Veritas Access 7.3 on Amazon Web Services Cloud

Deployment Guide - Linux

7.3

September 2017

Veritas Access on Amazon Web Services Cloud

Last updated: 2017-09-11

Document version: 7.3 Rev 0

Legal Notice

THE DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. VERITAS TECHNOLOGIES LLC SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

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

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 document version appears on page 2 of each guide. The latest documentation is available on the Veritas website:

https://sort.veritas.com/documents

Documentation feedback

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

doc.feedback@veritas.com

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

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

Veritas Services and Operations Readiness Tools (SORT)

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

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

Contents

Chapter 1

Veritas Access in Amazon Web Services Cloud

This chapter includes the following topics:

Overview

Companies around the world are realizing that moving applications and data to the cloud can reduce expenses and streamline IT operations. Enterprises are looking to deploy hybrid cloud architectures for their mission-critical OLTP and OLAP applications for increased flexibility and savings. Enterprises also want to ensure seamless data movement, security, enterprise-class application performance, scalability, as well as application and workload resilience.

Veritas Access is a software-only, scale-out network-attached storage (NAS) solution for unstructured data that can be installed on any x86 commodity server and can provide access to storage using multiple network protocols for on-premises and cloud ecosystems such as Amazon Web Services (AWS). Veritas Access enables high performance and cost-optimized software-defined storage for unstructured data workloads on AWS by offering required workload performance and intelligent AWS cloud storage tiering.

For on-premises workloads, Veritas Access can provision AWS S3 storage as a low-cost storage tier for unstructured data workloads by optimizing the migration of that data to AWS, driven by automated policies. With Veritas Access, organizations of any size can now take full advantage of the AWS cloud to experience significant cost savings while enabling their IT teams to be more productive.

With the enterprise grade capabilities of Veritas Access technology, you can now:

- Export NFS, CIFS shares using Veritas Access.
  Veritas Access makes use of Flexible Shared storage (FSS) to create a shared pool of the local Elastic Block Storage(EBS) storage from the hosts to create shared volumes or file systems on shared nothing architectures that redefine the storage, performance and scalability potential of your cloud infrastructure. FSS leverages Amazon EBS volumes to create shared storage in the cloud.
- Make use of S3 as an underlying file system tier. Using this, the data is tiered between EBS and S3 and the data is then exported as an NFS or CIFS share.
- Leverage the Veritas Access replication capabilities to replicate the Veritas Access cluster to different AWS regions or replicate from on-premises to cloud.

About this document

This document provides instructions for a fully functional automated deployment of Veritas Access on Amazon Web Services (AWS) cloud by using AWS CloudFormation templates.

You can deploy fully functional EC2 instances running Veritas Access on AWS cloud by using the CloudFormation templates. The templates setup and configure the AWS environment for Veritas Access with essential resources—Amazon Elastic Compute Cloud(EC2) instances, Amazon Virtual Private Cloud(VPC), Elastic Network Interfaces (ENICs), and Elastic Block Storage (EBS).

The intended audience for this guide includes storage administrators, architects, and system administrators who are planning to deploy the Veritas Access solution on AWS cloud.

Architecture

The Veritas Access cluster is deployed using the CloudFormation service. Veritas Access deployment through CloudFormation templates supports a maximum of two EC2 instances in an Availability Zone.

The Veritas Access cluster deployment creates:

1. A VPC is created in a single Amazon Zone.
2. One public subnet and three private subnets are created with the VPC.
3. Two Veritas Access nodes launched from EBS backed AMIs are deployed with:

   Two ENICs per instance in one private subnet for accessing the shares.

   Two more ENICs per instance distributed in the remaining private subnets for formation of the cluster.
4. The Veritas Access nodes form a cluster using LLT links that are configured (as LLT over UDP) within the two private subnets.
5. A Bastion host is deployed in a public subnet with an attached Elastic IP which acts as a gateway host.
6. The Veritas Access nodes require web access for accessing the AWS resource, hence a NAT gateway (with an Elastic IP assigned) is created in the public subnet.

Figure 1-1 illustrates deployment of Veritas Access in AWS environments.

