Cloudian HyperStore Installation Guide

Version 7.2

This page left intentionally blank

This page left intentionally blank

Contents

This page left intentionally blank

Chapter 1. HyperStore Installation Introduction

This documentation describes how to do a **fresh installation** of Cloudian HyperStore 7.2.

For instructions on **upgrading** to 7.2 from an older HyperStore version, see "Upgrading Your HyperStore Software Version" in the *HyperStore Administrator's Guide*.

If you do not yet have the HyperStore 7.2 package, you can obtain it from the Cloudian FTP site *ftp.cloudian.com*. You will need a login ID and password (available from Cloudian Support). Once logged into the FTP site, change into the *Cloudian_HyperStore* directory and then into the *cloudian-7.2* sub-directory. From there you can download the HyperStore software package, which is named *CloudianHyperStore-7.2.bin*.

Note The HyperStore ISO file (with file name extension *.iso*) is intended for setting up a HyperStore Appliance machine. Do not use this on other host hardware.

To install and run HyperStore software you need a **HyperStore license file** — either an evaluation license or a production license. If you do not have a license file you can obtain one from your Cloudian sales representative or by registering for a free trial on the Cloudian website.

Note Users of the AWS MMS version of HyperStore: Use the *HyperStore for AWS MMS Quick Start Guide* to set up your system. That document includes information specific to setting up HyperStore to work with AWS MMS, as well as high level instructions for installing an on-premise HyperStore cluster. You can refer to this *HyperStore Installation Guide* if you need more detail on the topic of installing a HyperStore cluster.

This page left intentionally blank

Chapter 2. Preparing Your Environment

Before installing HyperStore, Cloudian recommends that you prepare these aspects of your environment:

- **"DNS Set-Up"** (page 4)
- **"Load Balancing"** (page 7)

2.1. DNS Set-Up

For your HyperStore system to be accessible to external clients, you must configure your DNS name servers with entries for the HyperStore service endpoints. **Cloudian recommends that you complete your DNS configuration prior to installing the HyperStore system.** This section describes the required DNS entries.

> **Note** If you are doing just a small evaluation and do not require that external clients be able to access any of the HyperStore services, you have the option of using the lightweight domain resolution utility *dnsmasq* which comes bundled with HyperStore -- rather than configuring your DNS environment to support HyperStore service endpoints. If you're going to use *dnsmasq* you can skip ahead to **"Preparing Your Nodes"** (page 9). You will subsequently use the *configure-dnsmasq* option when you launch the HyperStore installation script, as described later in this document.

The table that follows shows the DNS entries that you must configure on your name servers to resolve HyperStore service endpoints. By default the HyperStore system automatically derives service endpoint values from your organization’s domain, which you will supply when you run the HyperStore interactive installer. The table shows the default format of each service endpoint. The default S3 endpoint formats are consistent with the format that Amazon uses for its S3 endpoints.

If you do not want to use the default service endpoint formats, the HyperStore system allows you to specify custom endpoint values during the installation. If you intend to create custom endpoints, configure DNS entries to resolve the custom endpoint values that you intend to use, rather than the default-formatted endpoint values shown below. Make a note of the custom endpoints for which you configure DNS entries, so that later you can correctly specify those custom endpoints when you perform the HyperStore installation.

In a production environment, each of these service endpoints must resolve to the IP addresses of two or more load balancers (configured for fail-over), with the load balancers in turn distributing request traffic across **all** the nodes in the cluster.

> **Note** In a **multi-region system**, each region's S3 endpoint should resolve to load balancers in that region, which distribute traffic across nodes within that region. By contrast the system-wide Admin and CMC service endpoints should resolve to load balancers in the **default service region** which distribute traffic only to nodes in the default service region. For background information on service regions see "Service Regions Feature Overview" in the "Major Features" section of the HyperStore Administrator's Guide. Note that only Admin Service nodes in the default region support the full Admin API functionality.

> **Note** In an evaluation or testing environment, using round-robin DNS is an acceptable alternative to using load balancers. See the Example that follows below the table. **Do not use round-robin DNS in a production environment.**

DNS Entry	Default Format and Example	Description
S3 service endpoint	*s3-<region>.<your-domain>* *s3-tokyo.enterprise.com*	This is the service endpoint to which S3 client applications will submit requests.

DNS Entry	Default Format and Example	Description
(**one per service region**)		The *<region>* segment indicates the HyperStore service region. You must choose a service region name for your HyperStore installation, even if you have only one service region. Only alphanumeric characters and dashes are allowed in the name. If you are installing a HyperStore system across multiple service regions, each region will have its own S3 service endpoint, and therefore you must create a DNS entry for each of those region-specific endpoints — for example *s3-tokyo.enterprise.com* and *s3-osaka.enterprise.com*. If you want to use a **custom S3 endpoint** that does not include a region string, the installer allows you to do so. Note however that if your S3 endpoints lack region strings the system will not be able to support the region name validation aspect of AWS Signature Version 4 authentication for S3 requests (but requests can still succeed without the validation). If you want to use **multiple S3 endpoints per service region** (for example, having different S3 endpoints resolve to different data centers within one service region), the installer allows you to do this. For this approach, the recommended syntax is *s3-<region>.<dcname>.<domain>* — for example *s3-tokyo.dc1.enterprise.com* and *s3-tokyo.dc2.enterprise.com*.
S3 service endpoint wildcard (**one per service region**)	**.s3-<region>.<your-domain>* **.s3-tokyo.enterprise.com*	This S3 service endpoint wildcard entry is necessary to resolve S3 requests pertaining to specific storage buckets (which is nearly all S3 requests).
S3 static website service endpoint (**one per service region**)	*s3-website-<region>.<your-domain>* *s3-website-tokyo.enterprise.com*	This S3 service endpoint is used for buckets configured as static websites.
S3 static website endpoint wildcard (**one per service region**)	**.s3-website-<region>.<your-domain>* **.s3-website-tokyo.enterprise.com*	This S3 static website endpoint wildcard entry is necessary to make S3 requests resolvable, for buckets configured as static websites.
Admin Service endpoint (**one per entire system**)	*s3-admin.<your-domain>* *s3-admin.enterprise.com*	This is the service endpoint for HyperStore's Admin API and also for HyperStore's implementation of the IAM API. The Cloudian Management Console accesses these APIs, and you

DNS Entry	Default Format and Example	Description
		can also access these APIs directly with a third party client (such as a command line tool like *cURL*). **Note** The Admin Service and IAM Service use different ports as described in **"HyperStore Listening Ports"** (page 25).
Cloudian Management Console (CMC) domain (**one per entire system**)	*cmc.<your-domain>* *cmc.enterprise.com*	The CMC is HyperStore's web-based console for making S3 requests (such as creating storage buckets or uploading objects) and performing system administration tasks.

2.1.0.1. Port Requirements

For HyperStore listening port requirements see **"HyperStore Listening Ports"** (page 25).

2.1.0.2. Round-Robin DNS Example

Below is an example of a round-robin DNS configuration for a three-node HyperStore system that has only one service region. The organization's top-level domain is *enterprise.com*. Note that HyperStore uses a peer-to-peer architecture in which the S3 Service, Admin Service, and CMC all run on every HyperStore node, and thus in the example all of the nodes are part of the round-robin for each service endpoint.

> **Note** In an evaluation or testing environment, using round-robin DNS is an acceptable alternative to using load balancers. **Do not use round-robin DNS in a production environment.**

> **Note** Because the CMC requires persistent ("sticky") sessions with clients, using round-robin DNS is inappropriate for the CMC, even in a non-production environment. Consequently in the example below there is only one A record for the CMC.

```
s3-tokyo.enterprise.com IN A 10.1.1.1
                             10.1.1.2
                             10.1.1.3
*.s3-tokyo.enterprise.com IN A  10.1.1.1
                                10.1.1.2
                                10.1.1.3
s3-website-tokyo.enterprise.com IN A 10.1.1.1
                                     10.1.1.2
                                     10.1.1.3
*.s3-website-tokyo.enterprise.com IN A 10.1.1.1
                                       10.1.1.2
                                       10.1.1.3
s3-admin.enterprise.com IN A 10.1.1.1
                             10.1.1.2
                             10.1.1.3
cmc.enterprise.com IN A 10.1.1.1
```

2.2. Load Balancing

HyperStore uses a peer-to-peer architecture in which each node in the cluster can service requests to the S3, Admin, or CMC service endpoints. **In a production environment you must use load balancers to distribute S3, Admin, and CMC service endpoint requests** evenly across all the nodes in your cluster. In your DNS configuration the S3, Admin, and CMC service endpoints should resolve to the IP addresses of your load balancers; and the load balancers should in turn distribute request traffic across all your nodes. Cloudian recommends that you **set up your load balancers prior to installing the HyperStore system**.

