dev.conf
0000:42:00.0 0000:42:00.1
```

Note: The writeback cache is supported for the two-node clusters only. Therefore, adding nodes to a two-node cluster, changes the caching to read-only.

Newly added nodes should have the same configuration of InfiniBand NICs.

If your cluster has configured the FSS pool and the pool's node group is missing a node, the newly added node is added into the FSS node group. The installer adds the new node's local data disks into the FSS pool.

To add the new node to the cluster

1. Sign in to Veritas Access using the `master` or the `system-admin` account.
2. In the Veritas Access command-line interface, enter the `Cluster` command to enter the `Cluster>` mode.
3. To add the new nodes to the cluster, enter the following:

   ```
   Cluster> add node1ip, node2ip.....
   ```

   where *node1ip,node2ip,....* are the IP address list of the additional nodes for the ssh connection.

   It is important to note that:

   - The node IPs are preserved and additional required are assigned from (unused) pool of physical IPs.

- The physical IPs of new nodes are usable IPs found from the configured public IP starting addresses.
- The virtual IPs are re-balanced to the new node but additional virtual IPs are not assigned.
  Go to step 3 to add new virtual IP addresses to the cluster after adding a node.
- The IPs that are accessible to the new nodes should be given.
- The accessible IPs of the new nodes should be in the public network, they should be able to ping the public network's gateway successfully.

For example:

```
Cluster> add 192.168.30.10
```

Note: You cannot add nodes to a two node cluster when the writeback caching is enabled. Before you add a node, change the cache mode to read and then try again.

---

4 If a cache exists on the original cluster, the installer prompts you to choose the SSD disks to create cache on the new node when CFS is mounted.

```
1) emc_clariion1_242
2) emc_clariion1_243
b) Back to previous menu
Choose disks separate by spaces to create cache on 192.168.30.11
[1-2,b,q] 1
Create cache on snas-02 .....................Done
```

5 If the cluster nodes have created FSS pool, and there are more than two local data disks on the new node, the installer asks you to select the disks to add into the FSS pool. Make sure that you select at least two disks for stripe volume layout. The total selected disk size should be no less than the FSS pool's capacity size.

```
Following storage pools need to add disk from the new node:
     1)  fsspool1
     2)  fsspool2
     3)  Skip this step

Choose a pool to add disks [1-3,q] 1
     1)  emc_clariion0_1570 (5.000 GB)
     2)  installres_03_sdc (5.000 GB)
     3)  installres_03_sde (5.000 GB)
     4)  sdd (5.000 GB)
     b)  Back to previous menu

Choose at least 2 local disks with minimum capacity of 10 GB [1-4,b,q] 2 4
Format disk installres_03_sdc,sdd ............................... Done

The disk name changed to installres_03_sdc,installres_03_sdd
    Add disk installres_03_sdc,installres_03_sdd to storage pool fsspool1  Done
```

6 If required, add the virtual IP addresses to the cluster. Adding the node does not add new virtual IP addresses or service groups to the cluster.

To add additional virtual IP addresses, use the following command in the Network mode:

```
Network> ip addr add ipaddr virtual
```

For example:

```
Network> ip addr add 192.168.30.14 255.255.252.0 virtual

ACCESS ip addr SUCCESS V-288-1031 ip addr add successful.
```

If a problem occurs while you are adding a node to a cluster (for example, if the node is temporarily disconnected from the network), do the following to fix the problem:

To recover the node:

- Power off the node.

- Use the `Cluster> del` *`nodename`* command to delete the node from the cluster.
- Power on the node.
- Use the `Cluster> add` *`nodeip`* command to add the node to the cluster.

Adding a node in mixed mode environment

To add a node in mixed mode

1. **Prerequisites:**

   The number of IPv4 and IPv6 IPs that are added should be equal to the number of public Interfaces.

   Use the same type of IP (that is, IPv4 or IPv6) that you have used at the time of the Veritas Access installation.

   Make sure that the IPv6 IP auto-assignment is disabled on the new node.

2. Do one of the following:

   - If you have used IPv4 address at the time of the Veritas Access installation, run the following command:

     ```
     cluster> add <IPv4 IP>
     ```

   - If you have used IPv6 address at the time of the Veritas Access installation, run the following command:

     ```
     cluster> add <IPv6 IP>
     ```

Deleting a node from the cluster

You can delete a node from the cluster. Use the node name that is displayed in the `Cluster> show` command.

If the deleted node was in the RUNNING state prior to deletion, after you reboot the node, that node is assigned to the original IP address that can be used to add the node back to the cluster. The original IP address of the node is the IP address that the node used before it was added into the cluster.

If your cluster has an FSS pool configured, the delete node operation may result in permanent loss of data for file systems that have simple of striped layouts for which there are no backup copies. For such file systems, it is required to back up or evacuate the data first before deleting the node.

If your cluster has configured an FSS pool, you cannot use the installer to delete nodes that would result in a single node in the node group of the FSS pool.

Deleting a node from a two-node cluster that has writeback caching enabled changes the caching to read-only. Writeback caching is only supported for two nodes.

The IP address that was used by the node before it was deleted from the cluster is still accessible until you perform a restart operation.

After the node is deleted from the cluster, when you perform a reboot operation, the old IP configuration is restored. Therefore, make sure to remove the used IPs from Veritas Access for the deleted node or vice versa.

To delete a node from the cluster

1. To show the current state of all nodes in the cluster, enter the following:

   ```
   Cluster> show
   ```

2. To delete a node from a cluster, enter the following:

   ```
   Cluster> del nodename
   ```

   where *nodename* is the node name that appeared in the listing from the `Cluster> show` command. You cannot specify a node by its IP address.

   Note: This command is not supported in a single-node cluster.

   For example:

   ```
   Cluster> del snas-01
   ```

3 After a node is deleted from the cluster, the physical IP addresses that it used are marked as unused physical IP addresses. The IP addresses are available for use if you add new nodes. The virtual IP addresses used by a node which has been deleted are not removed. Deleting a node moves the virtual IP addresses on the deleted node to the remaining nodes in the cluster.

For example:

```
Network> ip addr show
IP             Netmask/Prefix  Device     Node            Type     Status
--             --------------  ------     ----            ----     ------
192.168.30.10 255.255.252.0   pubeth0    source-30a-01   Physical
192.168.30.11 255.255.252.0   pubeth1    source-30a-01   Physical
192.168.30.12 255.255.252.0              ( unused )      Physical
192.168.30.13 255.255.252.0              ( unused )      Physical
192.168.30.14 255.255.252.0   pubeth0    source-30a-01   Virtual  ONLINE (Con IP)
192.168.30.15 255.255.252.0   pubeth0    source-30a-01   Virtual  ONLINE
192.168.30.16 255.255.252.0   pubeth0    source-30a-01   Virtual  ONLINE
192.168.30.17 255.255.252.0   pubeth1    source_30a-01   Virtual  ONLINE
192.168.30.18 255.255.252.0   pubeth1    source-30a-01   Virtual  ONLINE
```

If the physical or virtual IP addresses are not going to be used, they can be removed using the following command:

```
Network> ip addr del ipaddr
```

For example:

```
Network> ip addr del 192.168.30.18
ACCESS ip addr SUCCESS V-288-1031 ip addr del successful.
```

Note: If the cluster has configured NIC bonding, you also need to delete the configuration of the deleted node on the switch.

Shutting down the cluster nodes

You can shut down a single node or all of the nodes in the cluster. Use the node name that is displayed in the `Cluster> show` command.

To shut down a node or all the nodes in a cluster

1 To shut down a node, enter the following:

```
Cluster> shutdown nodename
```

nodename indicates the name of the node you want to shut down. You cannot specify a node by its IP address.

For example:

```
Cluster> shutdown snas-04
Stopping Cluster processes on snas-04
Sent shutdown command to snas-04. SSH sessions to
snas-04 may terminate.
```

2 To shut down all of the nodes in the cluster, enter the following:

```
Cluster> shutdown all
```

Use `all` as the *nodename* to shut down all of the nodes in the cluster.

For example:

```
Cluster> shutdown all
Stopping Cluster processes on all
SSH sessions to all nodes may terminate.
Sent shutdown command to snas-02
Sent shutdown command to snas-03
Sent shutdown command to snas-04
Sent shutdown command to snas-01
```

Chapter 8

Upgrading the operating system and Veritas Access

This chapter includes the following topics:

Supported upgrade paths for upgrades on RHEL

Table 8-1 provides details about the supported upgrade paths for upgrades on Red Hat Enterprise Linux (RHEL) from 7.4.2.300 or 7.4.2.301 release of Veritas Access to Veritas Access 7.4.2.400.

See “Upgrading the operating system and Veritas Access” on page 122.

Table 8-1 Supported upgrade paths for upgrades on RHEL

From product version	From operating	To operating system versions system versions
7.4.2.300	RHEL 7 Update 5	RHEL Update 7
7.4.2.301	RHEL 7 Update 5	RHEL Update 7

Upgrading the operating system and Veritas Access

This section describes the procedure to upgrade the operating system and Veritas Access.

This patch can be installed only on Veritas Access 7.4.2 release with Red Hat Enterprise Linux (RHEL) version 7.7 or 7.9. If you are on an earlier version of RHEL, you must upgrade to version 7.7 or 7.9 before installing the patch.

Upgrading the operating system and Veritas Access includes the following steps:

- To export the Veritas Access configurations
- To verify the Veritas Access configuration export
- To install the required version of RHEL
- To install Veritas Access 7.4.2.400
- To verify the Veritas Access installation
- To import the Veritas Access configuration

To export the Veritas Access configurations

1. You can export the Veritas Access configurations by using the script provided by Veritas Access.

   Prerequisites:

   - RHEL 7.7 or 7.9 must be installed.
   - Veritas Access version 7.4.2.300, 7.4.2.301, or later should be installed.
   - Make sure that you have stopped all I/Os and services related to Veritas Access such as CIFS, NFS, and FTP using the Veritas Access command-line interface.
   - Make sure that operations like create, destroy, add, and remove are not running since they may update the Veritas Access configuration.

2. From the ISO, copy the `upgrade_scripts/config_export` directory to the `root` directory of the cluster node on which the management console service group is online.

3. From the directory, run the following command on the shell (terminal) by using the `support` login to export the Veritas Access configurations:

   `/bin/bash -f export_lib.sh export local` *`filename`*

To verify the Veritas Access configuration export

- Run the following command using the Veritas Access command-line interface to see the list of available configurations:

  ```
  system config list
  ```

  you can find the configuration files in the `/opt/VRTSnas/conf/backup` location.

  Note: Store the configuration files on a node that is not part of the cluster nodes to avoid any damage to the configuration file.

To install the required version of RHEL

1. **Prerequisites:**

   - Make sure that you stop all the I/O processes and modules that are running on the Veritas Access command-line interface.
   - Run the `network> ip addr show` command and `cluster> show` command on the Veritas Access command-line interface before you install the RHEL operating system. Make a note of these IP addresses and cluster node names. Make sure to use the same IP addresses and cluster name when you install the Veritas Access cluster after the RHEL operating system is installed.

   Examples:

   ```
   upgrade> network ip addr show