Figure 1-1 Veritas Access architecture - Deployment time

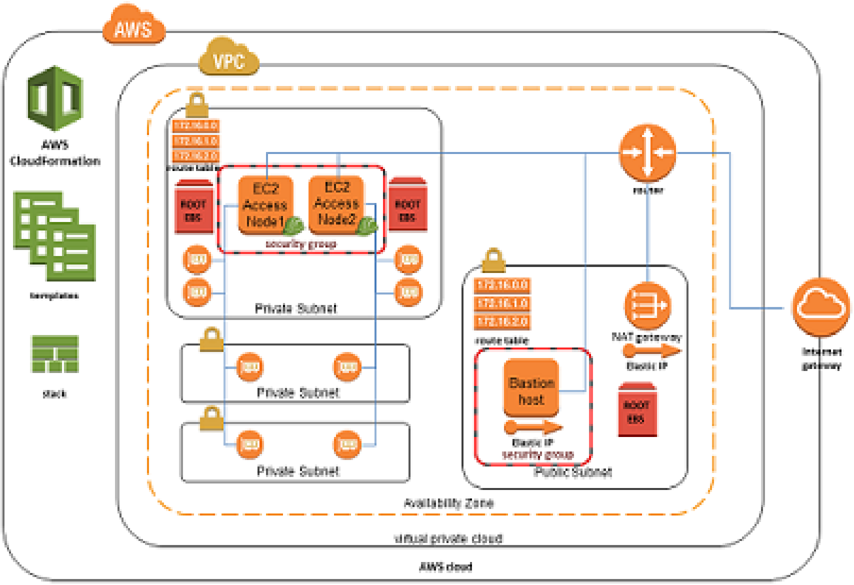

Figure 1-2 Veritas Access architecture - Run time

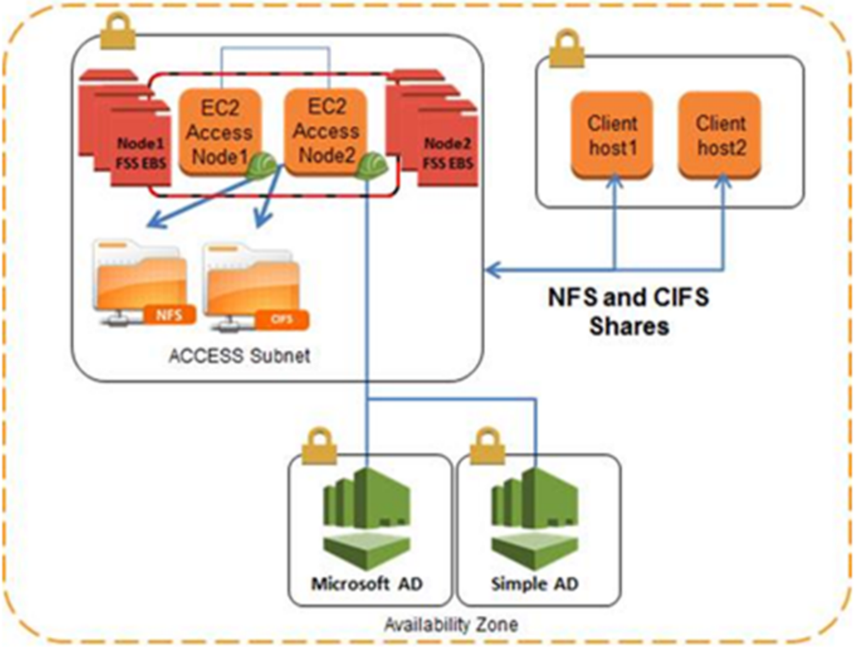

Figure 1-2 illustrates Veritas Access in AWS environments during run time.

Pre-requisites for replication

Ensure that you do the following settings before you use replication:

- Set up the VPN between the regions or zones where you want replication to be set up
- The Maximum Transmission Unit (MTU) of pubeth0 (NIC for Veritas Volume Replication (VVR)) for all nodes (including VPN systems and the system on which Access is installed) in the setup must be set to 1500.
- The following rules need to be created in the AWS console.

  ```
  Port:4145     UDP      Remote cluster subnet as source
  Port:4145     TCP      Remote cluster subnet as source
  Port:8199     UDP      Remote cluster subnet as source
  Port:8199     TCP      Remote cluster subnet as source
  Port:8989     UDP      Remote cluster subnet as source
  Port:8989     UDP      Remote cluster subnet as source
  ```

- The security groups need to be updated with remote cluster CIDR.
  Rules need to be updated as follows:

  ```
  OLD entry: Custom TCP Rule    TCP        56987     <LOCAL CIDR>
  New entry: Custom TCP Rule    TCP        56987     <REMOTE CIDR>
  ```

  Add the following:

  ```
  SSH                  TCP         22       <REMOTE CIDR>
  All ICMP - IPV4      ALL         N/A      <REMOTE CIDR>
  ```