For high availability it is preferable to use two or more load balancers configured for failover between them (as versus having just one load balancer which would then constitute a single point of failure). The load balancers could be commercial products or you can use open source technologies such as **HAProxy** (load balancer software for TCP/HTTP applications) and **Keepalived** (for failover between two or more load balancer nodes). If you use software-defined solutions such as these open source products, for best performance you should install them on dedicated load balancing nodes -- not on any of your HyperStore nodes.

For detailed guidance on load balancing set-up, request a copy of the *HyperStore Load Balancing Best Practice Guide* from your Cloudian Sales Engineering representative.

Note For an evaluation or testing environment round-robin DNS is an acceptable alternative to using load balancers. For more information see **"DNS Set-Up"** (page 4). **Do not use round-robin DNS in a production environment.**

Note The HyperStore S3 Service supports **PROXY Protocol** for incoming connections from a load balancer. This is disabled by default, but after HyperStore installation is complete you can enable it by configuration if you wish. For more information see s3_proxy_protocol_enabled in **common.csv**.

Note In a **multi-region system**, each region's S3 endpoint must resolve to load balancers in that region, which distribute traffic across nodes within that region. By contrast the system-wide Admin and CMC service endpoints must resolve to load balancers in the **default service region** which distribute traffic only to nodes in the default service region. For background information on service regions see "Service Regions Feature Overview" in the "Major Features" section of the HyperStore Administrator's Guide. Note that only Admin Service nodes in the default region support the full Admin API functionality.

This page left intentionally blank

Chapter 3. Preparing Your Nodes

To prepare your hosts for HyperStore software installation:

First confirm that each host meets HyperStore **"Host Hardware and OS Requirements"** (page 9).

Then complete these node preparation tasks in this order:

1. **"Installing HyperStore Prerequisites"** (page 12)
2. **"Configuring Network Interfaces, Time Zone, and Data Disks"** (page 15)
3. **"Running the Pre-Install Checks Script"** (page 17)

3.1. Host Hardware and OS Requirements

Subjects covered in this section:

- ***"Hardware Requirements"** (page 9)*
- ***"Operating System Requirements"** (page 10)*
- ***"Host Firewall Services Must Be Disabled"** (page 10)*
- ***"Python 2.7.x is Required"** (page 11)*
- ***"Automatic Exclusions to OS Package Updates"** (page 11)*
- ***"Load Balancing Requirements"** (page 11)*

3.1.1. Hardware Requirements

The table below shows the recommended and minimum hardware specifications for individual host machines in a HyperStore system. Only Intel x86-64 systems are supported. (AMD x86-64 may work, but has not been tested.)

Note For guidance regarding how many nodes you should use to meet your initial workload requirements, consult with your Cloudian sales or support representative. For guidance about ongoing HyperStore capacity management and cluster resizing, see "Cluster Resizing Feature Overview" in the *HyperStore Administrator's Guide*.

Recommended for production systems	• 1 CPU, 8 cores • 128GB RAM • 2 x 960GB SSD (for RAID-1 mirrored hosting of the OS as well as Cassandra and Redis databases storing system metadata) • 12 x 4TB HDD (for *ext4* file systems storing object data) (JBOD, no RAID) • 2 x 10GbE Ports **Note** If you plan to use **erasure coding** for object data storage, 2 CPUs per node is recommended. Also, be aware that the higher your erasure coding *m* value (such as with *k*+*m* = 9+3 or 8+4), the higher the need for

	Cassandra metadata storage capacity. Consult with your Cloudian representative to ensure that you have adequate Cassandra storage capacity to support your desired *m* value.
Minimum for production systems	• 1 CPU, 8 cores • 64GB RAM • 2 x 480GB SSD (for RAID-1 mirrored hosting of the OS as well as Cassandra and Redis databases storing system metadata) • 12 x 4TB HDD (for *ext4* file systems storing object data) (JBOD, no RAID) • 2 x 10GbE Ports
Minimum for installation	HyperStore software can be installed on a single host that has just one data drive. The host should have at least 1GB of hard drive space, at least 16GB RAM, and preferably at least 8 processor cores. If you install HyperStore software on a host with less resources than this, the install script will display a warning about the host having less than recommended resources. If you try to install HyperStore software on a host with less 100MB hard drive space or less than 2GB RAM, the installation will abort.

3.1.2. Operating System Requirements

To install HyperStore 7.2 you must have a **RHEL 7.x or CentOS 7.x** Linux operating system on each host. HyperStore 7.2 does not support installation on RHEL/CentOS 6.x . Also, HyperStore does not support other types of Linux distribution, or non-Linux operating systems.

If you have not already done so, install RHEL 7.x or CentOS 7.x in accordance with your hardware manufacturer's recommendations.

Note Cloudian recommends using RHEL/CentOS **7.6 or newer**.

3.1.2.1. Host Firewall Services Must Be Disabled

To install HyperStore the following services **must be disabled on each HyperStore host machine**:

- *firewalld*
- *iptables*
- *SELinux*

To disable ***filewalld***:

```
[root]# systemctl stop firewalld
[root]# systemctl disable firewalld
```

RHEL/CentOS 7 uses *firewalld* by default rather than the *iptables* service (*firewalld* uses *iptables* commands but the *iptables* service itself is not installed on RHEL/CentOS by default). So you do not need to take action in regard to ***iptables*** unless you installed and enabled the *iptables* service on your hosts. If that's the case, then disable the *iptables* service.

To disable **SELinux**, edit the configuration file */etc/selinux/config* so that *SELINUX=disabled*. Save your change and then restart the host.

HyperStore nodes sometimes communicate with each other via JMX, and when they do, after initial connection establishment on a designated JMX port a random port is used for continued communication. Therefore **there cannot be any port restrictions on communication between HyperStore nodes**. Consequently, the HyperStore installation will abort if *firewalld*, *SELinux*, or *iptables* is running on a host.

HyperStore includes a built-in firewall service (a HyperStore-custom version of the *firewalld* service) that is configured to protect HyperStore internal services while keeping HyperStore public services open. In fresh installations of HyperStore 7.2 or later, the HyperStore firewall is enabled by default upon the completion of HyperStore installation. In HyperStore systems originally installed as a version older than 7.2 and then later upgraded to 7.2 or newer, the HyperStore firewall is available but is disabled by default. After installation of or upgrade to HyperStore 7.2 or later, you can enable or disable the HyperStore firewall by using the installer's Advanced Configuration Options. For instructions see "Configuring the HyperStore Firewall" in the System Configuration section of the *HyperStore Administrator's Guide*.

> **Note** For more information about HyperStore port usage see **"HyperStore Listening Ports"** (page 25).

3.1.2.2. Python 2.7.x is Required

The HyperStore installer **requires Python version 2.7.x**. The installer will abort with an error message if any host is using Python 3.x. To check the Python version on a host:

```
[root]# python --version
Python 2.7.5
```

After verifying that your hosts meet hardware and OS requirements:

- If you are doing a **fresh cluster installation**, proceed to **"Installing HyperStore Prerequisites"** (page 12).
- If you are **adding nodes to an existing cluster**, return to and continue with the "Preparing to Add Nodes" part of the "Adding Nodes" section of your HyperStore Administrator's Guide

3.1.2.3. Automatic Exclusions to OS Package Updates

As part of HyperStore installation, the HyperStore installation script will install prerequisites including Puppet, Facter, Ruby, and Salt on your HyperStore host machines. If you subsequently use *yum update* or *yum upgrade* to update your OS packages, HyperStore automatically excludes Puppet, Facter, Ruby, and Salt related packages from the update. This is to ensure that only the correct, tested versions of these packages are used together with HyperStore. After HyperStore installation, this auto-exclusion is configured in the */etc/yum/pluginconf.d/versionlock.list* file on your host machines. You can review that file if you wish to see specifically which packages are "locked" at which versions, but do not remove any entries from the lock list.

3.1.3. Load Balancing Requirements

In a HyperStore production environment you must use load balancers to distribute incoming service requests. For more information see **"DNS Set-Up"** (page 4) and **"Load Balancing"** (page 7).

> **Note** For an evaluation or testing environment round-robin DNS is an acceptable alternative to using load balancers. For more information see **"DNS Set-Up"** (page 4). **Do not use round-robin DNS in a production environment.**

3.2. Installing HyperStore Prerequisites

> **Note** These instructions assume that you have already configured basic networking on each of your nodes. In particular, each node must already be configured with a hostname and IP address, and the nodes must be able to reach each other in the network.