   IP             Netmask/Prefix Device  Node        Type     Status
   --             -------------- ------  ----        ----     ------
   192.168.10.151 255.255.255.0  pubeth0 upgrade-01  Physical
   192.168.10.158 255.255.255.0  pubeth1 upgrade-01  Physical
   192.168.10.152 255.255.255.0  pubeth0 upgrade-02  Physical
   192.168.10.159 255.255.255.0  pubeth1 upgrade-02  Physical
   192.168.10.174 255.255.255.0  pubeth0 upgrade-01  Virtual  ONLINE(Con IP)
   192.168.10.160 255.255.255.0  pubeth0 upgrade-01  Virtual  ONLINE
   192.168.10.161 255.255.255.0  pubeth1 upgrade-01  Virtual  ONLINE

   upgrade> cluster show

   Node       State    CPU(15 min)   pubeth0(15 min)     pubeth1(15 min)
                           %        rx(MB/s) tx(MB/s)   rx(MB/s) tx(MB/s)
   ----       -----    -----------  -------- --------   -------- --------
   ```

```
upgrade-01 RUNNING  11.52          0.67     0.06       0.60     0.00
upgrade-02 RUNNING   4.19          0.61     0.05       0.60     0.00
```

Note: In this example, the cluster name is `upgrade` and the cluster node names are `upgrade-01` and `upgrade-02`.

2 Restart all the nodes of the cluster.

3 Install the supported RHEL operating system on the desired nodes.

See "Installing the operating system on each node of the cluster" on page 50.

Note: Veritas recommends to select the same disk or disks for the installation on which your previous version of the RHEL operating system was installed. Make sure that you do not select any other disk, because those disks may be part of a pool, and may result in data loss.

To install Veritas Access 7.4.2.400

- Restart the nodes and when they are up, start the Veritas Access 7.4.2.400 installation using the installer.

Note: Make sure that you use the same IP addresses and cluster name that were used for the previous Veritas Access installation.

See "Installing Veritas Access on the target cluster nodes" on page 52.

To verify the Veritas Access installation

1 By using the console IP, check whether the Veritas Access command-line interface is accessible.

2 Run the following command on the Veritas Access command-line interface to see whether the disks are accessible:

```
storage disk list
```

Note: If the disks are not visible in the command line output, run the `storage scanbus force` command on the Veritas Access command-line.

3 Run the following command to see whether the pools are accessible:

```
storage pool list
```

Note: If the pools are not visible in the command line output, run the `storage scanbus force` command on the Veritas Access command-line.

4 Run the following command to see whether the file systems are accessible:

```
storage fs list
```

Note: If the file systems are not visible in the command line output, run the `storage scanbus force` command on the Veritas Access command-line.

5 Make sure that the file systems are online. If the file systems are not online, you need to run the following command on the Veritas Access command-line to bring them online:

```
storage fs online fs name
```

To import the Veritas Access configuration

1 **Prerequisites:**

Make sure that the file systems are online. If the file systems are not online, you need to run the following command in the Veritas Access command-line interface to bring them online:

```
storage fs online <fs name>
```

Note: Make sure that the cluster uses the same IP addresses and cluster names that were used for the Veritas Access installation on the previous version of the operating system.

If the same virtual IP addresses which were used for Veritas Access on the previous RHEL version are not added during installation, add the virtual IPs from the Veritas Access command-line interface after Veritas Access is installed on the latest RHEL version, and then import the configuration.

2 Copy the exported configuration files to the cluster nodes to the `/opt/VRTSnas/conf/backup/` location.

3 Run the following command to see the available exported configuration:

```
system config list
```

4 Import the module configuration file by using the following command:

```
system config import local config-filename module-to-import
```

You can import the following modules:

Note: The module names are auto-suggested in the Veritas Access command-line interface.

Chapter 9

Migrating from scale-out and erasure-coded file systems

This chapter includes the following topics:

Preparing for migration

Veritas provides a utility to migrate from scale-out file system to CFS. You can use the same utility to migrate from erasure-coded file systems to non erasure-coded file system.

Perform the following steps to migrate your files from scale-out file system to CFS. You can follow the same steps to migrate from erasure-coded file systems to non erasure-coded file systems.

- Check the size of the source file system size using the following command:.

  ```
  storage> fs list
  ```

- Create a new file system with the desired layout. The size of the new file system should be ~10% greater than the source file system.

  ```
  storage> fs create
  ```

- Run the script for migration on any one of the nodes for each file system. Refer to the *Access Appliance Upgrade Guide* for more details.
- In case a cloud tier has been configured, move the data present in the cloud tier to the primary tier before you start migration.

  ```
  # storage fs policy add operation=move policy_name
  fs_name from_tier to_tier retrieval_option=Expedited|Standard|Bulk
  pattern [atime condition] [mtime condition]
  ```

- Stop and delete all the shares in the source scale-out file system. You have to create the shares again with the same settings in the destination CFS after migration is completed successfully.

Migration of data

You can migrate your data by performing the following steps:

- Use `migration_tool.py` to migrate files from scale-out file system to CFS.

  `./migration_tool.py -src src_filesystem -dest dest_filesystem`

  where

src_filesystem	Specifies the name of the source scale-out file system from which you want to migrate files
dest_filesystem	Specifies the name of the destination CFS to which you want to migrate the files.

- The script copies all the files with their data as well as attributes.
- Log are created in the `/opt/VRTSnas/log/` directory with *migrate_<src_file><dest_file><timestamp>.log* format.

Note: In case of any failures during migration, remove all the files from the destination CFS and start the transfer again using the script.

Migration of file systems which are exported as shares

Perform the following steps to migrate file systems which are exported as shares.

- Check for the presence of shares for the given scale-out file system.

  ```
  storage> fs list
  ```

- Save the share settings.
- Stop the shares and delete the shares present in the scale-out file system.
- Create the shares with the same settings in the destination CFS after migration is complete.

Migrating NFS shares

To migrate NFS shares

1. Save the settings of the existing shares and use he same settings to recreate the shares on the destination file system.

   For NFS, scale-out file system uses NFS Ganesha (GNFS). You have to switch to the Kernel NFS (KNFS) server for the destination CFS.

   Use the following command to find out share information like nfs options, export dir and hostname.

   ```
   nfs> share show
   ```

2. Unexport the share using the following command:

   ```
   nfs> share delete <export_dir>
   ```

3. Stop the NFS server using the following command:

   ```
   nfs> server stop
   ```

4. Recreate the NFS on the destination CFS. Start the NFS server in KNFS mode. Switch to KNFS if it is in GNFS.

   ```
   nfs> server status

   nfs> server switch
   ```

5 Start the server.

```
nfs> server start
```

6 Add the share to the destination CFS using the same settings as in the source scale-out file system.

```
nfs> share add <nfsoptions> <export_dir> [client]
```

For example:

```
nfs> share add rw,async /vx/scaleoutfs1
```

Migrating CIFS shares

To migrate CIFS shares

1 Save the existing configuration.

```
cifs> show
Name                    Value
----                    -----
netbios name            vmclus1
ntlm auth               yes
allow trusted domains   no
homedirfs
aio size                0
idmap backend           rid:10000-1000000
workgroup               WORKGROUP
security                user
Domain
Domain user
Domain Controller
Clustering Mode         normal
Data Migration          no
```

2 Make a note of the share name.

```
cifs> share show
ShareName    File System  Share Options
=========== ==========    ============================================
mycifsshare lfs2          owner=root,group=root,fs_mode=1777,rw,full_acl
```

3 Make a note of the homedir, if any.

```
cifs> homedir show
```

4 Get the list of local users.

```
cifs> local user show
List of Users
-------------
admin
user1
```

5 Get the local group.

```
cifs> local group show
List of groups
-------------
nogroup
selftest
mygrp
```

6 Create a new CIFS share on the destination CFS. Check the status of the CIFS server.

```
cifs> server status
```

7 Start the CIFS server.

```
cifs> server start
```

8 Add the file system to CIFS by specifying the name of the CFS, share name and modes.

```
cifs> share add file_system sharename [@virtual_ip]
[cifsoptions]
```

For example:

```
cifs> share add mycifs myshare ro
```

```
cifs> share show
```

9 Allow the user to access the share.

```
cifs> share allow sharename @group1
[,@group2,user1,user2,...]
```

For example:

```
cifs> share allow myshare user1
```

Migrating S3 shares

To migrate the existing buckets on the scale-out file system

1. Get the name of the file system on which the bucket was created.

```
objectaccess> bucket show
Bucket Name FileSystem     Pool(s) Owner
=========== ============== ======= =====
cloud1      S3fs1611925982 mypool  root
```

2. Create the new CFS.
3. Copy the data of the old file system on which the bucket was created to the new file system using the `migration_tool.py` script.
4. When you migrate theS3 bucket from a scale-out file system to CFS, the mapping of existing bucket to the new file system is not possible as it already exists in the S3 database. The `unconfig_s3bucket.py` script removes the existing entry of the scale-out file system from the S3 database to allow the mapping of the bucket to the new CFS directory path.

   ./unconfig_s3bucket.py *fs_name*

   where *fs_name* is the name of the file system to be removed from the S3 bucket mapping.
5. After the entry is removed from the S3 database, use the `objectaccess map` command to map the new CFS directory.

   `objectaccess>` **map *fs_path user_name***

6. Delete the old scale-out file system bucket, if required. This operation does not delete the source file system.

To chage the S3 configuration for future buckets if the fs_type was set to largefs

1. Check the current settings of th eS3 server.

   `objectaccess>` **show**

2. If the `fs_type` is set to `largefs`, set the `fs_type` to the desired layout.

   `objectaccess>` **set fs_type**

To recreate the bucket:

1. Set the default pool.

   ```
   objectaccess> set pools pool_name
   ```

2. Enable the server.

   ```
   ObjectAccess> server enable
   ```

3. Start the server.

   ```
   ObjectAccess> server start
   ```

4. Set the file system size, as required.

   ```
   ObjectAccess> set fs_size size
   ```

5. Set the file system type and layout for the CFS.

   ```
   ObjectAccess> set fs_type layout
   ```

6. Create keys for user authentication and save the access key and secret key.

   ```
   /opt/VRTSnas/scripts/utils/objectaccess/objectaccess_client.py
   --create_key --server ADMIN_URL --username root --password
   P@ssw0rd --insecure
   ```

   The *ADMIN_URL* is *admin.<cluster_name>:port*. The port is 8144. This url should point to the Access Appliance management console IP address.

7. Map the bucket to the existing file system.

   ```
   Objectaccess> map fs_path user_name
   ```

Chapter 10

Migrating LLT over Ethernet to LLT over UDP

This chapter includes the following topics:

Overview of migrating LLT to UDP

Veritas Access uses Low Latency Transport (LLT) for data transfer between applications on nodes. LLT functions as a high-performance, low-latency replacement for the IP stack, and is used for all cluster communications. If your Access cluster includes Direct Attached Storage (DAS) disks, for improved performance, Veritas recommends that you configure LLT over UDP.

This section explains how to migrate a cluster where LLT is already configured over Ethernet to LLT over UDP by utilizing an additional private NIC and an additional network subnet. You must restart the cluster after completing the migration steps.

Note: At any point of time, if the expectations set in this document don't match those in the cluster, please stop immediately and contact Veritas Technical Support for further assistance.

Limitations

- The steps described in this document for LLT migration are applicable only for Veritas Access 7.4.2.300 and Veritas Access 7.4.2.301 and should not be used for any other releases.

- After a cluster is migrated from LLT over Ethernet to LLT over UDP for Access 7.4.2.300 or 7.4.2.301 release, the add node operation on the same release is not expected to work. The cluster must be upgraded to at least version 7.4.2.400 for adding a node to the cluster.

Prerequisites

- “root” access to log in to the nodes and complete the steps listed in this document.
- An additional private network subnet, which has the same number of IP addresses as the existing private network subnet.

Migrating LLT to UDP

The following section describes how to migrate LLT to UDP. For reference, this guide assumes **ens161** to be the first private NIC, which already has an IP address assigned to it, and **ens256** to be second private NIC, which is unconfigured initially.

Note: The actual NIC names may vary depending on the naming scheme.

Step 1: Back up files (to be done on each node separately)

Log in to the node with root privileges and back up the following files as shown below:

```
# cp /etc/llttab /etc/llttab.llteth
# cp /opt/VRTSnas/conf/net_priv_dev.conf
/opt/VRTSnas/conf/net_priv_dev.conf.llteth
# cp /opt/VRTSnas/conf/net_priv_nic.conf
/opt/VRTSnas/conf/net_priv_nic.conf.llteth
# cp /opt/VRTSnas/conf/net_priv_ip_list.conf
/opt/VRTSnas/conf/net_priv_ip_list.conf.llteth
# cp /opt/VRTSnas/nodeconf/nasinstall.conf
/opt/VRTSnas/nodeconf/nasinstall.conf.llteth
```

Step 2: Set iptables (to be done on each node separately)

1. List the first two private NICs from the `/opt/VRTSnas/conf/net_priv_nic.conf` file and add the NICs to the `/opt/VRTSnas/conf/net_priv_dev.conf` file.

   After adding the first two private NICs, the `/opt/VRTSnas/conf/net_priv_dev.conf` file includes both the private NICs as shown below:

   ```
   # cat /opt/VRTSnas/conf/net_priv_dev.conf
   ens161 ens256
   #
   ```

   The only NIC that was already present in `/opt/VRTSnas/conf/net_priv_dev.conf` will be referred as the “first nic”, and the NIC that is newly added will be referred as the “second nic”.

2. Identify six ports to be used for communication over UDP.

   For example: 50000, 50001, 50002, 50003, 50004, 50005 and append them as shown below in `/opt/VRTSnas/nodeconf/nasinstall.conf` along with PORT_LLTLINK1 and PORT_LLTLINK2:

   ```
   PORT_LLTLINK1="ens161"
   PORT_LLTLINK2="ens256"
   LLT_UDPPORT_LIST="50000,50001,50002,50003,50004,50005"
   ```

3. Create a new system service `/etc/systemd/system/nas_pre_vx.service` as shown below:

   ```
   # cat /etc/systemd/system/nas_pre_vx.service
   [Unit]
   Description=This service will start before veki, llt
   After=network.target network-online.target
   Requires=network-online.target network.target
   Before=vxvm-boot.service veki.service llt.service
   [Service]
   Type=oneshot
   RemainAfterExit=yes
   TimeoutStartSec=300s
   ExecStart=/opt/VRTSnas/scripts/misc/nas_pre_vx.sh start
   ExecStop=/opt/VRTSnas/scripts/misc/nas_pre_vx.sh stop
   [Install]
   WantedBy=multi-user.target
   #
   ```

4 Create the script `/opt/VRTSnas/scripts/misc/nas_pre_vx.sh` to be run via `systemd` service exactly as shown below:

```
# cat /opt/VRTSnas/scripts/misc/nas_pre_vx.sh
#!/bin/bash

op=$1
if [[ $op == "start" ]]; then
        key="I"
elif [[ $op == "stop" ]]; then
        key="D"
elif [[ $op == "restart" ]]; then
        /opt/VRTSnas/scripts/misc/nas_pre_vx.sh stop
        /opt/VRTSnas/scripts/misc/nas_pre_vx.sh start
        exit 0
fi

nics=`cat /opt/VRTSnas/conf/net_priv_dev.conf`
ports=`grep LLT_UDPPORT_LIST /opt/VRTSnas/nodeconf/nasinstall.conf |
cut -d '=' -f2 | tr -s ',' ' ' | tr -d '"'`

for nic in $nics
do
  for port in $ports
  do
    iptables -w -${key} INPUT -i $nic -p udp -m udp --dport $port -j
    ACCEPT 2>/dev/null
    iptables -w -${key} OUTPUT -p udp -m udp --sport $port -j
    ACCEPT 2>/dev/null
    ip6tables -w -${key} INPUT -i $nic -p udp -m udp --dport $port -j
    ACCEPT 2>/dev/null
    ip6tables -w -${key} OUTPUT -p udp -m udp --sport $port -j
    ACCEPT 2>/dev/null
  done
done

exit 0
#
```

5 Add the line **/opt/VRTSnas/scripts/misc/nas_pre_vx.sh start** towards the end in the script `/opt/VRTSnas/scripts/net/net_iptables.sh` as shown below:

```
        #
        # unknown option
        #

        refresh
        /opt/VRTSnas/scripts/misc/nas_pre_vx.sh start
esac
set +x

exit 0
```

6 Mark the script as executable:

```
# chmod +x /opt/VRTSnas/scripts/misc/nas_pre_vx.sh
```

7 Start and enable this newly created service:

```
# systemctl start nas_pre_vx.service
# systemctl enable nas_pre_vx.service
```

Step 3: Configure the second private NIC

1 From any one of the nodes, find the NLM IP from `/opt/VRTSnas/nodeconf/nasinstall.conf`. The entry should look as shown below:

```
NLMMASTERIP="172.16.0.2"
```

2 Derive the new separate private subnet with netmask 255.255.255.0 based on the NLM IP by choosing the next 255.255.255.0 based subnet. For the above IP 172.16.0.2, the new subnet will be 172.16.1.1/255.255.255.0.

3 Create 32 entries using the first 32 IPs from this newly derived subnet (172.16.1.1 to 172.16.1.32) and add all the 32 lines to `/opt/VRTSnas/conf/net_priv_ip_list.conf` along with the netmask on each node. These entries will be exactly the same on each node:

Before the changes the file is as shown below:

```
172.16.0.3 255.255.255.0 node1 ens161
172.16.0.4 255.255.255.0 node2 ens161
172.16.0.5 255.255.255.0
172.16.0.6 255.255.255.0
.
.
172.16.0.34 255.255.255.0
```

After the changes, the file is as shown below:

```
172.16.0.3 255.255.255.0 node1 ens161
172.16.0.4 255.255.255.0 node2 ens161
172.16.0.5 255.255.255.0
172.16.0.6 255.255.255.0
.
.
172.16.0.34 255.255.255.0
172.16.1.1 255.255.255.0
172.16.1.2 255.255.255.0
172.16.1.3 255.255.255.0
172.16.1.4 255.255.255.0
.
.
172.16.1.31 255.255.255.0
172.16.1.32 255.255.255.0
```

4 Select each IP address from the above list of the newly derived subnet and configure the second NIC on each node by creating/modifying its `/etc/sysconfig/network-scripts/ifcfg-`*`second nic`* file. After updating this file, it should look as shown below:

On node 1:

```
DEVICE=second nic
BOOTPROTO=none
TYPE=Ethernet
NM_CONTROLLED=no
HWADDR=MAC Address of second nic on node1
ONBOOT=yes
IPADDR=172.16.1.1
NETMASK=255.255.255.0
```

On node 2:

```
DEVICE=second nic
BOOTPROTO=none
TYPE=Ethernet
NM_CONTROLLED=no
HWADDR=MAC Address of second nic on node2
ONBOOT=yes
IPADDR=172.16.1.2
NETMASK=255.255.255.0
```

5 Bring the newly configured second NIC online by running the following commands:

```
ifdown second nic
ifup second nic
```

For example:

```
# ifdown ens256
# ifup ens256
```

6 Check if the IP address is assigned correctly to the interface by using the command:

```
# ip addr show second nic
```

For example, on node 1:

```
# ip addr show ens256
5: ens256: BROADCAST,MULTICAST,UP,LOWER_UP mtu 1500 qdisc mq state
 UP group default qlen 1000
link/ether 00:50:56:9c:78:bc brd ff:ff:ff:ff:ff:ff
inet 172.16.1.1/24 brd 172.16.1.255 scope global ens256
valid_lft forever preferred_lft forever
inet6 fe80::250:56ff:fe9c:78bc/64 scope link
valid_lft forever preferred_lft forever
#
```

On node 2

```
[root@node2 ~]# ip addr show ens256
5: ens256: BROADCAST,MULTICAST,UP,LOWER_UP mtu 1500 qdisc mq state
 UP group default qlen 1000
link/ether 00:50:56:9c:61:79 brd ff:ff:ff:ff:ff:ff
inet 172.16.1.2/24 brd 172.16.1.255 scope global ens256
valid_lft forever preferred_lft forever
inet6 fe80::250:56ff:fe9c:6179/64 scope link
valid_lft forever preferred_lft forever
[root@node2 ~]#
```

7 Update `/opt/VRTSnas/conf/net_priv_ip_list.conf` with the correct details about the IP address, node name and NIC name as shown below:

Before the change the file is as shown below:

```
172.16.0.3 255.255.255.0 node1 ens161
172.16.0.4 255.255.255.0 node2 ens161
172.16.0.5 255.255.255.0
.
.
172.16.0.34 255.255.255.0
172.16.1.1 255.255.255.0
172.16.1.2 255.255.255.0
172.16.1.3 255.255.255.0
.
.
172.16.1.32 255.255.255.0
```

After changes the file is as shown below:

```
172.16.0.3 255.255.255.0 node1 ens161
172.16.0.4 255.255.255.0 node2 ens161
172.16.0.5 255.255.255.0
.
.
172.16.0.34 255.255.255.0
172.16.1.1 255.255.255.0 node1 ens256
172.16.1.2 255.255.255.0 node2 ens256
172.16.1.3 255.255.255.0
172.16.1.4 255.255.255.0
.
.
172.16.1.31 255.255.255.0
172.16.1.32 255.255.255.0
```

Step 4: Set up the LLT configuration file

1 Remove the lines containing “eth” from the `/etc/llttab` file on each node.

Before removing the lines, the file looks similar to as shown below:

```
# cat /etc/llttab
set-node node1
set-cluster 14358
link ens161 eth-00:50:56:9c:6e:c6 - ether - -
link ens256 eth-00:50:56:9c:78:bc - ether - -
set-flow highwater:10000
set-flow lowwater:8000
#
```

After removing the lines:

```
# cat /etc/llttab
set-node node1
set-cluster 14358
set-flow highwater:10000
set-flow lowwater:8000
#
```

2 Copy the below code to /tmp/llt.sh on each node:

```
# cat /tmp/llt.sh
#!/bin/bash
declare -A nics; set_addr_lines=""; link_lines=""; i=0;
IFS='='; read -ra udp_ports_str_arr <<< "$(grep LLT_UDPPORT_LIST
/opt/VRTSnas/nodeconf/nasinstall.conf)";
udp_ports_str=`echo ${udp_ports_str_arr[1]} | sed 's/"//g'`;
IFS=","; read -ra udp_ports_arr <<< "$udp_ports_str";
IFS=' '; for nic in  $(cat /opt/VRTSnas/conf/net_priv_nic.conf); do
               nics[$nic]=${udp_ports_arr[i]};i+=1;
done
IFS=$'\n'; for llthosts in $(cat /etc/llthosts);
do
  IFS=' ';
  read -ra hosts <<< "$llthosts";
  nodeid=${hosts[0]};node=${hosts[1]};
   IFS=$'\n'; for line in $(grep $node /opt/VRTSnas/conf/net_priv_ip_list
   .conf); do
     IFS=' ';
     read -ra priv_ips <<< "$line";
     ip=${priv_ips[0]};nic=${priv_ips[3]};
       if [[ "$node" = `hostname` ]];
     then
           link_lines=$link_lines"link $nic udp - udp ${nics[$nic]} - $ip
           -"$'\n';
     else
           set_addr_lines=$set_addr_lines"set-addr $nodeid $nic $ip"$'\n'
     fi
  done
done
echo $link_lines | awk NF;
echo $set_addr_lines | awk NF;
#
```

3 On each node, mark this script as executable and run it. This script produces output as shown below. This output is very specific to the node on which the script was executed.

On node 1:

```
# chmod +x /tmp/llt.sh
# /tmp/llt.sh
link ens161 udp - udp 50000 - 172.16.0.3 -
link ens256 udp - udp 50001 - 172.16.1.1 -
set-addr 1 ens161 172.16.0.4
set-addr 1 ens256 172.16.1.2
#
```

On node 2:

```
[root@node2 ~]# chmod +x /tmp/llt.sh
[root@node2 ~]# /tmp/llt.sh
link ens161 udp - udp 50000 - 172.16.0.4 -
link ens256 udp - udp 50001 - 172.16.1.2 -
set-addr 0 ens161 172.16.0.3
set-addr 0 ens256 172.16.1.1
[root@node2 ~]#
```

4 Use the output produced by the script on the same node where the script was executed and copy this output in `/etc/llttab` of the respective node, immediately after the “set-cluster” line. The `/etc/llttab` file should look as shown below after copying the output.

On node 1:

```
# cat /etc/llttab
set-node node1
set-cluster 14358
link ens161 udp - udp 50000 - 172.16.0.3 -
link ens256 udp - udp 50001 - 172.16.1.1 -
set-addr 1 ens161 172.16.0.4
set-addr 1 ens256 172.16.1.2
set-flow highwater:10000
set-flow lowwater:8000
#
```

On node 2:

```
# cat /etc/llttab
set-node node2
set-cluster 14358
link ens161 udp - udp 50000 - 172.16.0.4 -
link ens256 udp - udp 50001 - 172.16.1.2 -
set-addr 0 ens161 172.16.0.3
set-addr 0 ens256 172.16.1.1
set-flow highwater:10000
set-flow lowwater:8000
#
```

5 On each node, append the below tunables to the `/etc/llttab`. These attributes are common and not specific to any node:

```
set-udpsockets 4
set-udpthreads 2
set-bcasthb 0
set-arp 0
```

Step 5: Reboot the cluster

1 Reboot all the nodes one by one using the reboot command:

```
# reboot
[root@node2 ~]# reboot
```

Wait for the nodes to come up. After the nodes are up and the cluster is formed, validate as shown below.

2 Verify that the status of all the NICs is UP:

```
# lltstat -nvv active
LLT node information:
Node State Link Status Address
* 0 node1 OPEN
ens161 UP 172.16.0.3
ens256 UP 172.16.1.1
1 node2 OPEN
ens161 UP 172.16.0.4
ens256 UP 172.16.1.2
#
```

3 Verify that the cluster should not be in jeopardy at this stage. If the cluster is not in jeopardy, it should look as shown below:

```
# gabconfig -a
GAB Port Memberships
===============================================================
Port a gen 184402 membership 01
Port b gen 184401 membership 01
Port f gen 184410 membership 01
Port h gen 184405 membership 01
Port m gen 184407 membership 01
Port u gen 18440e membership 01
Port v gen 184409 membership 01
Port w gen 18440b membership 01
Port y gen 184408 membership 01
#
```

If the cluster is in jeopardy, it will look as shown below:

```
# gabconfig -a
GAB Port Memberships
===============================================================
Port a gen 44c401 membership 01
Port a gen 44c401 jeopardy ;1
Port b gen 44c403 membership 01
Port b gen 44c403 jeopardy ;1
Port f gen 44c410 membership 01
Port f gen 44c410 jeopardy ;1
Port h gen 44c406 membership 01
Port h gen 44c406 jeopardy ;1
Port m gen 44c408 membership 01
Port m gen 44c408 jeopardy ;1
Port u gen 44c40e membership 01
Port u gen 44c40e jeopardy ;1
Port v gen 44c409 membership 01
Port v gen 44c409 jeopardy ;1
Port w gen 44c40c membership 01
Port w gen 44c40c jeopardy ;1
Port y gen 44c40a membership 01
Port y gen 44c40a jeopardy ;1
#
```

Chapter 11

Performing a rolling upgrade

This chapter includes the following topics:

About rolling upgrade

The Veritas Access 7.4.2.400 supports rolling upgrade from Veritas Access version 7.4.2.300 and 7.4.2.301. Rolling upgrade is supported on RHEL 7.7.

A rolling upgrade minimizes the service and application downtime for highly available clusters by limiting the upgrade time to the amount of time that it takes to perform a service group failover. Nodes with different product versions can be run in one cluster.

The rolling upgrade has two main phases. The installer upgrades kernel RPMs in phase 1 and VCS agent RPMs in phase 2. Upgrade should be done on each node individually one by one. You need to perform upgrade first on an each slave node and thereafter on the master node. The upgrade process stops all services and resources on the node, which is being upgraded. All services (including the VIP groups) fail over to the one of the other node from the cluster. During the failover process, the clients that are connected to the VIP groups of nodes are intermittently interrupted. For those clients that do not time-out, the service is resumed after the VIP groups become Online on the node that is being upgraded.

While the upgrade process is running on the first node, other nodes of the cluster continues to serve the clients. After the first node has been upgraded, it restarts the services and resources on the first-stage node. After the first node comes up, the upgrade process stops the services and resources on the next slave node and

so on. All services and resources are online and serve clients. Meanwhile, the rolling upgrade starts the upgrade process on the remaining nodes. After the upgrade is complete on the remaining nodes, the cluster recovers and services are balanced across the cluster.

Workflow for rolling upgrade

A rolling upgrade has two main phases where the installer upgrades the kernel RPMs in Phase 1 and VCS agent-related non-kernel RPMs in Phase 2.

1. The upgrade process is performed on each node one after another.
2. In phase 1, the upgrade process is performed first on the slave node(s) and then on the master node. The upgrade process stops all services on the node and failover service group to an another node in the cluster.
3. During the failover process, the clients that are connected to the VIP groups of the nodes are intermittently interrupted. For those clients that do not time out, the service is resumed after the VIP groups become online on one of the nodes.
4. The installer upgrades the kernel RPMs on the node. The nodes continue to serve the clients.
5. After the phase 1 for first slave node is complete, upgrade is started for the second slave node and so on. After slave nodes master node is upgraded. And all the service groups from master node failover to some other node.
6. After phase 1 for first node is successful, you need to check if recovery task is also complete before starting upgrade phase 1 for the next node.

   Note: Make sure that the upgraded node is not out of the cluster. If the node is out of cluster, wait for the node to join the existing cluster.

7. During Phase 2 of the rolling upgrade, all remaining RPMs are upgraded on all the nodes of the cluster simultaneously. VCS and VCS-agent packages are upgraded. The kernel drivers are upgraded to the new protocol version. Applications stay online during Phase 2. The High Availability Daemon (HAD) stops and starts again.

See “Performing a rolling upgrade using the installer” on page 151.

Performing a rolling upgrade using the installer

Note: See the "Known issues> Upgrade issues" section of the *Veritas Access Release Notes* before starting the rolling upgrade.

Before you start a rolling upgrade, ensure that the following prerequisites are completed:

- The Veritas Cluster Server (VCS) is running on all the nodes of the cluster.
- Stop all activity for all the VxVM volumes that are not under VCS control. For example, stop any applications such as databases that access the volumes, and unmount any file systems that have been created on the volumes. Then stop all the volumes.
- Unmount all the VxFS file systems that are not under VCS control.
- Flexible Storage Sharing (FSS) on Ethernet, scale-out file system, or an erasure-coded file system is not configured. The pre-upgrade check fails if any one of these are configured during an upgrade to 7.4.2.400. To migrate files from scale-out file system to CFS or to migrate from erasure-coded file systems to non erasure-coded file system:
  See “Preparing for migration” on page 127.
  See “Migration of data” on page 128.
  See “Migration of file systems which are exported as shares” on page 128.
- LLT over Ethernet is no longer supported. If LLT is configured over Ethernet in the cluster, you must migrate LLT to UDP.
  See “Overview of migrating LLT to UDP” on page 134.
  See “Migrating LLT to UDP” on page 135.

Note: The Veritas Access GUI is not accessible from the time that you start rolling upgrade on the master node till the time rolling upgrade is complete.

Note: It is recommended that during rolling upgrade, you use only `list` and `show` commands in the Veritas Access command-line interfae. Using other commands like `create`, `destroy`, `add`, and `remove` may update the Veritas Access configuration which is not recommended during rolling upgrade.

Starting with version 7.4.2.400, the `upgrade` command is deprecated. You can continue to use the `installaccess` script to perform a rolling upgrade.

To perform a rolling upgrade

1. In case of the LTR-configured Veritas Access cluster, make sure that the backup or restore jobs from NetBackup are stopped.
2. Phase 1 of a rolling upgrade begins on the second subcluster. Complete the preparatory steps on the second subcluster.

   Unmount all VxFS file systems not under VCS control:

   ```
   # umount mount_point
   ```

3. Complete updates to the operating system, if required.

   Make sure that the existing version of Veritas Access supports the operating system update you apply. If the existing version of Veritas Access does not support the operating system update, first upgrade Veritas Access to a version that supports the operating system update.

   For instructions, see the Red Hat Enterprise Linux (RHEL) operating system documentation.

   Switch applications to the remaining subcluster and upgrade the operating system of the first subcluster.

   The nodes are restarted after the operating system update.
4. If a cache area is online, you must take the cache area offline before you upgrade the VxVM RPMs. Use the following command to take the cache area offline:

   ```
   # sfcache offline cachename
   ```

5. Log on as superuser and mount the Veritas Access 7.4.2.400 installation media.
6. From root, start the installer.

   ```
   # ./installaccess -rolling_upgrade
   ```

7. The installer checks system communications, release compatibility, version information, and lists the cluster name, ID, and cluster nodes. The installer asks for permission to proceed with the rolling upgrade.

   ```
   Would you like to perform rolling upgrade on the cluster? [y,n,q] (y)
   ```

   Type **y** to continue.

8 Phase 1 of the rolling upgrade begins. Phase 1 must be performed on one node at a time. The installer asks for the system name.

Enter the system names separated by spaces on which you want to perform rolling upgrade: [q?]

Enter the name or IP address of one of the slave nodes on which you want to perform the rolling upgrade.

9 The installer performs further prechecks on the nodes in the cluster and may present warnings. You can type **y** to continue or quit the installer and address the precheck's warnings.

10 If the boot disk is encapsulated and mirrored, you can create a backup boot disk.

If you choose to create a backup boot disk, type **y**. Provide a backup name for the boot disk group or accept the default name. The installer then creates a backup copy of the boot disk group.

11 After the installer detects the online service groups, the installer prompts the user to do one of the following:

- Manually switch service groups
- Use the CPI to automatically switch service groups

The downtime is the time that it takes for the failover of the service group.

Note: Veritas recommends that you manually switch the service groups. Automatic switching of service groups does not resolve dependency issues.

---

12 The installer prompts you to stop the applicable processes. Type **y** to continue.

The installer evacuates all service groups to the node or nodes that are not upgraded at this time. The installer stops parallel service groups on the nodes that are to be upgraded.

The installer stops all the related processes, uninstalls the old kernel RPMs, and installs the new RPMs.

13 The installer performs the upgrade configuration and starts the processes. If the boot disk is encapsulated before the upgrade, the installer prompts you to restart the node after performing the upgrade configuration.

14 Complete the preparatory steps on the nodes that you have not yet upgraded.

Unmount all the VxFS file systems that are not under the VCS control on all the nodes.