For example:

```
Update security group:
On machine with CIDR 172.16.0.0/16
 OLD:     Custom TCP Rule     TCP     56987    172.16.0.0/16
 New:     Custom TCP Rule     TCP     56987    12.16.0.0/16
Here 172.16.0.0/16 is local cluster’s CIDR and
12.16.0.0/16 is remote cluster’s CIDR

Add the following
 SSH                TCP     22      12.16.0.0./16
 ALL ICMP - IP4     TCP     N/A     12.16.0.0./16
```

```
On machine with CIDR 12.16.0.0/16
 OLD:     Custom  TCP Rule     TCP     56987    12.16.0.0/16
 New:     Custom  TCP Rule     TCP     56987    172.16.0.0/16
Here 12.16.0.0/16 is local cluster’s CIDR and
172.16.0.0 /16 is remote cluster’s CIDR)
 OLD:     Custom  TCP Rule     TCP     56987    12.16.0.0/16
 New:     Custom  TCP Rule     TCP     56987    172.16.0.0/16

Add the following
 SSH              TCP     22      172.16.0.0/16
 ALL ICMP - IP4   TCP     N/A     172.16.0.0/16
```

Replication on AWS for a multizone deployment

Multizone deployment of Veritas Access on AWS works in the same manner as a single zone deployment. During a multizone deployment of Veritas Access on AWS, the virtual IPs, console IPs and replication IPs are not added to the ENI directly. The IPs are plumbed on NICs at host level and routed at AWS level. Hence, you need to take care of the routing when setting up the VPN.

The steps below describe how the routing changes are done keeping an OpenSwan VPN in perspective.

- Make configuration file changes on the VPN host.
  The files `/etc/ipsec.d/vpc2-to-vpc1.conf` and `/etc/ipsec.d/vpc1-to-vpc2.conf` are required to make use of the **leftsubnets** and **rightsubnets** tags instead of **leftsubnet** and **rightsubnet**.
  The entries are as follows:

  ```
  leftsubnets={CIDR1,CIDR2,....,}
  rightsubnets={CIDR3,CIDR4,...,}
  ```

  You can have the following deployments for Veritas Access:
  Deployment1:

  ```
  CIDR: 10.16.0.0/16
  ConsoleIP: 10.17.1.0/32
  Replication IP: 10.18.1.3/32
  ```

  Deployment2:

  ```
  CIDR: 12.16.0.0/16
  ConsoleIP: 12.17.1.0/32
  Replication IP: 12.18.1.3/32
  ```

Then

```
On Deployment1 (/etc/ipsec.d/vpc1-to-vpc2.conf)
    leftsubnets={10.16.0.0/16, 10.17.1.0/32,10.18.1.3/32}
    rightsubnets={12.16.0.0/16, 12.17.1.0/32,12.18.1.3/32}
On Deployment2 (/etc/ipsec.d/vpc2-to-vpc1.conf)s
    leftsubnets={12.16.0.0/16, 12.17.1.0/32,12.18.1.3/32}
    rightsubnets={10.16.0.0/16, 10.17.1.0/32,10.18.1.3/32}
```

- Make routing changes at the AWS level for virtual IP, console IPs, and replication IPs.

  Since the VPN host understands the IPs on the Veritas Access cluster at the remote site, the current Veritas Access nodes have to route their requests to the remote IP using the VPN host. Hence, the AWS routing tables have to be updated to route traffic allocated to the remote Veritas Access cluster's IPs to the VPN host.

  For example:

  If we use the same subnets and VIPs as before and if *rtb-c88306ae*, *rtb-97be44ef*, and *rtb-8abe44f2* are the route tables used by the Veritas Access clusters subnets and if the VPN host's NIC is mapped to *eni-2aa3bbf2*, then the following routes need to be added

```
aws ec2 create-route --route-table-id rtb-c88306ae --destination-cidr-block
12.17.1.0/24 --network-interface-id eni-2aa3bbf2
aws ec2 create-route --route-table-id rtb-97be44ef --destination-cidr-block
12.17.1.0/24 --network-interface-id eni-2aa3bbf2
aws ec2 create-route --route-table-id rtb-8abe44f2 --destination-cidr-block
12.17.1.0/24 --network-interface-id eni-2aa3bbf2
```

You have to make similar entries for the other virtual IPs.

The same steps have to be followed for the remote cluster.

Licensing Veritas Access

You have to obtain a license to install and use Veritas Access. Trialware license is applied to the Veritas Access cluster by default, which is valid for 60 days upon the set up of the cluster.

To apply a permanent license:

1 Log on to any of the Veritas Access cluster nodes using the support account.

2 Copy the permanent license file to the node.

3 Enter the following command to invoke the licensing menu.

```
# /opt/VRTS/install/access72 -license
```

Deployment steps

The procedure for an end-to-end deployment of Veritas Access on AWS consists of the following steps:

1. Step 1: Prepare your AWS account.

   See “Step 1: Prepare your AWS account” on page 12.

2. Step 2: Subscribe to the Veritas Access Amazon Machine Image (AMI).

   See “Step 2: Subscribe to the Access AMI” on page 13.

3. Step 3: Launch the stack.

   See “Step 3: Launch the stack” on page 14.

4. Step 4: Access the Veritas Access nodes

   .See “Step 4: Access Veritas Access nodes” on page 15.