Follow these steps to install and configure HyperStore prerequisites on all of your nodes. Working from a single node you will be able to perform this task for your whole cluster.

1. Log into one of your nodes as *root*. This will be the node from which you will orchestrate the HyperStore installation for your whole cluster. Also, as part of the HyperStore installation, **Puppet** configuration management software will be installed and configured in the cluster, and this node will become the "Puppet Master" node for purposes of ongoing cluster configuration management. Note that the Puppet Master node **must** be one of your HyperStore nodes. It cannot be a separate node outside of your HyperStore cluster.

2. On the node that you've chosen to become your Puppet Master node, download or copy the HyperStore product package (*CloudianHyperStore-7.2.bin* file) into any directory. Also copy your Cloudian license file (**.lic* file) into that same directory. Pay attention to the license file name since you will need the file name in the next step.

3. In that directory run the commands below to unpack the HyperStore package:

```
[any-directory]# chmod +x CloudianHyperStore-7.2.bin
[any-directory]# ./CloudianHyperStore-7.2.bin <license-file-name>
```

This creates an installation staging directory named */opt/cloudian-staging/7.2*, and extracts the HyperStore package contents into the staging directory.

> **Note** The installation staging directory must persist for the life of your HyperStore system. Do not delete the staging directory after completing the install.

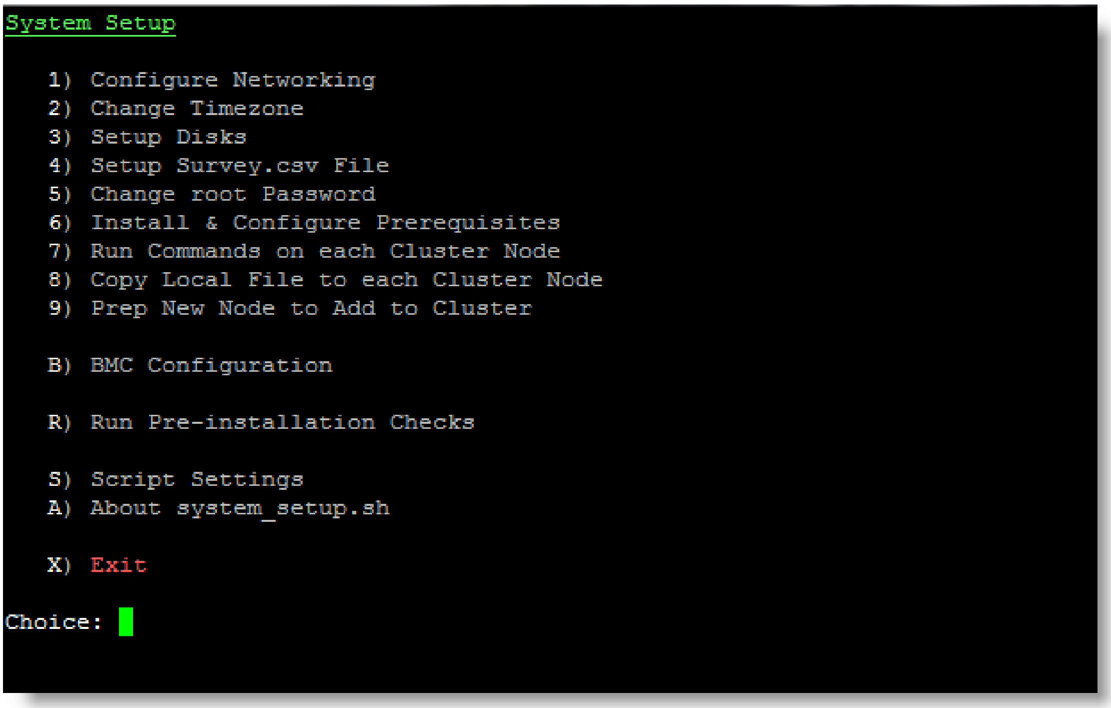
4. Change into the installation staging directory:

```
[any-directory]# cd /opt/cloudian-staging/7.2
```

5. In the staging directory, launch the *system_setup.sh* tool:

```
[7.2]# ./system_setup.sh
```

This displays the tool's main menu.



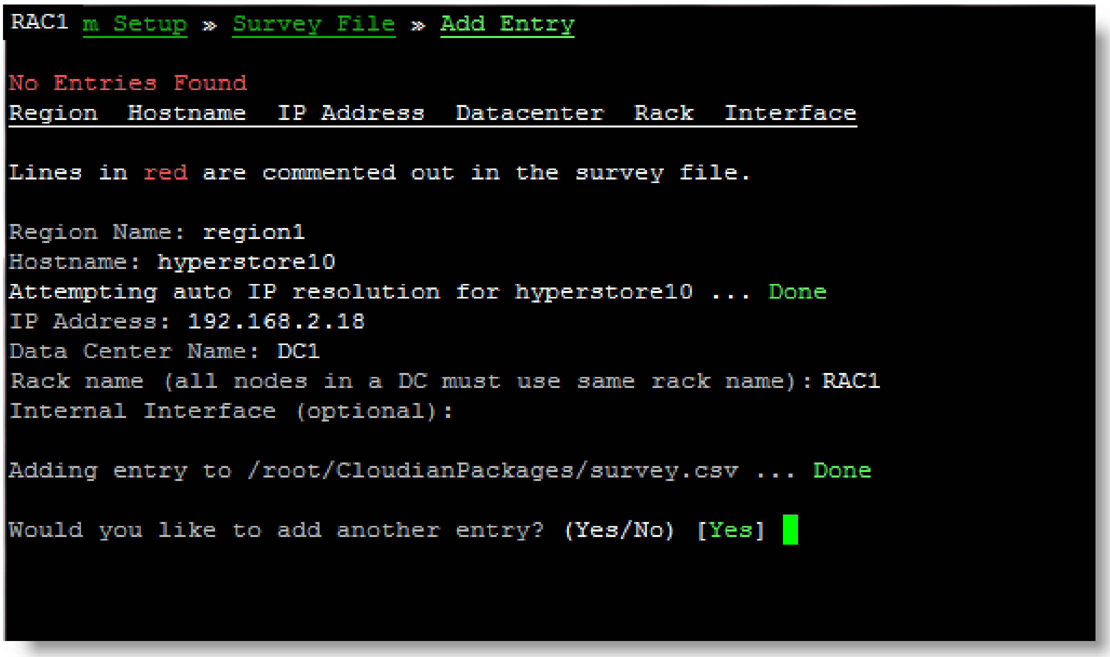
```
System Setup

   1) Configure Networking
   2) Change Timezone
   3) Setup Disks
   4) Setup Survey.csv File
   5) Change root Password
   6) Install & Configure Prerequisites
   7) Run Commands on each Cluster Node
   8) Copy Local File to each Cluster Node
   9) Prep New Node to Add to Cluster

   B) BMC Configuration

   R) Run Pre-installation Checks

   S) Script Settings
   A) About system_setup.sh

   X) Exit

Choice:
```

6. From the setup tool's main menu, enter "**4**" for **Setup Survey.csv File** and follow the prompts to create a cluster survey file with an entry for each of your HyperStore nodes (including the Puppet Master node). For each node you will enter a region name, hostname, public IP address, data center name, and rack name.

    - For each node the hostname that you enter must exactly match the node's hostname -- as would be returned by running the *hostname* command on the node.

        > **Note:** If you enter a hostname that includes upper case letters the setup tool automatically converts the hostname to entirely lower case letters. This is true both for the setup tool's function for creating the survey file, and for the setup tool's function for setting/changing a node's hostname (as described in **"Configuring Network Interfaces, Time Zone, and Data Disks"** (page 15)).

    - For the region, data center, and rack name the only allowed character types are ASCII alphanumerical characters and dashes. For the region name letters must be lower case.
    - Within a data center, **use the same "rack name" for all of the nodes**, even if some nodes are on different physical racks than others.
    - Make sure the region name matches the region string that you use in your S3 endpoints in your **"DNS Set-Up"** (page 4)



```
RAC1 m Setup » Survey File » Add Entry

No Entries Found
Region  Hostname  IP Address  Datacenter  Rack  Interface

Lines in red are commented out in the survey file.

Region Name: region1
Hostname: hyperstore10
Attempting auto IP resolution for hyperstore10 ... Done
IP Address: 192.168.2.18
Data Center Name: DC1
Rack name (all nodes in a DC must use same rack name): RAC1
Internal Interface (optional):

Adding entry to /root/CloudianPackages/survey.csv ... Done

Would you like to add another entry? (Yes/No) [Yes]
```

For each node there is also an optional prompt for specifying the node's internal interface name. You only need to provide this information if the node is using a different internal interface than the rest of the nodes in the cluster. If all nodes use the same internal interface you can leave this value empty for each node (later in the installation process you will specify a default internal interface name for the cluster).