```
# umount mount_point
```

15 If the operating system updates are not required, skip this step.

Go to step 16 .

Else, complete updates to the operating system on the nodes that you have not yet upgraded. For the instructions, see the Red Hat Enterprise Linux (RHEL) operating system documentation.

Repeat steps 4 to 13 for each node.

16 After the upgrade of phase 1 is done on the node, make sure that the node is not out of the cluster.

Enter the `# vxclustadm nidmap` command.

If the upgraded node is out of the cluster, wait for the node to join the cluster before you start the upgrade of phase 1 for the next node.

17 Phase 1 of the rolling upgrade is complete for the first node. You can start with the upgrade of phase 1 for the next slave node. Installer again asks for the system name.

Before you start phase 1 of rolling upgrade for the next node, check if any recovery task is still in-progress. Wait for the recovery task to complete.

On the master node, enter the following command:

```
# vxtask list
Check if following keywords are present:
ECREBUILD/ATCOPY/ATCPY/PLXATT/VXRECOVER/RESYNC/RECOV
```

If any recovery task is in progress, wait for the task to complete, and then start for upgrade of phase 1 for the next node.

18 Set up all cache areas as offline on the remaining node or nodes:

```
# sfcache offline cachename
```

The installer asks for a node name on which upgrade is to be performed.

19 Enter the system names separated by spaces on which you want to perform rolling upgrade: [q,?].

Type cluster node name or **q** to quit.

The installer repeats step 8 through step 13 .

For clusters with larger number of nodes, this process may repeat several times. Service groups come down and are brought up to accommodate the upgrade.

20 When phase 1 of the rolling upgrade completes, mount all the VxFS file systems that are not under VCS control manually.

Before you start phase 2 of rolling upgrade, make sure that all the nodes have joined the cluster and all recovery tasks are complete.

Begin Phase 2 of the upgrade. Phase 2 of the upgrade includes downtime for the VCS engine (HAD), which does not include application downtime. Type **y** to continue. Phase 2 of the rolling upgrade begins here.

21 The installer determines the remaining RPMs to upgrade. Press **y** to continue.

22 The installer stops the Veritas Cluster Server (VCS) processes but the applications continue to run. Type **y** to continue.

The installer performs a prestop, uninstalls the old RPMs, and installs the new RPMs. It performs post-installation tasks, and the configuration for the upgrade.

23 If you have a network connection to the Internet, the installer checks for updates.

If updates are discovered, you can apply them now.

24 Verify the cluster's status:

```
# hastatus -sum
```

See “About rolling upgrade” on page 149.

Chapter 12

Uninstalling Veritas Access

This chapter includes the following topics:

Before you uninstall Veritas Access

Perform the following steps before uninstalling Veritas Access:

- Before you remove Veritas Access from any node (but not in all the nodes) in a cluster, make sure the node has already been deleted from the running cluster. You can use the `Cluster> show` command to view the cluster node state, and use the `Cluster> delete` command to delete a running node from the Veritas Access cluster.
  See the relevant man pages for more information on the `Cluster> show` and `Cluster> delete` commands.
- Stop all the applications that access the file system over NFS, CIFS, or FTP.
- Destroy all the replication jobs from the cluster.
  Use the `Replication> job show` command to list all the replication jobs on the cluster.

```
Replication> job show
Job Name Role Job Type Encryption Debug Schedule
======== ====== ======== ========== ===== ========
job1 SOURCE DATA OFF ON sch1
State CKPT Count Exclunit Source repunit Target repunit(s)
======== ========== ======== ============== =================
```

```
ENABLED 1 -- scr1 trg1
Link name(s)
============
link1
```

Use the `Replication> job destroy` command to destroy the replication jobs.

```
Replication> job destroy job1
ACCESS replication SUCCESS V-288-0 Removing bandwidth limit on the
link: link1
ACCESS replication SUCCESS V-288-0 Job 'job1' disabled successfully.
ACCESS replication SUCCESS V-288-0 Job 'job1' deleted successfully.
```

- Stop the NFS, CIFS, FTP, GUI, and the replication service on the cluster using the appropriate Veritas Access command-line interface command.

```
CLISH> cifs server stop
Stopping CIFS Server.....Success.
CLISH>
CLISH> nfs server stop
Success.
CLISH>
CLISH> ftp server stop
Success.
CLISH>
CLISH.Support> gui server stop
GUI service is OFFLINE.
CLISH>
CLISH> replication service stop
ACCESS replication SUCCESS V-288-0 Replication service stopped
CLISH>
```

- Run the following command to stop AMF:

```
# /etc/init.d/amf stop
Stopping AMF...
AMF: Module unloaded
```

- Run the following command and wait for a couple of minutes:

```
# /opt/VRTS/bin/hastop -all
```

- Run the following command and verify that you only see Port a and Port b:

```
# gabconfig -a
GAB Port Memberships
===============================
Port a gen 7f2d0a membership 01
Port b gen 7f2d09 membership 01
```

Uninstalling Veritas Access using the installer

You can perform an uninstallation of Veritas Access. The Veritas Access uninstall program lets you uninstall Veritas Access without requiring a reinstallation of the operating system. You can also use the uninstall program in cases where there was an incomplete installation of Veritas Access.

Before you use the uninstall program to uninstall Veritas Access on all nodes in the cluster at the same time, make sure that communication exists between the nodes. By default, Veritas Access cluster nodes can communicate with each other using ssh.

If the nodes cannot communicate with each other, then you must run the uninstall program on each node in the cluster. The uninstall program removes all Veritas Access RPMs.

Removing Veritas Access 7.4.2.400 RPMs

The uninstall program stops the Veritas Access processes that are currently running during the uninstallation process.

To uninstall Veritas Access 7.4.2.400 RPMs

1. Sign in as the support user from the node where you want to uninstall Veritas Access.
2. Start the uninstall program.

   ```
   # cd /opt/VRTS/install

   # ./uninstallaccess
   ```

   The program specifies the directory where the logs are created. The program displays a copyright notice and a description of the cluster.

3 Enter the IP addresses of the nodes from which you want to uninstall Veritas Access.

The program performs node verification checks and asks to stop all running Veritas Access processes.

4 Enter **y** to stop all the Veritas Access processes.

The program stops the Veritas Access processes and uninstalls the software.

The uninstall program does the following tasks:

- Verifies the communication between nodes.
- Checks the installations on each node to determine the RPMs to be uninstalled.
- Unloads kernel modules and removes the RPMs.

Review the output as the uninstaller stops processes.

You can make a note of the location of the summary, response, and log files that the uninstaller creates after removing all the RPMs.

Running uninstall from the Veritas Access 7.4.2.400 disc

You may need to use the uninstall program on the Veritas Access 7.4.2.400 disc in one of the following cases:

- You need to uninstall Veritas Access after an incomplete installation.
- The uninstall program is not available in `/opt/VRTS/install`.

If you mounted the installation media to `/mnt`, access the uninstall program by changing the directory.

```
cd /mnt/

./uninstallaccess
```

Appendix A

Installation reference

This appendix includes the following topics:

Installation script options

Table A-1 lists the available command line options for the Veritas Access installation script. For an initial install or upgrade, options are not usually required.

Table A-1 Available command line options

Command Line Option	Function
-configure	Configures an unconfigured product after it is installed.
-install	Installs the product on systems.
-precheck	Performs checks to confirm that systems have met the products installation requirements before installing the product.
-license	Registers or updates product licenses on the specified systems.
-licensefile	Specifies the location of the Veritas perpetual or subscription license key file.
-requirements	Displays the required operating system version, required patches, file system space, and other system requirements to install the product.

Table A-1 Available command line options *(continued)*

Command Line Option	Function
-responsefile *response_file*	Performs automated installations or uninstallations using information stored in a file rather than prompting for the information. *response_file* is the full path of the file that contains the configuration definitions.
-rolling_upgrade	Performs a rolling upgrade. Using this option, the installer detects the rolling upgrade status on cluster systems automatically without the need to specify rolling upgrade phase 1 or phase 2 explicitly.
-prestop_script *prestop_script*	Executes the customized script provided by user on each host before stop processes during the upgrade procedure.
-poststart_script *poststart_script*	Executes the customized script provided by user on each host after start processes during the upgrade procedure.
-uninstall	Uninstalls the product from systems.

Appendix B

Configuring the secure shell for communications

This appendix includes the following topics:

Manually configuring passwordless secure shell (ssh)

The secure shell (ssh) program enables you to log into and execute commands on a remote system. ssh enables encrypted communications and an authentication process between two untrusted hosts over an insecure network.

In this procedure, you first create a DSA key pair. From the key pair, you append the public key from the source system to the `authorized_keys` file on the target systems.

To create the DSA key pair

1. On the source system (sys1), log in as ***root***, and navigate to the root directory.

   ```
   sys1 # cd /root
   ```

2. To generate a DSA key pair on the source system, type the following command:

   ```
   sys1 # ssh-keygen -t dsa
   ```

   System output similar to the following is displayed:

   ```
   Generating public/private dsa key pair.
   Enter file in which to save the key (/root/.ssh/id_dsa):
   ```

3. Press Enter to accept the default location of `/root/.ssh/id_dsa`.

4. When the program asks you to enter the pass phrase, press the Enter key twice.

   ```
   Enter passphrase (empty for no passphrase):
   ```

   Do not enter a pass phrase. Press Enter.

   ```
   Enter same passphrase again:
   ```

   Press Enter again.

5. Output similar to the following lines appears.

   ```
   Your identification has been saved in /root/.ssh/id_dsa.
   Your public key has been saved in /root/.ssh/id_dsa.pub.
   The key fingerprint is:
   1f:00:e0:c2:9b:4e:29:b4:0b:6e:08:f8:50:de:48:d2 root@sys1
   ```

To append the public key from the source system to the authorized_keys file on the target system using secure file transfer

1. From the source system (sys1), move the public key to a temporary file on the target system (sys2).

   Use the secure file transfer program.

   In this example, the file name `id_dsa.pub` in the root directory is the name for the temporary file for the public key.

   Use the following command for secure file transfer:

   ```
   sys1 # sftp sys2
   ```

   If the secure file transfer is set up for the first time on this system, output similar to the following lines is displayed:

   ```
   Connecting to sys2 ...
   The authenticity of host 'sys2 (10.182.00.00)'
   can't be established. DSA key fingerprint is
   fb:6f:9f:61:91:9d:44:6b:87:86:ef:68:a6:fd:88:7d.
   Are you sure you want to continue connecting (yes/no)?
   ```

2. Enter `yes`.

   Output similar to the following is displayed:

   ```
   Warning: Permanently added 'sys2,10.182.00.00'
   (DSA) to the list of known hosts.
   root@sys2 password:
   ```

3. Enter the root password of sys2.
4. At the `sftp` prompt, type the following command:

   ```
   sftp> put /root/.ssh/id_dsa.pub
   ```

   The following output is displayed:

   ```
   Uploading /root/.ssh/id_dsa.pub to /root/id_dsa.pub
   ```

5. To quit the SFTP session, type the following command:

   ```
   sftp> quit
   ```

6 Add the `id_dsa.pub` keys to the `authorized_keys` file on the target system. To begin the `ssh` session on the target system (sys2 in this example), type the following command on sys1:

```
sys1 # ssh sys2
```

Enter the root password of sys2 at the prompt:

```
password:
```

Type the following commands on sys2:

```
sys2 # cat /root/id_dsa.pub >> /root/.ssh/authorized_keys
sys2 # rm  /root/id_dsa.pub
```

7 Run the following commands on the source installation system. If your ssh session has expired or terminated, you can also run these commands to renew the session. These commands bring the private key into the shell environment and make the key globally available to the *root* user.