Step 1: Prepare your AWS account

Preparing your AWS account involves the following:

- Choosing a region
- Creating a key pair
- Requesting increases for account limits (if necessary)

To prepare your AWS account

1. If you do not have an existing AWS account, create one at http://aws.amazon.com by following the on-screen instructions. Part of the sign-up process involves receiving a phone call and entering a PIN using the phone keypad.

2. Use the region selector in the navigation bar to choose the Amazon EC2 region where you want to deploy Veritas Access on AWS.

   Amazon EC2 locations are composed of Regions and Availability Zones. Regions are dispersed and located in separate geographic areas.

   Note: Consider choosing a region closest to your data center or corporate network to reduce network latency between systems running on AWS and the systems and users on your corporate network.

3. Create a key pair in your preferred region. To do this, in the navigation pane of the Amazon EC2 console, choose **Key Pairs**, **Create Key Pair**, type a name, and then choose **Create**.

   Amazon EC2 uses public-key cryptography to encrypt and decrypt log in information. To be able to log on to your instances, you must create a key pair. On Linux, Veritas Access uses the key pair to authenticate SSH log in.

4. (Production deployments only): If necessary, request a service limit increase for the instance type you are using. If you already have an existing deployment that uses this instance type, and you think you might exceed the default limit with this reference deployment, you need to request an increase. To do this, in the AWS Support Center, choose **Create Case**, **Service Limit Increase**, **EC2 instances**, and then complete the fields in the limit increase form. It might take a few days for the new service limit to become effective.

   For more information, see the *Amazon EC2 Service Limits* in the AWS documentation.

Step 2: Subscribe to the Access AMI

Before you launch the Quick Start, you need to subscribe to the Access AMI in the AWS Marketplace.

To subscribe to the Access AMI

1 Log on to the AWS Marketplace at http://aws.amazon.com/marketplace.

2 Subscribe to **Access**.

3 Choose **Continue**, and then use the **1-Click Launch** option to launch the AMI into your account on Amazon EC2.

This involves accepting the terms of the license agreement and receiving confirmation email.

For detailed instructions, see the AWS Marketplace documentation.

Step 3: Launch the stack

This section contains general instructions for deploying Veritas Access on to a new VPC.

To launch the stack

1 Open the Cloud Formation Console page.

2 Choose the option **Create stack** to launch the AWS CloudFormation template into your AWS account.

Note: You are responsible for the cost of the AWS services used while running this Quick Start reference deployment. There is no additional cost for using this Quick Start. Prices are subject to change. For full details, see the pricing pages for each AWS service you want to use in this Quick Start or the AWS Simple Monthly Calculator.

3 Check the region that is displayed in the upper-right corner of the navigation bar, and change it, if necessary. This is where the network infrastructure for Veritas Access is built.

4 On the **Select Template** page, keep the default URL for the AWS CloudFormation template, and then choose **Next**.

5 On the **Specify Details** page, change the stack name if needed. Review the parameters for the template. Provide values for the parameters that require input. For all other parameters, review the default settings and customize them as necessary. When you finish reviewing and customizing the parameters, choose **Next**.

   The parameters are grouped by category.

   See "Network configuration parameters" on page 16.

   See "Server and storage configuration parameters" on page 17.

6 On the **Options** page, you can specify tags (key-value pairs) for resources in your stack and set additional options. When you are done, choose **Next**.

7 On the **Review** page, review and confirm the template settings. Under **Capabilities**, select the check box to acknowledge that the template creates IAM resources.

8 Choose **Create** to deploy the stack.

9 Monitor the status of the stack. When the status is **CREATE_COMPLETE**, the Veritas Access cluster is ready.

Once the deployment is complete, the information for the Veritas Access cluster is displayed in tags as below:

- The virtual IPs (VIPs) created as part of the Veritas Access deployment are displayed in the tags "`AccessVirtualIPs`".
- The console IP for the Veritas Access cluster is displayed under the tag "`AccessConsoleIP`".
- The Veritas Access GUI url is displayed under the tag "`Access GUI URL`".

Step 4: Access Veritas Access nodes

Veritas Access instances are placed in a private subnet and are not directly accessible from the Internet. You can access the Veritas Access instances only through instances placed either in the public subnet or the DMZ layer.

You can access the nodes through the DMZ layer by using an OS-level access. You can SSH to the bastion host and then access the Veritas Access instance(s) by using an SSH client of your choice.

Using OS-level access

1 On the Amazon EC2 console, choose **Running Instances**.

2 Select your bastion host, and note the public Elastic IP address displayed below your running instances.

3 You can use an SSH client of your choice (for example, PuTTY) to connect to the bastion host. You have to use the key pair you specified earlier during the deployment process.

Note: If your connection times out, you may need to adjust the security group rules for the bastion host to allow access from your system's IP address or proxy server. For more information, see Security Group Rules in the *Amazon EC2 User Guide*.