After you've added an entry for each node, return to the setup tool's main menu.

> **Note** Based on your input at the prompts, the setup tool creates a survey file named *survey.csv*, in your installation staging directory. This file must remain in your staging directory -- do not delete or move it. For information about the contents of the survey file, see the Installation Reference topic **"Cluster Survey File (survey.csv)"** (page 33).

7. If you want to change the root password for your nodes, do so now by entering "**5**" for **Change Root Password** and following the prompts. It's recommended to use the same password for each node. Otherwise the pre-installation cluster validation tool described later in the procedure will not be fully functional.
8. Back at the setup tool's main menu enter "**6**" for **Install & Configure Prerequisites**. When prompted about whether you want to perform this action for all nodes in your survey file enter "**yes**". The tool will connect to each of your nodes in turn and install the prerequisite packages. You will be prompted to provide the root password either for the whole cluster (if, as recommended, each node has the same root password) or for each node in turn (if the nodes have different passwords). When the prerequisite installation completes for all nodes, return to the setup tool's main menu.

> **Note:** If *firewalld* is running on your hosts the setup tool prompts you for permission to disable it. And if *Selinux* is enabled on your hosts, the tool automatically disables it without prompting for permission (or more specifically, changes it to "permissive" mode for the current running session and changes the configuration so it will be disabled for future boots of the hosts). For information on why these services must be disabled on HyperStore host machines see **"Operating System Requirements"** (page 10).

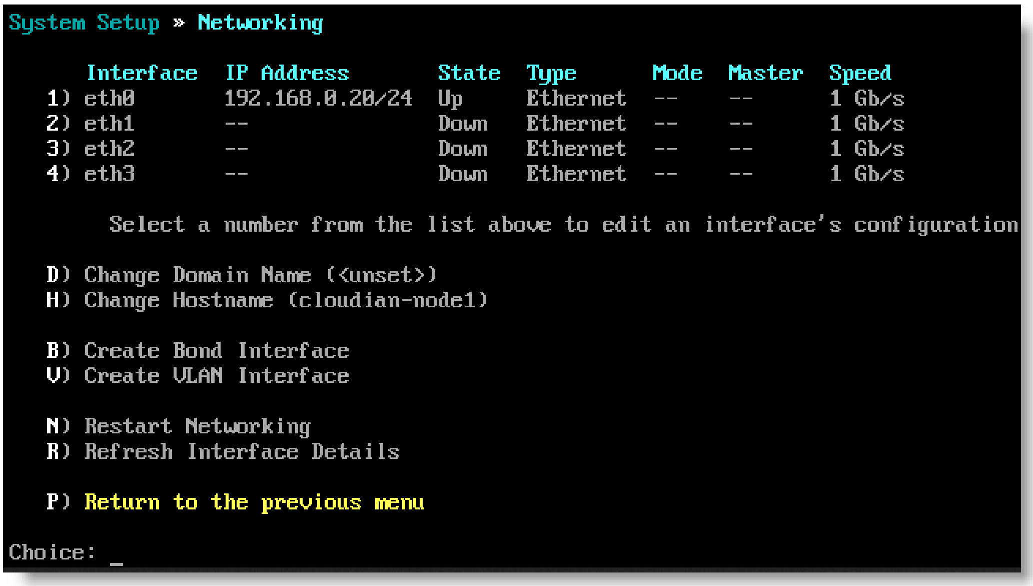
After the prerequisite installation completes for all nodes and you're back at the setup tool's main menu. Next, proceed to **"Configuring Network Interfaces, Time Zone, and Data Disks"** (page 15).

3.3. Configuring Network Interfaces, Time Zone, and Data Disks

Having finished **"Installing HyperStore Prerequisites"** (page 12), you should be at the main menu of the *system_setup.sh* tool, in the installation directory on your Puppet Master node. Next follow these steps to configure network interfaces (if you haven't already fully configured them), set the time zone, and configure data disks on each node in your HyperStore cluster.

1. On the Puppet Master node, from the system setup tool's main menu, complete the setup of the Puppet Master node itself:

   a. From the system setup tool's main menu, enter "**1**" for **Configure Networking**. This displays the Networking configuration menu.



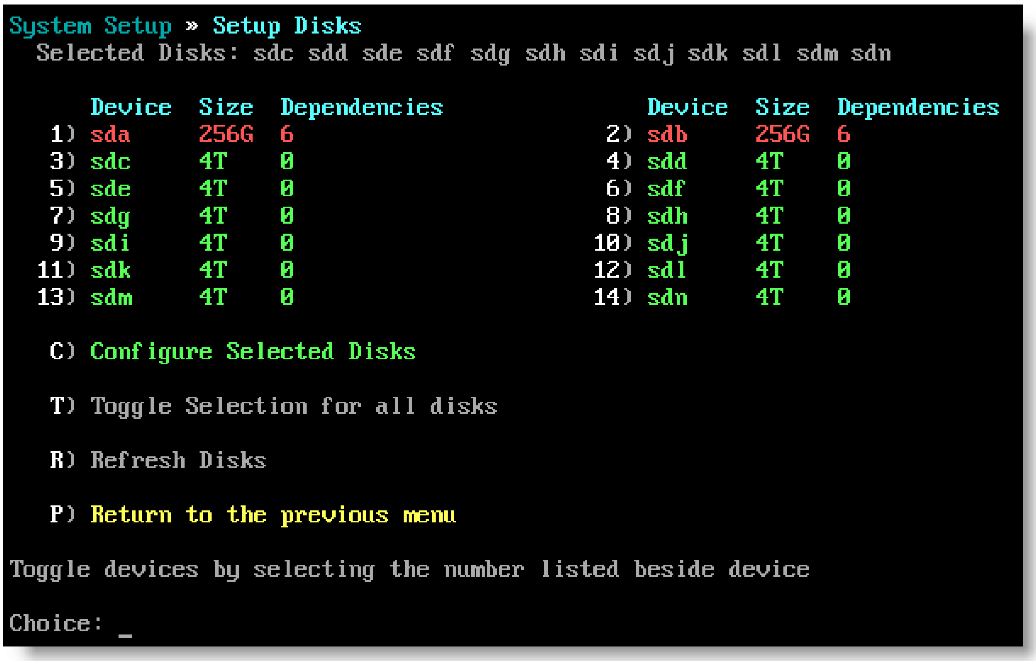
```
System Setup » Networking

      Interface  IP Address        State  Type      Mode  Master  Speed
   1) eth0       192.168.0.20/24   Up     Ethernet  --    --      1 Gb/s
   2) eth1       --                Down   Ethernet  --    --      1 Gb/s
   3) eth2       --                Down   Ethernet  --    --      1 Gb/s
   4) eth3       --                Down   Ethernet  --    --      1 Gb/s

        Select a number from the list above to edit an interface's configuration

   D) Change Domain Name (<unset>)
   H) Change Hostname (cloudian-node1)

   B) Create Bond Interface
   V) Create VLAN Interface

   N) Restart Networking
   R) Refresh Interface Details

   P) Return to the previous menu

Choice: _
```

   Here you can review the current network interface configuration for the Puppet Master node, and if you wish, perform additional configuration such as configuring an internal/back-end interface. When you are done with any desired network interface configuration changes for this node, return to the setup tool's main menu.

   > **Note** When setting/changing a node's hostname, if you enter a hostname that includes upper case letters the setup tool automatically converts the hostname to entirely lower case letters.

   b. At the setup tool's main menu, enter "**2**" for **Change Timezone** and set the time zone for this node.

   c. At the setup tool's main menu, enter "**3**" for **Setup Disks**. This displays the Setup Disks menu.



```
System Setup » Setup Disks
  Selected Disks: sdc sdd sde sdf sdg sdh sdi sdj sdk sdl sdm sdn

      Device  Size  Dependencies                Device  Size  Dependencies
   1) sda     256G  6                       2) sdb     256G  6
   3) sdc     4T    0                       4) sdd     4T    0
   5) sde     4T    0                       6) sdf     4T    0
   7) sdg     4T    0                       8) sdh     4T    0
   9) sdi     4T    0                      10) sdj     4T    0
  11) sdk     4T    0                      12) sdl     4T    0
  13) sdm     4T    0                      14) sdn     4T    0

   C) Configure Selected Disks

   T) Toggle Selection for all disks

   R) Refresh Disks

   P) Return to the previous menu

Toggle devices by selecting the number listed beside device

Choice: _
```

From the list of disks on the node select the disks to format as HyperStore data disks, for storage of S3 object data. By default the tool automatically selects all disks that are not already mounted and do not contain a */root*, */boot* or *[swap]* mount indication. Selected disks display in green font in the disk list. The tool will format these disks with *ext4* file systems and assign them mount points */cloudian1*, */cloudian2*, */cloudian3*, and so on. You can toggle (select/deselect) a disk by entering at the prompt the disk's number from the displayed list (such as "**3**"). Once you're satisfied with the selected list in green font, enter "**c**" for **Configure Selected Disks** and follow the prompts to have the tool configure the selected disks.