```
sys1 # exec /usr/bin/ssh-agent $SHELL
sys1 # ssh-add

Identity added: /root/.ssh/id_dsa
```

This shell-specific step is valid only while the shell is active. You must execute the procedure again if you close the shell during the session.

To verify that you can connect to a target system

1 On the source system (sys1), enter the following command:

```
sys1 # ssh -l root sys2 uname -a
```

where sys2 is the name of the target system.

2 The command should execute from the source system (sys1) to the target system (sys2) without the system requesting a pass phrase or password.

3 Repeat this procedure for each target system.

Setting up ssh and rsh connections using the pwdutil.pl utility

The password utility, `pwdutil.pl`, is bundled in the 7.4.2.400 release in the `/opt/VRTS/repository/ga/images/SSNAS/7.4.2.400.0/scripts/pwdutil.pl`

directory. The users can run the utility in their script to set up the ssh and rsh connection automatically.

```
# ./pwdutil.pl -h
Usage:

Command syntax with simple format:

    pwdutil.pl check|configure|unconfigure ssh|rsh <hostname|IP addr>
    [<user>] [<password>] [<port>]

Command syntax with advanced format:

    pwdutil.pl [--action|-a 'check|configure|unconfigure']
               [--type|-t 'ssh|rsh']
               [--user|-u  '<user>']
               [--password|-p '<password>']
               [--port|-P '<port>']
               [--hostfile|-f '<hostfile>']
               [--keyfile|-k '<keyfile>']
               [-debug|-d]
               <host_URI>

    pwdutil.pl -h | -?
```

Table B-1 Options with pwdutil.pl utility

Option	Usage
--action|-a 'check|configure|unconfigure'	Specifies action type, default is 'check'.
--type|-t 'ssh|rsh'	Specifies connection type, default is 'ssh'.
--user|-u '<user>'	Specifies user id, default is the local user id.
--password|-p '<password>'	Specifies user password, default is the user id.
--port|-P '<port>'	Specifies port number for ssh connection, default is 22
--keyfile|-k '<keyfile>'	Specifies the private key file.
--hostfile|-f '<hostfile>'	Specifies the file which list the hosts.
-debug	Prints debug information.

Table B-1 Options with pwdutil.pl utility *(continued)*

Option	Usage
-h|-?	Prints help messages.
<host_URI>	Can be in the following formats: <hostname> <user>:<password>@<hostname> <user>:<password>@<hostname>: <port>

You can check, configure, and unconfigure ssh or rsh using the `pwdutil.pl`utility. For example:

- To check ssh connection for only one host:

  `pwdutil.pl check ssh hostname`

- To configure ssh for only one host:

  `pwdutil.pl configure ssh hostname user password`

- To unconfigure rsh for only one host:

  `pwdutil.pl unconfigure rsh hostname`

- To configure ssh for multiple hosts with same user ID and password:

  `pwdutil.pl -a configure -t ssh -u user -p password hostname1 hostname2 hostname3`

- To configure ssh or rsh for different hosts with different user ID and password:

  `pwdutil.pl -a configure -t ssh user1:password1@hostname1 user2:password2@hostname2`

- To check or configure ssh or rsh for multiple hosts with one configuration file:

  `pwdutil.pl -a configure -t ssh --hostfile /tmp/sshrsh_hostfile`

- To keep the host configuration file secret, you can use the 3rd party utility to encrypt and decrypt the host file with password.
  For example:

```
### run openssl to encrypt the host file in base64 format
# openssl aes-256-cbc -a -salt -in /hostfile -out /hostfile.enc
enter aes-256-cbc encryption password: <password>
Verifying - enter aes-256-cbc encryption password: <password>

### remove the original plain text file
# rm /hostfile

### run openssl to decrypt the encrypted host file
# pwdutil.pl -a configure -t ssh `openssl aes-256-cbc -d -a
-in /hostfile.enc`
enter aes-256-cbc decryption password: <password>
```

- To use the ssh authentication keys which are not under the default $HOME/.ssh directory, you can use --keyfile option to specify the ssh keys. For example:

```
### create a directory to host the key pairs:
# mkdir /keystore

### generate private and public key pair under the directory:
# ssh-keygen -t rsa -f /keystore/id_rsa

### setup ssh connection with the new generated key pair under
the directory:
# pwdutil.pl -a configure -t ssh --keyfile /keystore/id_rsa
user:password@hostname
```

You can see the contents of the configuration file by using the following command:

```
# cat /tmp/sshrsh_hostfile
user1:password1@hostname1
user2:password2@hostname2
user3:password3@hostname3
user4:password4@hostname4

# all default: check ssh connection with local user
hostname5
The following exit values are returned:

0     Successful completion.
1     Command syntax error.
2     Ssh or rsh binaries do not exist.
3     Ssh or rsh service is down on the remote machine.
```

```
4     Ssh or rsh command execution is denied due to password is required.
5     Invalid password is provided.
255   Other unknown error.
```

Appendix C

Manual deployment of Veritas Access

This appendix includes the following topics:

Deploying Veritas Access manually on a two-node cluster in a non-SSH environment

This section describes the manual steps for deploying a two-node Veritas Access cluster when SSH communication is disabled.

Pre-requisites

- Consider a two-node cluster.
- Supported operating system version is: RHEL 7 update 7 and Update 9
- It is assumed that Veritas Access image is present in your local system at the `/access_build_dir/rhel7_x86_64/` location.
- The cluster is named as *clus* and the cluster nodes are named as *clus-01* and *clus-02*. Cluster names should be unique for all nodes.
- SSH service is stopped on all nodes.
- Assume that the public NICs are pubeth0, pubeth1, and private NICs are priveth0 and priveth1. NIC names should be consistent across all nodes. Public NIC names and private NIC names should be same across all nodes.

- Use `172.16.0.3` as private IP address for *clus-01* and `172.16.0.4` as private IP address for *clus-02*.

To deploy Veritas Access manually on a two-node cluster

1. Copy the Veritas Access image on all nodes of the desired cluster.
2. Stop the SSH daemon on all the nodes.

   ```
   # systemctl stop sshd
   ```

3. Verify if the following rpms are installed. If not, install the rpms from the RHEL repository.

   ```
   bash-4.2.46-28.el7.x86_64
   lsscsi-0.27-6.el7.x86_64
   initscripts-9.49.39-1.el7.x86_64
   iproute-3.10.0-87.el7.x86_64
   kmod-20-15.el7.x86_64
   coreutils-8.22-18.el7.x86_64
   binutils-2.25.1-31.base.el7.x86_64
   python-requests-2.6.0-1.el7_1.noarch
   python-urllib3-1.10.2-3.el7.noarch
   ```

4. Install the required operating system rpms.
   - Create a `repo` file.

     ```
     cat /etc/yum.repos.d/os.repo
       [veritas-access-os-rpms]
       name=Veritas Access OS RPMS
       baseurl=file:///access_build_dir/rhel7_x86_64/os_rpms/
       enabled=1
       gpgcheck=0
     ```

   - Run the following command:

     ```
     # yum updateinfo
     ```

   - Run the following command:

     ```
     # cd /access_build_dir/rhel7_x86_64/os_rpms/
     ```

   - Before running the following command, make sure that there is no RHEL subscription in the system. The `yum repolist` should point to `veritas-access-os-rpms` only.

```
# /usr/bin/yum -y install --setopt=protected_multilib=false
perl-5.16.3-292.el7.x86_64.rpm nmap-ncat-6.40-7.el7.x86_64.rpm
perl-LDAP-0.56-5.el7.noarch.rpm perl-Convert-ASN1-0.26-4.el7.noarch.rpm
net-snmp-5.7.2-28.el7_4.1.x86_64.rpm
net-snmp-utils-5.7.2-28.el7_4.1.x86_64.rpm
openldap-2.4.44-5.el7.x86_64.rpm nss-pam-ldapd-0.8.13-8.el7.x86_64.rpm
rrdtool-1.4.8-9.el7.x86_64.rpm wireshark-1.10.14-14.el7.x86_64.rpm
vsftpd-3.0.2-22.el7.x86_64.rpm openssl-1.0.2k-12.el7.x86_64.rpm
openssl-devel-1.0.2k-12.el7.x86_64.rpm
iscsi-initiator-utils-6.2.0.874-4.el7.x86_64.rpm
libpcap-1.5.3-9.el7.x86_64.rpm libtirpc-0.2.4-0.10.el7.x86_64.rpm
nfs-utils-1.3.0-0.48.el7_4.2.x86_64.rpm
kernel-debuginfo-common-x86_64-3.10.0-693.el7.x86_64.rpm
kernel-debuginfo-3.10.0-693.el7.x86_64.rpm
kernel-headers-3.10.0-693.el7.x86_64.rpm
krb5-devel-1.15.1-8.el7.x86_64.rpm
krb5-libs-1.15.1-8.el7.x86_64.rpm
krb5-workstation-1.15.1-8.el7.x86_64.rpm
perl-JSON-2.59-2.el7.noarch.rpm telnet-0.17-64.el7.x86_64.rpm
apr-devel-1.4.8-3.el7_4.1.x86_64.rpm
apr-util-devel-1.5.2-6.el7.x86_64.rpm
glibc-common-2.17-196.el7_4.2.x86_64.rpm
glibc-headers-2.17-196.el7_4.2.x86_64.rpm
glibc-2.17-196.el7_4.2.x86_64.rpm glibc-2.17-196.el7_4.2.i686.rpm
glibc-devel-2.17-196.el7_4.2.x86_64.rpm
glibc-utils-2.17-196.el7_4.2.x86_64.rpm
nscd-2.17-196.el7_4.2.x86_64.rpm sysstat-10.1.5-12.el7.x86_64.rpm
libibverbs-utils-13-7.el7.x86_64.rpm libibumad-13-7.el7.x86_64.rpm
opensm-3.3.19-1.el7.x86_64.rpm opensm-libs-3.3.19-1.el7.x86_64.rpm
infiniband-diags-1.6.7-1.el7.x86_64.rpm
sg3_utils-libs-1.37-12.el7.x86_64.rpm sg3_utils-1.37-12.el7.x86_64.rpm
libyaml-0.1.4-11.el7_0.x86_64.rpm
memcached-1.4.15-10.el7_3.1.x86_64.rpm
python-memcached-1.59-1.noarch
python-paramiko-2.1.1-4.el7.noarch.rpm
python-backports-1.0-8.el7.x86_64.rpm
python-backports-ssl_match_hostname-3.4.0.2-4.el7.noarch.rpm
python-chardet-2.2.1-1.el7_1.noarch.rpm
python-six-1.9.0-2.el7.noarch.rpm
python-setuptools-0.9.8-7.el7.noarch.rpm
python-ipaddress-1.0.16-2.el7.noarch.rpm
targetcli-2.1.fb46-1.el7.noarch.rpm
```