4 The Veritas Access nodes can be accessed by either using the master account or the support account.

The private keys for these are saved in the home directory of the Veritas Access node, ec2-user account as `/home/ec2-user/`*`hostname-pem`*. You can find the master and support .pem keys for accessing the respective accounts in this directory.

The keys can be copied back either to the bastion host or any other host from where you intend to log on to the Veritas Access hosts. You can also use the keys to log on to the Veritas Access instance.

```
ssh -i master-<host>.pem master@<host>
ssh -i support-<host>.pem support@<host>
```

Parameters for deploying Veritas Access on a new VPC

Provide the following information for deploying Veritas Access on a new VPC.

Network configuration parameters	See “Network configuration parameters” on page 16.
Server and storage configuration parameters	See “Server and storage configuration parameters” on page 17.

Network configuration parameters

Table 1-1 describes the network configuration parameters for deploying Veritas Access into an existing VPC.

Table 1-1 Network configuration parameters for deploying Veritas Access into an existing VPC

Parameter label	Default value	Description
VPCCIDR	172.16.0.0/16	CIDR block for the subnet from which the Veritas Access cluster is accessible.
PublicNicsSubCIDR	172.16.3.0/24	CIDR block for the private interconnect subnet.
Priv1SubCIDR	172.16.0.0/24	CIDR block for the private interconnect subnet.
Priv2SubCIDR	172.16.2.0/24	CIDR block for the subnet within which the bastion host is present.
DMZCIDR	172.16.7.0/24	CIDR block for the public DMZ subnet located in the new VPC.
RemoteAccessCIDR	0.0.0.0/0	CIDR block from where you want to access your bastion instances.
AvailabilityZone	Requires user input	The Availability Zone where Veritas Access subnets are created.

Server and storage configuration parameters

Table 1-2 describes the server and storage configuration parameters for deploying Veritas Access into an existing VPC.

Table 1-2 Server and storage configuration parameters for deploying Veritas Access into an existing VPC

Parameter label	Default value	Description
MyInstanceType	m4.xlarge	Instance type for the Veritas Access node.
MyInstance1IP	172.16.3.10	Private IP address for the first node of the Veritas Access cluster

Table 1-2 Server and storage configuration parameters for deploying Veritas Access into an existing VPC *(continued)*

Parameter label	Default value	Description
MyInstance2IP	172.16.3.11	Private IP address for the second node of the Veritas Access cluster
AccessRootPassword	Requires user input	Veritas Access password. It must contain at least 8 characters, including uppercase, lowercase, and numeric values.
BastionInstanceType	t2.micro	Amazon EC2 instance type for the bastion host.
KeyName	home	Name of an existing Amazon EC2 key pair. All instances launch with this key pair.

Troubleshooting

If you run into any issues while deploying the Veritas Access Quick Start, contact Veritas Technical Support at https://www.veritas.com/support.

Resources

AWS Services

- AWS CloudFormation documentation
- Amazon EBS
  User Guide
  Volume types
  Optimized instances
- Amazon EC2 user guide for Linux
- Amazon VPC

Veritas Access documents

Documentation

Feedback

We welcome your questions and comments. Please post your feedback on the AWS Quick Start Discussion Forum.

Configuration reference

This appendix includes the following topics:

About configuring the Veritas Access network

Veritas Access has the following types of networks:

- Private network
  The network between the nodes of the cluster itself. The private network is not accessible to the Veritas Access client nodes.
- Public network
  The public network is visible to all the clients. Veritas Access uses static IP addresses for its public networking interface. Veritas Access does not support DHCP for public network configuration.

You can use the CLI `Network> show` command to display the current cluster configuration and related statistics for the cluster network configuration.

Configuring DNS settings

The Domain Name System (DNS) service resolves names to IP addresses. The DNS commands let you view or change a Veritas Access cluster's DNS settings. You can configure Veritas Access to use DNS to look up domain names and IP addresses. You enable the DNS service for the cluster, then specify up to three DNS servers.

Table A-1 DNS commands

Command	Definition
`dns show`	Displays the current settings of the Veritas Access cluster's DNS lookup service.
`dns enable`	Enables Veritas Access to perform DNS lookups. When DNS is enabled, the Veritas Access cluster's DNS service uses the data center's DNS server(s) to determine the IP addresses of network entities such as SNMP, NTP, LDAP, and NIS servers with which the cluster must communicate.
`dns disable`	Disables DNS lookups.
`dns set domainname`	Enter the domain name that the Veritas Access cluster will be in. For the required information, contact your Network Administrator. This command clears any previously set domain name. Before you use this procedure, you must enable the DNS server.
`dns set nameservers`	Specifies the IP addresses of DNS name servers to be used by the Veritas Access DNS lookup service. The order of the IP addresses is the order in which the name servers are used. Enter the IP addresses of the name servers. The order of the IP addresses is the order in which the name servers are used.