> **IMPORTANT:** Cloudian recommends using the HyperStore system setup tool to format and mount your data disks. **If you have already formatted and mounted your data disks using third party tools**, then instead of using the disk configuration instructions in this section follow the guidelines and instructions in **"File System Requirements"** (page 30).

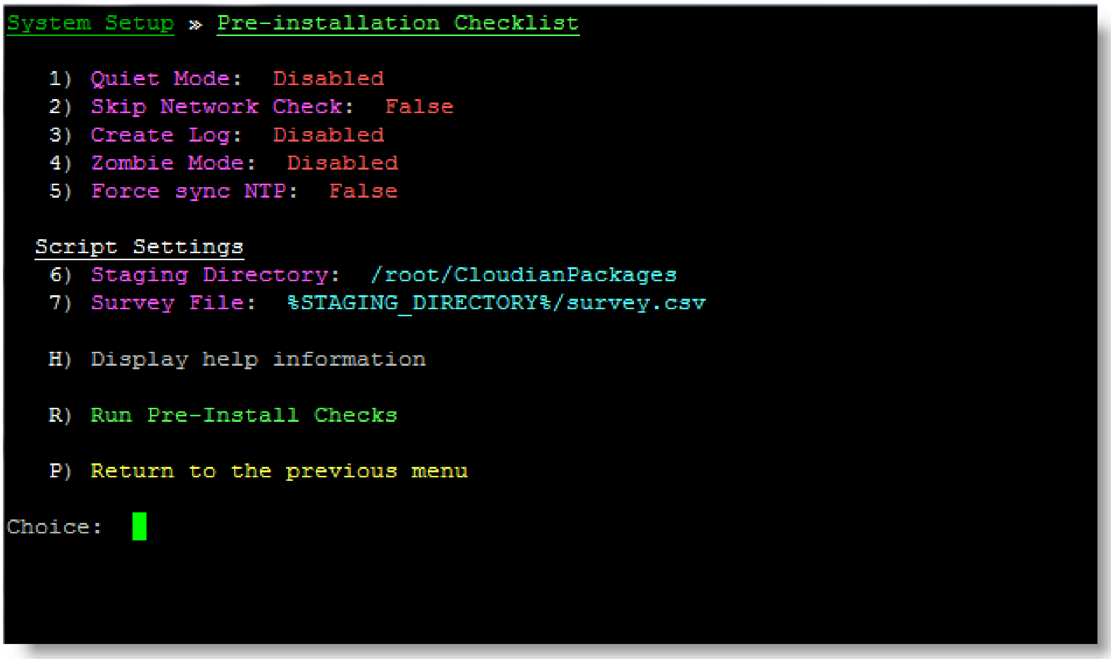
2. Next, complete the setup of the other nodes in your cluster:
   a. From the setup tool's main menu select "**9**" for **Prep New Node to Add to Cluster**.
   b. When prompted enter the IP address of one of the remaining nodes (the nodes other than the Puppet Master node), and then enter the login password for the node.
   c. Using the node preparation menu that displays:
      i. Review and complete network interface configuration for the node.
      ii. Set the time zone for the node.
      iii. Configure data disks for the node. Then return to the system setup tool's main menu.
   d. Repeat Steps "a" through "c" for each of the remaining nodes in your installation cluster.

After you've prepared all your nodes and returned to the setup tool's main menu, proceed to **"Running the Pre-Install Checks Script"** (page 17).

3.4. Running the Pre-Install Checks Script

Follow these steps to verify that your cluster now meets all HyperStore requirements for hardware, prerequisite packages, and network connectivity.

1. At the setup tool's main menu enter "**r**" for **Run Pre-Installation Checks**. This displays the Pre-Installation Checklist menu.



```
System Setup » Pre-installation Checklist

   1) Quiet Mode:  Disabled
   2) Skip Network Check:  False
   3) Create Log:  Disabled
   4) Zombie Mode:  Disabled
   5) Force sync NTP:  False

  Script Settings
   6) Staging Directory:  /root/CloudianPackages
   7) Survey File:  %STAGING_DIRECTORY%/survey.csv

   H) Display help information

   R) Run Pre-Install Checks

   P) Return to the previous menu

Choice:
```

2. From the Pre-Installation Checklist menu enter "**r**" for **Run Pre-Install Checks**. The script then checks to verify that your cluster meets all requirements for hardware, prerequisite packages, and network connectivity.

   Note The script only supports your providing one root password, so if some of your nodes do not use that password the script will not be able to check them and you may encounter errors during HyperStore installation if requirements are not met.

   At the end of its run the script outputs to the console a list of items that the script has evaluated and the results of the evaluation. You should review any “Warning” items but they don’t necessarily require action (an example is if the hardware specs are less than recommended but still adequate for the installation to proceed). **You must resolve any “Error” items before performing the HyperStore software installation**, or the installation will fail.

   When you’re done reviewing the results, press any key to continue and then exit the setup script. If you make any system changes to resolve errors found by the pre-install check, run the pre-install check again afterward to verify that your environment meets HyperStore requirements.

After your cluster has successfully passed the pre-install checks, proceed to **"Installing a New HyperStore System"** (page 19).

This page left intentionally blank

Chapter 4. Installing a New HyperStore System

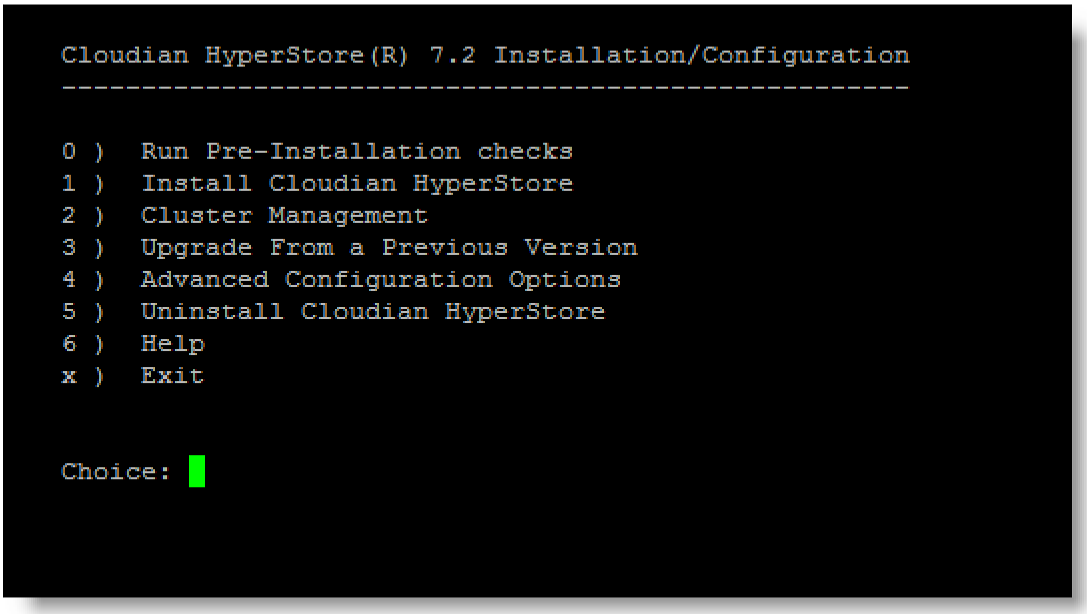
This section describes how to do a fresh installation of HyperStore 7.2 software, after **"Preparing Your Environment"** (page 3) and **"Preparing Your Nodes"** (page 9). From your Puppet Master node you can install HyperStore software across your whole cluster.

1. On your Puppet Master node, in your installation staging directory, launch the HyperStore installation script as follows:

```
[7.2]# ./cloudianInstall.sh -s survey.csv
```

> **Note If you have not configured your DNS environment** for HyperStore (see **"DNS Set-Up"** (page 4)) and you want to instead use the included *dnsmasq* utility to resolve HyperStore service endpoints, launch the install script with the *configure-dnsmasq* option as shown below. This is not appropriate for production systems.
>
> *[7.2]# ./cloudianInstall.sh -s survey.csv configure-dnsmasq*
>
> For more script launch options, see the Installation Reference topic **"cloudianInstall.sh Command Line Options"** (page 35).