```
fuse-2.9.2-8.el7.x86_64.rpm fuse-devel-2.9.2-8.el7.x86_64.rpm
fuse-libs-2.9.2-8.el7.x86_64.rpm PyYAML-3.10-11.el7.x86_64.rpm
arptables-0.0.4-8.el7.x86_64.rpm ipvsadm-1.27-7.el7.x86_64.rpm
ntpdate-4.2.6p5-25.el7_3.2.x86_64.rpm ntp-4.2.6p5-25.el7_3.2.x86_64.rpm
autogen-libopts-5.18-5.el7.x86_64.rpm ethtool-4.8-1.el7.x86_64.rpm
net-tools-2.0-0.22.20131004git.el7.x86_64.rpm
cups-libs-1.6.3-29.el7.x86_64.rpm avahi-libs-0.6.31-17.el7.x86_64.rpm
psmisc-22.20-15.el7.x86_64.rpm strace-4.12-4.el7.x86_64.rpm
vim-enhanced-7.4.160-2.el7.x86_64.rpm at-3.1.13-22.el7_4.2.x86_64.rpm
rsh-0.17-76.el7_1.1.x86_64.rpm unzip-6.0-16.el7.x86_64.rpm
zip-3.0-11.el7.x86_64.rpm bzip2-1.0.6-13.el7.x86_64.rpm
mlocate-0.26-6.el7.x86_64.rpm lshw-B.02.18-7.el7.x86_64.rpm
jansson-2.10-1.el7.x86_64.rpm ypbind-1.37.1-9.el7.x86_64.rpm
yp-tools-2.14-5.el7.x86_64.rpm perl-Net-Telnet-3.03-19.el7.noarch.rpm
tzdata-java-2018d-1.el7.noarch.rpm
perl-XML-Parser-2.41-10.el7.x86_64.rpm
lsof-4.87-4.el7.x86_64.rpm cairo-1.14.8-2.el7.x86_64.rpm
pango-1.40.4-1.el7.x86_64.rpm libjpeg-turbo-1.2.90-5.el7.x86_64.rpm
sos-3.4-13.el7_4.noarch.rpm traceroute-2.0.22-2.el7.x86_64.rpm
openldap-clients-2.4.44-5.el7.x86_64.rpm
```

5 Install the following third-party rpms:

```
# cd /access_build_dir/rhel7_x86_64/ third_party _rpms/
# /bin/rpm -U -v --oldpackage --nodeps --replacefiles --replacepkgs
ctdb-4.6.6-1.el7.x86_64.rpm
perl-Template-Toolkit-2.24-5.el7.x86_64.rpm
perl-Template-Extract-0.41-1.noarch.rpm
perl-AppConfig-1.66-20.el7.noarch.rpm
perl-File-HomeDir-1.00-4.el7.noarch.rpm
samba-common-4.6.11-1.el7.x86_64.rpm
samba-common-libs-4.6.11-1.el7.x86_64.rpm
samba-client-4.6.11-1.el7.x86_64.rpm
samba-client-libs-4.6.11-1.el7.x86_64.rpm
samba-4.6.11-1.el7.x86_64.rpm
samba-winbind-4.6.11-1.el7.x86_64.rpm
samba-winbind-clients-4.6.11-1.el7.x86_64.rpm
samba-winbind-krb5-locator-4.6.11-1.el7.x86_64.rpm
libsmbclient-4.6.6-1.el7.x86_64.rpm
samba-krb5-printing-4.6.11-1.el7.x86_64.rpm
samba-libs-4.6.11-1.el7.x86_64.rpm
libwbclient-4.6.6-1.el7.x86_64.rpm
samba-winbind-modules-4.6.11-1.el7.x86_64.rpm
libnet-1.1.6-7.el7.x86_64.rpm lmdb-libs-0.9.13-2.el7.x86_64.rpm
nfs-ganesha-2.2.0-0.el7.x86_64.rpm
nfs-ganesha-vxfs-2.2.0-0.el7.x86_64.rpm gevent-1.0.2-1.x86_64.rpm
python-msgpack-0.4.6-1.el7ost.x86_64.rpm
python-flask-0.10.1-4.el7.noarch.rpm
python-itsdangerous-0.23-2.el7.noarch.rpm
libevent-libs-2.0.22-1.el7.x86_64.rpm
python-werkzeug-0.9.1-2.el7.noarch.rpm
python-jinja2-2.7.2-2.el7.noarch.rpm sdfs-7.4.0.0-1.x86_64.rpm
psutil-4.3.0-1.x86_64.rpm
python-crontab-2.2.4-1.noarch.rpm libuv-1.9.1-1.el7.x86_64.rpm
```

In this command, you can update the rpm version based on the rpms in the `/access_build_dir/rhel7_x86_64/ third_party _rpms/` directory.

6 Install the Veritas Access rpms.

- Run the following command:

```
# cd /access_build_dir/rhel7_x86_64/rpms/repodata/
# cat access73.repo > /etc/yum.repos.d/access73.repo
```

- Update the *baseurl* and *gpgkey* entry in the `/etc/yum.repos.d/access73.repo` for yum repository directory.
  - `baseurl=file:///access_build_dir/rhel7_x86_64/rpms/`
  - `gpgkey=file:///access_build_dir/rhel7_x86_64/rpms/RPM-GPG-KEY-veritas-access7`
- Run the following commands to refresh the yum repository.
  - `# yum repolist`
  - `# yum grouplist`
- Run the following command.

  `# yum -y groupinstall ACCESS73`
- Run the following command.

  `# /opt/VRTS/install/bin/add_install_scripts`

7 Install the Veritas NetBackup client software.

```
# cd /access_build_dir/rhel7_x86_64
# /opt/VRTSnas/install/image_install/netbackup/install_netbackup.pl
/access_build_dir/rhel7_x86_64/netbackup
```

8 Create soft links for Veritas Access. Run the following command.

```
# /opt/VRTSnas/pysnas/install/install_tasks.py
all_rpms_installed parallel
```

9 License the product.

- Register the permanent VLIC key.

  `# /opt/VRTSvlic/bin/vxlicinstupgrade -k <Key>`
- Verify that the VLIC key is installed properly:

  `# /opt/VRTSvlic/bin/vxlicrep`
- Register the SLIC key file:

  `# /opt/VRTSslic/bin/vxlicinstupgrade -k $keyfile`

- Verify that the SLIC key is installed properly:

  ```
  # /opt/VRTSslic/bin/vxlicrep
  ```

10 Take a backup of the following files:

- `/etc/sysconfig/network`
- `/etc/sysconfig/network-scripts/ifcfg-*`
- `/etc/resolv.conf`

11 Configure the private NIC:

```
# cd /etc/sysconfig/network-scripts/
```

- Configure the first private NIC.
  - Run the following command.

    ```
    # ip link set down priveth0
    ```

  - Update the `ifcfg-priveth0` file with the following:

    ```
    DEVICE=priveth0
    NAME=priveth0
    BOOTPROTO=none
    TYPE=Ethernet
    ONBOOT=yes
    ```

  - Add entries in the `ifcfg-priveth0` file.

    ```
    HWADDR=<MAC address>
    IPADDR= 172.16.0.3  (use IPADDR= 172.16.0.4 for second node)
    NETMASK=<netmask>
    NM_CONTROLLED=no
    ```

    For example:

    ```
    HWADDR=00:0c:29:0c:8d:69
    IPADDR=172.16.0.3
    NETMASK=255.255.248.0
    NM_CONTROLLED=no
    ```

  - Run the following command.

    ```
    # ip link set up priveth0
    ```

- Configure the second private NIC.
  You can configure the second private NIC in the same way. Instead of priveth0, use priveth1 for second node. You do not need to provide IPADDR for priveth1.

12 Configure the public NIC.

```
# cd /etc/sysconfig/network-scripts/
```

- Configure the second public NIC, pubeth1 (in which the host IP is not already configured).
  - Run the following command:

    ```
    # ip link set down pubeth1
    ```

  - Update the `ifcfg-pubeth1` file with the following:

    ```
    DEVICE=pubeth1
    NAME=pubeth1
    TYPE=Ethernet
    BOOTPROTO=none
    ONBOOT=yes
    ```

  - Add entries in the `ifcfg-pubeth1` file.

    ```
    HWADDR=<MAC address>
    IPADDR=<pubeth1_pub_ip>
    NETMASK=<netmask>
    NM_CONTROLLED=no
    ```

  - Run the following command.

    ```
    # ip link set up pubeth1
    ```

- Configure the first public NIC, pubeth0.
  - As the first public NIC will go down, make sure that you access the system directly from its console.
  - Run the following command:

    ```
    # ip link set down pubeth0
    ```

  - Update the `ifcfg-pubeth0` file with the following:

```
DEVICE=pubeth0
NAME=pubeth0
TYPE=Ethernet
BOOTPROTO=none
ONBOOT=yes
```

- Add entries in the `ifcfg-pubeth0` file.

  ```
  HWADDR=<MAC address>
  IPADDR=<pubeth0_pub_ip>
  NETMASK=<netmask>
  NM_CONTROLLED=no
  ```

- Run the following command.

  ```
  # ip link set up pubeth0
  ```

- Verify if pubeth1 is down. If yes, then bring it online.

  ```
  # ip link set up pubeth1
  ```

- Verify the changes.

  ```
  # ip a
  ```

- Run the following command.

  ```
  # service network restart
  ```

  SSH to the above-mentioned IP should work if you start the sshd service.

13 Configure the DNS.

Update the `/etc/resolv.conf` file by adding the following entries:

```
nameserver <DNS>
domain <master node name>
```

For example:

```
nameserver 10.182.128.134
domain clus-01
```

14 Configure the gateway.

Update the `/etc/sysconfig/network` file.

```
GATEWAY=$gateway
NOZEROCONF=yes
```

15 Update the `configfileTemplate` file.

- Enter the following command:

  ```
  # cd /access_build_dir/rhel7_x86_64/manual_install/network
  ```

- Update the `configfileTemplate` file with the current system details:
  - Use *master* as the mode for the master node and *slave* as the mode for the other nodes.
  - This template file is used by the configuration utility script to create configuration files.
  - Provide the same name (current host name) in *old_hostname* and *new_hostname*.

16 Generate the network configuration files.

- The configuration utility script named `configNetworkHelper.pl` creates the required configuration files.