Table A-1 DNS commands *(continued)*

Command	Definition
`dns set searchdomains`	Allows you to have multiple DNS search domains. Specify the search domains in the order in which the search domains should be used.
`dns clear domainname`	Removes the DNS domain name.
`dns clear nameservers`	Removes the IP addresses of DNS name servers from the cluster's DNS lookup service database.

Network mode - ip commands

Table A-2 lists the commands you can use to configure your IP addresses.

Table A-2 Network> IP commands

Command	Definition
`ip addr add`	Adds a virtual or a physical IP address to the Veritas Access cluster. You can specify either an IPv4 address or an IPv6 address. Veritas Access assigns the newly added IP address to an Ethernet interface or one of its nodes. Virtual IP addresses are used for communication among cluster nodes and with clients on the enterprise network. Veritas Access determines the node to which the IP address is assigned.
`ip addr del`	Deletes an IP protocol address from the cluster. You can delete physical IP addresses only if they are not being used by any interface of the cluster. You can also delete virtual IP addresses, except for the console IP address. When you add or delete an IP address from the cluster, the cluster automatically evens out the number of virtual IP addresses on each node. You can specify either an IPv4 address or an IPv6 address.
`ip addr modify`	Modifies an IP protocol address used by the cluster. You can change both the physical IP addresses and virtual IP addresses. If you change the virtual IP address it terminates the NFS connection on the old IP address. You can specify either an IPv4 address or an IPv6 address.

Table A-2 Network> IP commands *(continued)*

Command	Definition
`ip addr online`	Brings an IP address online on any running node in the cluster. The IP address does not need to be in the offline mode for this command to work. You can use this command to switch the IP address from an online node to another specified node. You can change an IP address to the online mode if it is in the `OFFLINE/FAULTED` state. This command also displays any faults for the IP address on the specified node. You can specify either an IPv4 address or an IPv6 address.
`ip addr show`	Displays the IP addresses, the devices (Ethernet interfaces) they are assigned to, and their attributes. **Note:** Any Ethernet interfaces excluded during the initial Veritas Access installation are not displayed.
`ip link set`	Changes the network Ethernet interface's attributes or states.
`ip link show`	Displays each Ethernet interface's (device) status, if it connected to each node in the cluster, the speed, MTU, and MAC address. **Note:** Any Ethernet interfaces excluded during the initial Veritas Access installation are not displayed.
`ip route add`	Adds a new route for the cluster. The routing table contains information about paths to other networked nodes. You can make routing table changes on each node of the cluster. Use `all` for the *nodename* to add the route to all of the nodes in the cluster. Use a netmask value of 255.255.255.255 for the *netmask* to add a host route to *ipaddr*. Use a value of 0.0.0.0 for the *gateway* to add a route that does not use any gateway. The `dev` *device* is an optional argument. Use any of the public Ethernet interfaces for the *device*, for example, pubeth0, pubeth1, or any.
`ip route del`	Deletes a route used by the cluster. Use `all` for *nodename* to delete the route from all of the nodes in the cluster. The combination of *ipaddr* and *netmask* specifies the network or host for which the route is deleted. Use a value of 255.255.255.255 for the *netmask* to delete a host route to *ipaddr*.

Table A-2 Network> IP commands *(continued)*

Command	Definition
`ip route show`	Displays the routing table of the nodes in the cluster. You can enter a specific *nodename* or use `all` to display the routing tables for all the nodes in the cluster.

System mode commands

The `System> mode` includes commands to set the system clock, configure the NTP server, and other system-wide configuration options.

The `System> ntp` commands describe coordinating cluster nodes to work with NTP servers. You can set the Network Time Protocol (NTP) server on all of the nodes in the cluster.

Table A-3 System> ntp commands

Command	Definition
`ntp disable`	Disables the NTP server on all of the nodes in the cluster.
`ntp enable`	Enables the NTP server on all of the nodes in the cluster.
`ntp servername`	Sets the NTP server on all of the nodes in the cluster.
`ntp show`	Displays the NTP status and server names.
`ntp sync`	Synchronizes the date on the NTP server across all of the nodes in the cluster.

The `System> clock` commands set or show the date and time of the system, including setting time zones and displaying the list of regions.

Table A-4 System> clock commands

Command	Definition
`clock regions`	Displays the clock regions and their cities.
`clock set`	Sets the system date and time.
`clock show`	Displays the current system date and time.
`clock timezone`	Sets the time zone for the system. **Note:** This command only accepts the name of a city.

Storage commands

The `Storage> disk` commands display the aggregated information of the disk devices connected to all of the nodes in the cluster.