When you launch the installer the main menu displays:



```
Cloudian HyperStore(R) 7.2 Installation/Configuration
-----------------------------------------------------

0 )  Run Pre-Installation checks
1 )  Install Cloudian HyperStore
2 )  Cluster Management
3 )  Upgrade From a Previous Version
4 )  Advanced Configuration Options
5 )  Uninstall Cloudian HyperStore
6 )  Help
x )  Exit


Choice:
```

> **Note** The installer menu includes an item "0" for Run Pre-Installation Checks. This is the same pre-installation check that you already ran from within the *system_setup.sh* tool as described in **"Running the Pre-Install Checks Script"** (page 17) -- so you can ignore this option in the

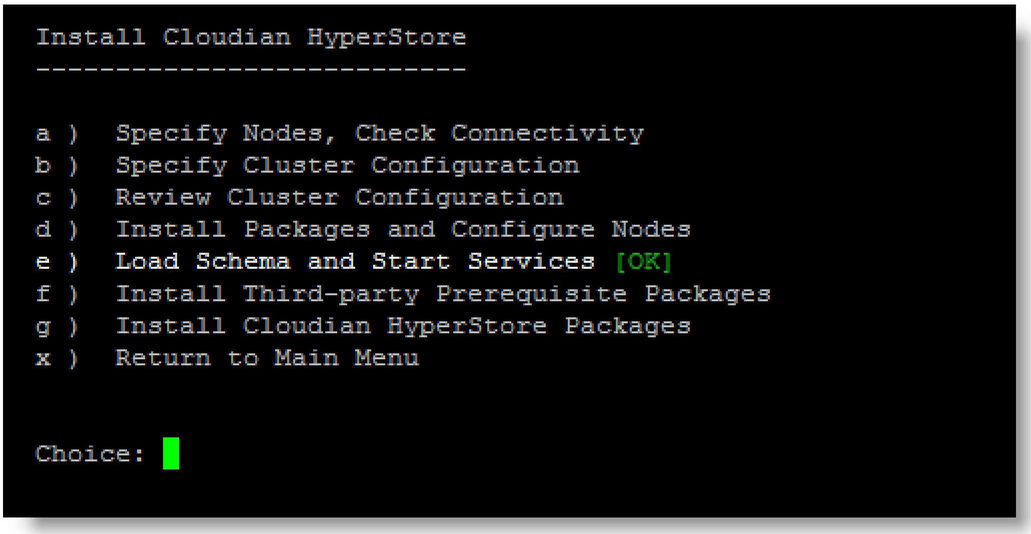
installer menu. If you did **not** run the pre-install check already, then do so from the installer menu before proceeding any further.

2. From the installer main menu, enter "**1**" for Install Cloudian HyperStore. Follow the prompts to perform the HyperStore installation across all the nodes in your cluster survey file (which you created earlier during the node preparation task).

   During the HyperStore installation you will be prompted to provide the following cluster configuration information:

   - The name of the **internal interface** that your nodes will use by default for internal cluster communications. For example, *eth1*. Cassandra, Redis, and the HyperStore Service are among the services that will utilize the internal interface for intra-cluster communications.
   - The starting "**replication strategy**" that you want to use to protect system metadata (such as usage reporting data and user account information). The replication strategy you enter must be formatted as "<datacenter_name>:<replication_#>". For example, "DC1:3" means that in the data center named DC1, three instances of each system metadata object will be stored (with each instance on a different host). If you are installing HyperStore into multiple data centers you must format this as a comma-separated list specifying the replicas per data center -- for example "DC1:2,DC2:1". The default is 3 replicas per service region, and then subsequently the system automatically adjusts the system metadata replication level based on the storage policies that you create. For more on this topic see "Storage of System Metadata" in the *HyperStore Administrator's Guide*.
   - Your **organization's domain**. For example, *enterprise.com*. From this input that you provide, the installation script will automatically derive HyperStore service endpoint values. You can accept the derived endpoint values that the script presents to you, or optionally you can enter customized endpoint values at the prompts. For S3 service endpoint the default is to have one endpoint per service region, but you also have the option of entering multiple comma-separated endpoints within a service region -- if for example you want different data centers within the region to use different S3 service endpoints. If you want to have different S3 endpoints for different data centers within the same service region, the recommended S3 endpoint syntax is *s3-<region>.<dcname>.<domain>*. See **"DNS Set-Up"** (page 4) for more details about HyperStore service endpoints.

   At the conclusion of the installation an "Install Cloudian HyperStore" sub-menu displays, with indication of the installation status. If the installation completed successfully, the "Load Schema and Start Services" menu item should show an "OK" status:



```
Install Cloudian HyperStore
---------------------------

a )  Specify Nodes, Check Connectivity
b )  Specify Cluster Configuration
c )  Review Cluster Configuration
d )  Install Packages and Configure Nodes
e )  Load Schema and Start Services [OK]
f )  Install Third-party Prerequisite Packages
g )  Install Cloudian HyperStore Packages
x )  Return to Main Menu


Choice: █
```

After seeing that the "Load Schema and Start Services" status is OK, return to the installer's main menu.

> **Note** The "Install Cloudian HyperStore" sub-menu supports re-executing specific installation operations on specific nodes or on all nodes. This may be helpful if the installer interface indicates that an operation has failed. If one of the operations in the menu indicates an error status, retry that operation by specifying the menu option letter at the prompt (such as "**e**" for "Load Schema and Start Services").

3. After installation has completed successfully, from the installer's main menu enter "**2**" for Cluster Management and then enter "**d**" for Run Validation Tests. This executes some basic automated tests to confirm that your HyperStore system is working properly. The tests include S3 operations such as creating an S3 user group, creating an S3 user, creating a storage bucket for that user, and uploading and downloading an S3 object.

After validation tests complete successfully, exit the installation tool.

For first steps to set up and try out your new HyperStore system, see "Getting Started with a New HyperStore System" in the *HyperStore Administrator's Guide*.

> **Note** For troubleshooting information, see the Installation Reference topic **"Installation Troubleshooting"** (page 24).

This page left intentionally blank

Chapter 5. HyperStore Installation Reference

This section of the installation documentation provides reference information that you may find useful in some installation scenarios and circumstances.

Note The install script's "Advance Configuration Options" are covered in the "System Configuration" section of the HyperStore Administrator's Guide.

5.1. Installation Troubleshooting

5.1.1. Installation Logs

When you run the HyperStore installer it generates the following logs that may be helpful for troubleshooting installation problems:

On the Puppet master node (on which you're running the install script):

- *<installation-staging-directory>/cloudian-installation.log*
- */var/log/puppetserver/puppetserver.log*

On each Puppet agent node (each node on which you're installing HyperStore):

- */tmp/puppet_agent.log*

Scanning these logs for error or warning messages should help you identify the stage at which the installation encountered a problem, and the nature of the problem. This information can further your own troubleshooting efforts, and also can help Cloudian Support pinpoint the problem in the event that you need assistance from Support.

> **Note** When you use *system_setup.sh* to prepare your nodes for HyperStore installation, that tool writes its logging output to *system_setup.sh.log*, in the same directory as the *system_setup.sh* tool is located (typically your installation staging directory).

5.1.2. Debug Mode

Another potentially useful source of troubleshooting information is to run the installer in debug mode:

```
[7.2]# ./cloudianInstall.sh -d
```

For example, if you encounter an error while running the installer in regular (non-debug) mode, you can exit the installer menu and then launch the installer again in debug mode. You can then either re-execute the installation starting from the beginning, or re-execute the installation starting from the step that had previously failed. If you had partially run the installation, then when you subsequently select Install Cloudian HyperStore at the main menu a sub-menu will display to let you choose from among several installation tasks to run again.

When run in debug mode, the installer will write highly granular messages to both the console and the installation log (*cloudian-installation.log*).

5.1.3. Specific Issues

ISSUE: You encounter the following warnings:

```
Warning: Could not retrieve fact fqdn
Warning: Host is missing hostname and/or domain: cloudian-singlenode
```

Solution

As suggested by the warning messages, the domain part is missing for the host named "cloudian-singlenode". To resolve this edit the */etc/hosts* file or */etc/resolv.conf* file.

1. Edit the */etc/hosts* file and make sure the following entry exists:

   ```
   Ip-address   cloudian-singlenode.MyDomain.Com  cloudian-singlenode
   ```

   - *Ip-address* should be replaced with host’s real IP address
   - *MyDomain.Com* should be replaced with your domain name of choice.

2. Edit the */etc/resolv.conf* file and make sure the following entry exists:

   ```
   Domain MyDomain.Com
   ```

   MyDomain.Com should be replaced with your domain name of choice.