  ```
  # cd /access_build_dir/rhel7_x86_64/manual_install/network
  # chmod +x configNetworkHelper.pl
  ```

- Run the configuration utility script.

  ```
  # ./configNetworkHelper.pl -f configfileTemplate
  ```

- ```
  # cat /opt/VRTSnas/scripts/net/network_options.conf >
  /opt/VRTSnas/conf/network_options.conf
  ```

- ```
  # sed -i -e '$a\' /opt/VRTSnas/conf/net_console_ip.conf
  ```

- Update the `/etc/hosts` file.

  ```
  # echo "172.16.0.3  <master hostname>" >> /etc/hosts
  # echo "172.16.0.4  <slave node name>" >> /etc/hosts
  ```

  For example:

```
# echo "172.16.0.3  clus-01" >> /etc/hosts
# echo "172.16.0.4  clus-02" >> /etc/hosts
```

17 Create the S3 configuration file.

```
# cat /opt/VRTSnas/conf/ssnas.yml
ObjectAccess:
  config: {admin_port: 8144, s3_port: 8143, server_enable: 'no',
  ssl: 'no'}
  defaults:
    fs_blksize: '8192'
    fs_encrypt: 'off'
    fs_nmirrors: '2'
    fs_options: ''
    fs_pdirenable: 'yes'
    fs_protection: disk
    fs_sharing: 'no'
    fs_size: 20G
    fs_type: mirrored
    poollist: []
  filesystems: {}
  groups: {}
  pools: {}
```

18 Set up the Storage Foundation cluster.

- `# cd /access_build_dir/rhel7_x86_64/manual_install/network/SetupClusterScripts`
- `# mkdir -p /opt/VRTSperl/lib/site_perl/UXRT72/CPIR/Module/veritas/`
- `# cp sfcfsha_ctrl.sh /opt/VRTSperl/lib/site_perl/UXRT72/CPIR/Module/veritas/sfcfsha_ctrl.sh`
- `# cp module_script.pl /tmp/`
- `# chmod +x /tmp/module_script.pl`
- Update the cluster name, system name, and NIC name in the following command and execute it:

  ```
  # /tmp/module_script.pl veritas::sfcfsha_config '{"cluster_name" =>
  "<Provide cluster name here>","component" => "sfcfsha","state" =>
  ```

```
"present","vcs_users" => "admin:password:Administrators,user1:
passwd1:Operators","vcs_clusterid" => 14865,"cluster_uuid" =>
"1391a-443ab-2b34c","method" => "ethernet","systems" =>
"<Provide hostnames separated by comma>","private_link" =>
"<provide private nic name separated by comma>"}'
```

For example, if the cluster name is *clus* and the host names are *clus-01* and *clus-02*.

```
/tmp/module_script.pl veritas::sfcfsha_config '
{"cluster_name" => "clus","component" => "sfcfsha",
"state" => "present","vcs_users" =>
"admin:password:Administrators,user1:passwd1:Operators",
"vcs_clusterid" => 14865,"cluster_uuid" => "1391a-443ab-2b34c",
"method" => "ethernet","systems" => "clus-01,clus-02",
"private_link" => "priveth0,priveth1"}'
```

- Update and configure the following files:
  - `# rpm -q --queryformat '%{VERSION}|%{BUILDTIME:date}|%{INSTALLTIME:date}|%{VERSION}\n' VRTSnas > /opt/VRTSnas/conf/version.conf`
  - `# echo NORMAL > /opt/VRTSnas/conf/cluster_type`
  - `# echo 'path /opt/VRTSsnas/core/kernel/' >> /etc/kdump.conf`
  - `# sed -i '/^core_collector\b/d;' /etc/kdump.conf`
  - `# echo 'core_collector makedumpfile -c --message-level 1 -d 31' >> /etc/kdump.conf`

19 Start the Veritas Access product processes.

- Provide the current host name in the following command and execute it.

```
# /tmp/module_script.pl veritas::process '{"state" => "present",
"seednode" => "<provide current hostname here>","component"
 => "sfcfsha"}'
```

For example, if the host name is *clus-01*:

```
# /tmp/module_script.pl veritas::process '{"state" =>
"present","seednode" => "clus-01","component" => "sfcfsha"}'
```

If you are running it on *clus-02*, then you have to provide “seednode" => "clus-02”.

- Run the following command.

```
# /opt/VRTSnas/pysnas/install/install_tasks.py
all_services_running serial
```

20 Create the CVM group.

If the `/etc/vx/reconfig.d/state.d/install-db` file exists, then execute the following command.

```
# mv /etc/vx/reconfig.d/state.d/install-db
/etc/vx/reconfig.d/state.d/install-db.a
```

If CVM is not configured already then run the following command on the master node.

```
# /opt/VRTS/bin/cfscluster config -t 200 -s
```

21 Enable hacli.

Verify in `/etc/VRTSvcs/conf/config/main.cf` file. If `HacliUserLevel = COMMANDROOT` exists, then move to step 22 , else follow below steps to enable hacli in your system.

```
# /opt/VRTS/bin/hastop -local
```

Update the `/etc/VRTSvcs/conf/config/main.cf` file.

If it does not exist, then add the following line:

```
HacliUserLevel = COMMANDROOT  in cluster <cluster name> ( ) loop
```

For example:

```
cluster clus (
   UserNames = { admin = aHIaHChEIdIIgQIcHF, user1 = aHIaHChEIdIIgFEb }
   Administrators = { admin }
   Operators = { user1 }
   HacliUserLevel = COMMANDROOT
# /opt/VRTS/bin/hastart
```

Verify that hacli is working.

```
# /opt/VRTS/bin/hacli -cmd "ls /" -sys clus-01
```

22 Verify that the HAD daemon is running.

```
# /opt/VRTS/bin/hastatus -sum
```

23 Configure Veritas Access on the second node by following steps 1 to 22 .

24 Verify that the system is configured correctly.

- Verify that LLT is configured correctly.

  ```
  # lltconfig -a list
  ```

  For example:

  ```
  [root@clus-02 SetupClusterScripts]# lltconfig -a list
  Link 0 (priveth0):
    Node    0 clus-01    :    00:0C:29:0C:8D:69
    Node    1 clus-02    :    00:0C:29:F0:CC:B6  permanent

  Link 1 (priveth1):
   Node    0 clus-01     :    00:0C:29:0C:8D:5F
   Node    1 clus-02     :    00:0C:29:F0:CC:AC  permanent
  ```

- Verify that GAB is configured properly.

  ```
  # gabconfig -a
  ```

  For example:

  ```
  [root@clus-01 network]# gabconfig -a
  GAB          Port   Memberships
  ================================
  Port a gen   43b804 membership 01
  Port b gen   43b807 membership 01
  Port h gen   43b821 membership 01
  ```

- Verify the LLT state.

  ```
  # lltstat -nvv
  ```

  For example:

  ```
  [root@clus-01 network]# lltstat -nvv
  LLT node information:
        Node         State    Link    Status  Address
     * 0 clus-01     OPEN
                              priveth0   UP    00:0C:29:0C:8D:69
                              priveth1   UP    00:0C:29:0C:8D:5F
       1 clus-02     OPEN
                              priveth0   UP      00:0C:29:F0:CC:B6
                              priveth1   UP      00:0C:29:F0:CC:AC
       2          CONNWAIT
                             priveth0   DOWN
                             priveth1   DOWN
  ```

- The `vxconfigd` daemon should be online on both nodes.

  ```
  # ps -ef | grep vxconfigd
  ```

  For example:

  ```
  # ps -ef | grep vxconfigd
  root   13393 1  0 01:33 ?  00:00:00 vxconfigd -k -m disable -x syslog
  ```

25 Run the Veritas Access post-start actions.

- Make sure that HAD is running on all the nodes.

  ```
  # /opt/VRTS/bin/hastatus
  ```

- On all the nodes, create a `communication.conf` file to enable hacli instead of ssh.

```
vim /opt/VRTSnas/conf/communication.conf
{
 "WorkingVersion": "1",
 "Version": "1",
 "CommunicationType": "HACLI"
}
```

- Run the installer to install Veritas Access. Run the following command only on the master node.

```
# /opt/VRTSnas/install/image_install/installer -m master
```

26 Run the join operation on the slave node.

```
# /opt/VRTSnas/install/image_install/installer -m join
```

27 Run the following command on both the nodes.

```
# echo "<first private nic name>" >
/opt/VRTSnas/conf/net_priv_dev.conf
```

For example:

```
# echo "priveth0" > /opt/VRTSnas/conf/net_priv_dev.conf
```

28 Enable NFS resources. Run the following commands on the master node.

```
# /opt/VRTS/bin/haconf -makerw
# /opt/VRTS/bin/hares -modify ssnas_nfs Enabled 1
# /opt/VRTS/bin/haconf -dump -makero
```

You can now use the two-node Veritas Access cluster.

Enabling internal sudo user communication in Veritas Access

By default, Veritas Access uses SSH communication between the nodes for the root user. If you want to use sudo user-based communication, you can set the internal communication to use the sudo user communication after you have installed Veritas Access successfully.

You can use the following steps to set up the sudo user communication.

- Phase 1: Create an access_user on each of the nodes of the Veritas Access cluster.
- Phase 2: Set up a passwordless communication between the root and access_user on each node
- Phase 3: Select the communication type as SUDO_SSH

Phase 1: Create an access_user on each of the nodes of the Veritas Access cluster

1 Create the `access_user` and set the password.

For example:

```
[root@access1-01 ~]# useradd access_user
[root@access1-01 ~]# passwd access_user
Changing password for user access_user.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

2 Add the `access_user` to the `sudoers` file.

For example:

```
[root@access1-01 ~]# echo "access_user ALL=(ALL) NOPASSWD: ALL"
>> /etc/sudoers
```

Complete Phase 1 on all the nodes of the cluster.

Phase 2: Set up a passwordless communication between the root and access_user on each node

1 Generate `rsa` key for the `root` user if it is not present.

For example:

```
[root@access1-01 ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:hRIBljcpsmGMCtfUUjyVGOfe957OXyiXcRyiYBprmZk root@access1-01
The key's randomart image is:
+---[RSA 2048]----+
| o o+*=*o.       |
|o *.= O+..       |
|oo + +.+oo.  . . |
|. .    oXo. . ...|
|       ES ... . o|
|      .    . . = |
|             . = .|
|              = ..|
|              .=..|
+----[SHA256]-----+
```

2 Copy the `rsakey.pub` of the `root` user to the `access_user` for each of the nodes in the cluster.

For example:

```
[root@access1-01 ~]# ssh-copy-id access_user@access1-01
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed:
"/root/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new
key(s),to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed --
if you are prompted now it is to install the new keys
access_user@access1-01's password:

Number of key(s) added: 1
```

Phase 3: Select the communication type as SUDO_SSH

- Create the `/opt/VRTSnas/conf/communication.conf` file.

```
[root@access1-01 ~]# cat /opt/VRTSnas/conf/communication.conf
{
    "WorkingVersion": "1",
    "Version": "1",
    "CommunicationType": "SUDO_SSH"
}
```

Index

L

M

N

O

P

R

S

U

V