Table A-5 Storage> disk commands

Command	Definition
`disk list stats` (default)	Displays a list of disks and nodes in tabular form. Each row corresponds to a disk, and each column corresponds to a node. ■ If an `OK` appears in the table, it indicates that the disk that corresponds to that row is accessible by the node that corresponds to that column. ■ If an `ERR` appears in the table, it indicates that the disk that corresponds to that row is inaccessible by the node that corresponds to that column. This list does not include the internal disks of each node.
`disk list detail`	Displays the disk information, including a list of disks and their properties. If the console server is unable to access any disk, but if any other node in the cluster is able to access that disk, then that disk is shown as "---."
`disk list paths`	Displays the list of multiple paths of disks connected to all of the nodes in the cluster. It also shows the status of each path on each node in the cluster.
`disk list types`	Displays the enclosure name, array name, and array type for a particular disk that is present on all of the nodes in the cluster.

The `Storage> pool` commands allow you to configure storage for Veritas Access.

Table A-6 Storage> pool commands

Command	Definition
`pool adddisk`	You can add a new disk to an existing pool. A disk can belong to only one pool. The minimum size of disks required for creating a pool or adding a disk to the pool is 10 MB. **Note:** Disks being used for the `pool adddisk` command must support SCSI-3 PGR registrations if I/O fencing is enabled.

Table A-6 Storage> pool commands *(continued)*

Command	Definition
`pool create`	Creates storage pools. You can build file systems on top of them. **Note:** Disks being used for the `Storage> pool create` command must support SCSI-3 PGR registrations if I/O fencing is enabled. **Note:** The minimum size of disks required for creating a pool or adding a disk to the pool is 10 MB.
`pool destroy`	Destroys storage pools used to create file systems. Destroying a pool does not delete the data on the disks that make up the storage pool.
`pool free`	Lists the free space in each of the pools. Free space information includes: ■ Disk name ■ Free space ■ Total space ■ Use %
`pool list`	Displays the pools and associated disks. A storage pool is a collection of disks from shared storage; the pool is used as the source for adding file system capacity as needed.
`pool mvdisk`	You can move disks from one storage pool to another.
`pool rename`	Renames a pool.
`pool rmdisk`	You can remove a disk from a pool. **Note:** You cannot remove a disk from a pool if the disk has data on it. If a specified disk does not exist, an error message is displayed. If one of the disks does not exist, then none of the disks are removed. A pool cannot exist if there are no disks assigned to it. If a disk specified to be removed is the only disk for that pool, the pool is removed as well as the assigned disk. If the specified disk to be removed is being used by a file system, then that disk will not be removed.

NFS mode commands

The `NFS>` mode includes the commands to configure Veritas Access with an NFS server.

Table A-7 NFS mode commands

Command	Definition
`server start`	Starts the NFS server.
`server status`	Displays the status of the NFS server.
`server stop`	Stops the NFS server.
`server switch`	Switches the NFS server.
`stat show`	Displays the NFS statistics..
`stat show all`	Displays the NFS statistics for all the nodes in the cluster for the kernel NFS server.
`stat reset`	Resets the NFS statistics for the kernel NFS server.

CIFS mode commands

The `CIFS>` mode includes commands to configure the Veritas Access CIFS service.

Each command listed is a keyword that represents a group of related commands

Table A-8 CIFS mode commands

Command	Definition
`homedir`	Manages the home directories of CIFS users.
`local`	Creates, deletes, or displays information about local groups and local users.
`server`	Starts, stops, or shows the status of the CIFS server.
`set`	Configures settings for the CIFS server.
`share`	Manages CIFS shares.
`show`	Displays the settings for the CIFS server.

CIFS mode - homedir commands

The `CIFS> homedir` commands manage the home directories of CIFS users.

Table A-9 CIFS> homedir commands

Command	Definition
`homedir delete`	Deletes a home directory share.
`homedir deleteall`	Deletes the home directories.
`homedir quota`	Enables use of quotas on home directory file systems.
`homedir set`	Sets the home directory for the specified user. If the home directory does not exist for the specified user, this command creates that user's home directory.
`homedir show`	Displays information about home directories.

CIFS mode - local commands

The `CIFS> local` commands enable you to create, delete, and display information about local users and local groups.

Table A-10 CIFS> local commands

Command	Definition
`local group add`	Creates a local CIFS group.
`local group delete`	Deletes a local CIFS group.
`local group show`	Displays the list of available local groups you created.
`local password`	The default password for a newly-created account is the same as the user name. You can change the default password using the `CIFS> local password` command. The maximum password length is eight characters.
`local user add`	Adds a new user to CIFS. You can add the user to a local group, by entering the group name in the optional *grouplist* variable. Before you add the user to a grouplist, you must create the *grouplist*. When you create a local user, Veritas Access assigns a default password to the new account. The default password is the same as the user name. For example, if you enter *usr1* for the user name, the default password is also *usr1*.