Verify that the *facter fqdn* and *hostname –f* commands output ‘ cloudian-singlenode.MyDomain.Com’ to the console.

ISSUE: Puppet is unable to propagate configuration settings to the agent nodes, and in the *puppet_agent.log* and/or *puppet_server.log* you see errors indicating certificate problems or access failures.

Solution

Try going to the installer’s "Advanced Options" sub-menu and executing task [**h**] — "Remove Existing Puppet SSL Certificates". Then go back to the main menu and choose the appropriate action below, depending on what you were doing when you encounted the Puppet run failure:

- If you are doing the initial installation of your HyperStore cluster, choose "Install Cloudian HyperStore", then execute task "Install Packages and Configure Nodes [includes Run Puppet]".
- If you are performing post-installation configuration tasks, choose "Cluster Management", then execute task "Push Configuration Settings to the Cluster [Run Puppet]".

ISSUE: While working with the installation script, you get a console message indicating that Puppet access is locked.

Solution

The Puppet process can sometimes end up left in a "locked" state if a Puppet run is interrupted, such as by a Ctrl-<c> command or a host shutdown.

To unlock Puppet, go to the installer’s "Advanced Options" sub-menu and execute task [**j**] — "Remove Puppet Access Lock". Then go back to the main menu and choose the appropriate Puppet-running action below, depending on what you were doing when you encountered the Puppet lock error:

- If you are doing the initial installation of your HyperStore cluster, choose "Install Cloudian HyperStore", then execute task "Install Packages and Configure Nodes [includes Run Puppet]".
- If you are performing post-installation configuration tasks, choose "Cluster Management", then execute task "Push Configuration Settings to the Cluster [Run Puppet]".

5.2. HyperStore Listening Ports

The HyperStore system uses the listening ports specified in the table below. On each of your HyperStore nodes:

- **All ports must be open to traffic originating from other HyperStore nodes** -- not only the ports in the table below, but **all ports**.This is because HyperStore nodes sometimes communicate with each other via JMX, and when they do, after initial connection establishment on the designated JMX port a random port is used for continued communication. Do not place any port restrictions on node-to-node communications within your HyperStore cluster. This applies also to multi- data center HyperStore clusters. All HyperStore nodes must be able to communicate with all other HyperStore nodes, on all ports.
- **Only the service ports for the CMC, S3, IAM, SQS, and Admin services** (the port numbers in italics in the "Listening Port" column) **should be open to traffic originating from outside the HyperStore cluster.** All other ports must be closed to traffic from outside the cluster, for system security.

Each HyperStore node includes a built-in firewall that implements these port availability requirements. The firewall is disabled by default in HyperStore systems that were originally installed as a version older than 7.2; and enabled by default in HyperStore systems that originally installed as version 7.2 or newer. You can enable or disable the firewall on all HyperStore nodes by using the installer's Advanced Configuration Options. For instructions see "Configuring the HyperStore Firewall" in the System Configuration section of the *HyperStore Administrator's Guide*.

Service	Listening Port	Interface(s) Binded To	Purpose
Cloudian Management Console (CMC)	*8888*	All	Requests from administrators' or end users' browsers over HTTP
	8443	All	Requests from administrators' or end users' browsers over HTTPS
S3 Service	*80*	All	Requests from the CMC or other S3 client applications over HTTP
	443	All	Requests from the CMC or other S3 client applications over HTTPS
	81	All	Requests relayed by an HAProxy load balancer using the PROXY Protocol (if enabled by configuration; see s3_proxy_protocol_enabled in common.csv)
	4431	All	Requests relayed by an HAProxy load balancer using the PROXY Protocol with SSL (if enabled by configuration)
	19080	All	JMX initial connection access
IAM Service	*16080*	All	Requests from the CMC or other IAM clients over HTTP
	16443	All	Requests from the CMC or other IAM clients over HTTPS
	19084	All	JMX initial connection access

Service	Listening Port	Interface(s) Binded To	Purpose
SQS Service	*18090*	All	Simple Queue Service requests over HTTP
	18443	All	Simple Queue Service requests over HTTPS
Admin Service	*18081*	All	Requests from the CMC or other Admin API clients over HTTP
	19443	All	Requests from the CMC or other Admin API clients over HTTPS (Note: The CMC by default uses HTTPS to access the Admin Service)
	19081	All	JMX initial connection access
Redis Monitor	9078	Internal	Communication between primary and backup Redis Monitor instances
	19083	All	JMX initial connection access
HyperStore Service	19090	Internal	Data operation requests from the S3 Service
	19050	Internal	Communication between HyperStore Service instances
	19082	All	JMX initial connection access
Redis DBs	6379	Internal	Requests to the Redis Credentials DB from the S3 Service, HyperStore Service, or Admin Service; and communication between Redis Credentials instances
	6380	Internal	Requests to the Redis QoS DB from the S3 Service, HyperStore Service, or Admin Service; and communication between Redis QoS instances
Cassandra	9042	Internal	Data operations requests from the S3 Service, HyperStore Service, or Admin Service, using CQL protocol
	9160	Internal	Data operations requests from the S3 Service, HyperStore Service, or Admin Service, using Thrift protocol
	7000	Internal	Communication between Cassandra instances

Service	Listening Port	Interface(s) Binded To	Purpose
	7199	All	JMX initial connection access
Cloudian Monitoring Agent	19070	All	Requests from the Cloudian Monitoring Data Collector
Puppet Master	8140	Internal	On your Puppet Master node (the HyperStore node from which you will manage cluster installation and configuration) this port will service incoming requests from Puppet agents on your other HyperStore nodes
Salt Master	4505	Internal	On your Salt Master node (which will be the same node that is the Puppet Master), this is the port to which Salt agents ("minions") establish a persistent connection so that the Master can publish to the minions.
	4506	Internal	Salt minions connect to this port on the Salt Master as needed to send results to the Salt Master, and to request files and minion-specific data values.
JMX	Random	All	When HyperStore nodes communicate with each other via JMX (Java Management Extensions protocol), after initial connection establishment on the designated JMX port a random port is used for continued communication. This behavior is not specific to HyperStore but rather is the default behavior of JMX communications. Because of this behavior, on each HyperStore node all ports must be open to traffic originating from other HyperStore nodes.
SSH	22	All	The HyperStore installer accesses this SSH port on each node on which you are installing HyperStore software (during initial cluster install or if you subsequently expand your cluster)

Service	Listening Port	Interface(s) Binded To	Purpose
NTP	123	All	NTP port for time synchronization between nodes

Note The Cloudian Monitoring Data Collector uses ICMP to check whether each node is reachable. If ICMP is unavailable the Data Collector tries to use Echo (port 7) to check whether each node is reachable.

5.2.1. Multi-DC Considerations

If you are installing HyperStore across multiple data centers and/or multiple service regions, the HyperStore nodes in each data center and region will need to be able to communicate with the HyperStore nodes in the other data centers and regions. This includes services that listen on the internal interface (such as Cassandra, the HyperStore Service, and Redis). Therefore you will need to configure your networking so that the internal networks in each data center and region are connected to each other (for example, by using a VPN).

5.3. Outbound Internet Access

The HyperStore installation process does not require outbound internet access. However, the following HyperStore features do access the internet once the system is in operation. If you use forward proxying in your environment, after HyperStore installation you may want to set up forward proxying to support these HyperStore features:

- **Smart Support** — The Smart Support feature (also known as "Phone Home") securely transmits HyperStore daily diagnostic information to Cloudian Support over the internet. HyperStore supports configuring this feature to use an explicit forward proxy for its outbound internet access (after installation, the relevant settings in common.csv are phonehome_proxy_host and the other *phonehome_proxy_** settings that follow after it).
- **Auto-Tiering and Cross-Region Replication** — If you want to use either the auto-tiering feature or the cross-region replication feature (CRR), the S3 Service running on each of your HyperStore nodes requires outbound internet access. These features do not support configuring an explicit forward proxy, but you can use transparent forward proxying if you wish.
- **Pre-Configured ntpd** — Accurate, synchronized time across the cluster is vital to HyperStore service. In of your HyperStore data centers four of your HyperStore nodes are automatically configured to act as internal NTP servers. (If a HyperStore data center has only four or fewer nodes, then all the nodes in the data center are configured as internal NTP servers.) These internal NTP servers are configured to connect to external NTP servers — by default the public servers from the *pool.ntp.org* project. In order to connect to the external NTP servers, the internal NTP servers must be allowed outbound internet access. This feature does not support configuring an explicit forward proxy, but you can use transparent forward proxying if you wish.