Table A-10 CIFS> local commands *(continued)*

Command	Definition
`local user delete`	Deletes local user accounts.
`local user members`	Adds a user to one or more groups. For existing users, this command changes a user's group membership.
`local user show`	Displays the user ID and lists the groups to which the user belongs. If you do not enter an optional *username*, the command lists all CIFS existing users.

CIFS mode - server commands

The `CIFS> server` commands are used to start, stop, and show the status of the CIFS server.

Table A-11 CIFS> server commands

Command	Definition
`server start`	Starts the server. Starts the service in standalone mode. The CIFS server joins the Active Directory domain only when the server is started after issuing the `CIFS> set security` command. Starts the server and causes it to leave the old domain and join the new Active Directory domain. You can only issue this command after you enter the `CIFS> set security` command.
`server status`	Checks the status of the CIFS server.
`server stop`	Stops the CIFS server if it is running.

CIFS mode - set commands

The `CIFS> set` command configures settings for the CIFS server.

In standalone mode you do not need to set the `domaincontroller`, `domainuser`, or `domain`.

To configure CIFS for the AD domain mode, you must first set the `domaincontroller`, `domainuser`, and `domain` before setting security for the domain.

Table A-12 CIFS> set commands

Command	Definition
`set aio_size`	Allows you to set an Asynchronous I/O (AIO) read/write size with an unsigned integer.
`set allow_trusted_domains`	Enables or disables the ability to specify the trusted domains that are allowed access to a CIFS server.
`set domain`	Sets the name of the domain for the AD domain mode that Veritas Access will join.
`set domaincontroller`	Sets the domain controller server name.
`set domainuser`	Sets the name of the domain user. The domain user's credentials will be used at the domain controller while joining the domain. Therefore, the domain user should be an existing AD user who has the permission to perform the join domain operation.
`set homedirfs`	Specifies one or more file systems to be used for home directories.
`set idmap_backend`	Veritas Access maps between the domain users and groups (their identifiers) and local representation of these users and groups. Information about these mappings can be stored locally on Veritas Access or remotely using the DC directory service. Veritas Access uses the idmap_backend configuration option to decide where this information is stored. This option can be set to one of the following: ■ ad ■ hash ■ ldap ■ rid
`set ntlm_auth`	Enables or disables NTLM.
`set security`	Sets security for the domain.
`set security user`	Sets security to user. This is the default value.
`set workgroup`	Sets the workgroup name. If the name of the WORKGROUP or NETBIOS domain name is different from the domain name, use this command to set the WORKGROUP name.

CIFS mode - share commands

The `CIFS> share` commands are used to manage CIFS shares.

Table A-13 Manage the CIFS shares commands

Command	Definition
`share add`	Exports a file system with the given *sharename* or re-export new options to an existing share. The new options are updated after this command is run. This CIFS command, which creates and exports a share, takes as input the name of the file system which is being exported, the share name, and optional attributes. You can use the same command for a share that is already exported. You can do this if it is required to modify the attributes of the exported share. A file system used for storing users home directories cannot be exported as a CIFS share, and a file system that is exported as a CIFS share cannot be used for storing users' home directories.
`share allow`	Allows only the specified users and groups to access the share. If `all` is specified, then default access restrictions are restored on the share. By default, all users and groups are allowed to access the share.
`share delete`	Stops the associated file system from being exported. Any files and directories which may have been created in this file system remain intact; they are not deleted as a result of this operation.
`share deny`	Denies the specified users and groups access to the share. If `all` is specified, then all the users and groups are not able to access the share. By default, none of the users or groups are denied access to the share. **Note:** If a user or group is present in both the `share allow` and `share deny` list, then access is denied to that user or group.
`share modify`	Re-exports the file system with the given share name.
`share show`	Displays information on one or all exported shares. The information is displayed for a specific share includes the name of the file system that is being exported and the values of the share options.

Replication mode commands

The `Replication> mode` includes the commands to perform replication services.

Table A-14 Replication mode commands

Command	Definition
`service`	Starts, stops, and displays the status of the replication service.
`config`	Exports and imports the public keys and authenticates the source and the destination clusters for replication service.
`bwlimit`	Manages bandwidth allocation across links between clusters to minimize impact on performance.
`repunit`	Creates, displays, and deletes repunit (replication unit) definitions. A repunit defines the exact items (such as a file system) that you want to replicate.
`exclunit`	Creates, displays, modifies, and deletes exclunit (excluding unit) definitions. An exclunit defines items (such as directories and files) that you do not want to replicate.
`schedule`	Creates, displays, modifies, and deletes replication schedules.
`job`	Creates, displays, modifies, disables, and destroys the job definitions.
`rpo`	Displays the replication job recovery point objective (RPO) report.