  To see which of your HyperStore nodes are internal NTP servers, after HyperStore installation log into the CMC and go to **Cluster** → **Cluster Config** → **Cluster Information**.

  For more information on HyperStore's NTP set-up, see "System Configuration" -> "Configuration Special Topics" -> "NTP Automatic Set-Up" in the *HyperStore Administrator's Guide*.

5.3.1. Multi-DC Considerations

If you are installing HyperStore across multiple data centers and/or multiple service regions, the HyperStore nodes in each data center and region will need to be able to communicate with the HyperStore nodes in the other data centers and regions. This includes services that listen on the internal interface (such as Cassandra, the HyperStore Service, and Redis). Therefore you will need to configure your networking so that the internal networks in each data center and region are connected to each other (for example, by using a VPN). See **"HyperStore Listening Ports"** (page 25) for HyperStore requirements regarding listening port access.

5.4. File System Requirements

Subjects covered in this section:

- *Introduction (immediately below)*
- ***"OS/Metadata Drives and ext4-Formatted Data Drives"** (page 30)*
- ***"Mount Point Naming Guidelines"** (page 30)*
- ***"Option for Putting Cassandra Data on Dedicated Disks Rather Than the OS Disk"** (page 31)*
- ***"You Must Use UUIDs in fstab"** (page 31)*
- ***"You Must Create a Data Directory Mount Point List (fslist.txt)"** (page 32)*
- ***"Reducing Reserved Space to 0% for HyperStore Data Disks"** (page 33)*

Cloudian recommends that you use the HyperStore *system_setup.sh* tool to configure the data disks and mount points on your HyperStore nodes, as described in **"Configuring Network Interfaces, Time Zone, and Data Disks"** (page 15). The tool is part of the HyperStore product package (when you extract the *.bin* file). **If you do not use the system setup tool for disk setup, use the information below to make sure that your hosts meet HyperStore file system requirements**.

5.4.1. OS/Metadata Drives and *ext4*-Formatted Data Drives

Although it's possible to install HyperStore on a host with just a single hard drive, for a rigorous evaluation or for production environments each host should have multiple drives (see **"Host Hardware and OS Requirements"** (page 9)). On host machines with multiple hard drives:

- HyperStore will by default use the drive that the OS is on for storing system metadata (in Cassandra and Redis databases). If a host machine has 10 or more drives in total, Cloudian recommends that you dedicate two drives to the OS (and system metadata) in a RAID-1 mirroring configuration. Preferably the OS/metadata drives should be SSDs.
- **You must format all other available hard drives with *ext4* file systems mounted on raw disks**. These drives will be used for storing S3 object data. RAID is not necessary on the S3 object data drives.

For example, on a machine with 2 SSDs and 12 HDDs, mirror the OS on the two SSDs. Format each of the 12 HDDs with *ext4* file systems and configure mount points such as */cloudian1*, */cloudian2*, */cloudian3* and so on.

HyperStore **does not support** XFS file systems; VirtIO disks; Logical Volume Manager (LVM); or Multipathing. For questions regarding these unsupported technologies, contact Cloudian Support:

5.4.2. Mount Point Naming Guidelines

If you are installing HyperStore on multiple hosts that each have multiple disk drives, use the same mount point naming scheme on each of your hosts. If all your hosts have the same number of disks, then they should all have the identical set of mount points for HyperStore. For example, if each host has 12 disks for S3 object storage, then on all your hosts you could name the mount points */cloudian1*, */cloudian2*, */cloudian3*, and so on up through */cloudian12*.

If in your installation cluster some hosts have more disks than others, use as much overlap in mount point naming as possible. For example, suppose that most of your hosts have 10 disks for storing S3 object data while one host has 12 disks. In this scenario, all of the hosts can have mount points */cloudian1*, */cloudian2*, */cloudian3*, and so on up through */cloudian10*, while the one larger host has those same mount points plus also */cloudian11* and */cloudian12*.

Note Although uniformity of mount point naming across nodes (to the extent possible) is desirable for simplicity's sake, the HyperStore installation does support a way to accommodate differences in the number or names mount points across nodes -- this is described in **"You Must Create a Data Directory Mount Point List (fslist.txt)"** (page 32)..

5.4.3. Option for Putting Cassandra Data on Dedicated Disks Rather Than the OS Disk

Regarding Cassandra data, another supported configuration — for a host with many drives — is to put your Cassandra data directory and Cassandra commit log directories each on dedicated disks, rather than on the OS disk. In this case you would have:

- OS drive (with Redis also)
- Cassandra data directory drive (mount point path **must** include */cassandra*)
- Cassandra commit log directory drive (mount point path **must** include */cassandra_commit*)
- Multiple drives for S3 object data (with mount points for example */cloudian1*, */cloudian2*, */cloudian3* and so on).

In this configuration, where Cassandra data is on a different disk than the OS, it’s advisable to use RAID-1 for the OS disk. It’s not necessary to use RAID for a dedicated Cassandra disk.

5.4.4. You Must Use UUIDs in fstab

In your *fstab* file, **you must use UUIDs** to identify the devices to which you will mount HyperStore S3 object data directories. Do not use device names or LABELs.

If you are not using UUIDs in *fstab* currently, follow the instructions below to modify your *fstab* so that it uses UUIDs for the devices to which you will mount S3 object data directories (you do not need to do this for the OS/-metadata mount points).

As *root*, do the following:

1. Check whether your *fstab* is currently using UUIDs for your S3 object data drives. In the example below, there are two S3 object data drives and they are currently identified by device name, not by UUID.

```
[root@hyperstore1 etc]# cat /etc/fstab
...
...
/dev/sdb1 /cloudian1  ext4  rw,noatime,barrier=0,data=ordered,errors=remount-ro
0 1
/dev/sdc1 /cloudian2  ext4  rw,noatime,barrier=0,data=ordered,errors=remount-ro
0 1
```

2. Back up your existing fstab file:

```
[root]# cp /etc/fstab /etc/fstab.backup.<today's date>
```

3. Retrieve the UUIDs for your devices by using the *blkid* command.

```
[root]# blkid
...
...
/dev/sdb1: UUID="a6fed29c-97a0-4636-afa9-9ba23e1319b4" TYPE="ext4"
/dev/sdc1: UUID="rP38Ux-3wzO-sP3Y-2CoD-2TDU-fjpO-ffPFZV" TYPE="ext4"
```

4. Open *fstab* in an editor.
5. For each device that you are using for S3 object storage, replace the device name with *UUID="<UUID>"*, copying the device’s UUID from the *blkid* response in the previous step. For example:

```
# Original line

/dev/sdb1 /cloudian1 ext4 rw,noatime,barrier=0,data=ordered,errors=remount-ro  0
1

# Revised line

UUID="a6fed29c-97a0-4636-afa9-9ba23e1319b4" /cloudian1  ext4  rw,noatime,bar-
rier=0,
data=ordered,errors=remount-ro   0 1
```

6. After editing *fstab* so that each device on which you will store S3 data is identified by a UUID, save your changes and close the *fstab* file.
7. Remount the host’s file systems:

```
[root]# mount -a
```

Repeat this process for **each host on which you will install HyperStore**.

5.4.5. You Must Create a Data Directory Mount Point List (fslist.txt)

If you do not use the HyperStore *system_setup.sh* script to configure the data disks and mount points on your nodes, you must manually create a data directory mount point list file and place it in your installation staging directory on the Puppet Master node. If all your nodes have the same data mount points -- for example if all nodes have as their data mount points */cloudian1*, */cloudian2*, and so on through */cloudian12* -- you only need to create one mount point list file. If some nodes have a different set of mount points than do other nodes -- for example if some nodes have more data disks than other nodes -- you will need to create a default mount point list file and also a node-specific mount point list file for each node that differs from the default.

> **Note** If you use the *system_setup.sh* script to configure the disks and mount points on your nodes, the script creates the needed mount point list files automatically.

In your installation staging directory create a file named *fslist.txt* and in the file enter one line for each of your S3 data directory mount points, with each line using the format below.

```
<deviceName> <mountPoint>
```

Example of a properly formatted file (truncated):

```
/dev/sdc1 /cloudian1
/dev/sdd1 /cloudian2
...
```

Note Use device names in your *fslist.txt* file, not UUIDs.

Optionally, you can also specify one mount point for metadata stored in Cassandra (must include the string "cassandra" in the mount point path) and/or one mount point for the Cassandra commit log (must include "cassandra_commit" in the mount point path). If you do not specify these Cassandra mount points in *fslist.txt*, by default the system automatically puts Cassandra data and commit log directories on the same disk on which the operating system and application files reside.

Do not use symbolic links when specifying your mount points. The HyperStore system does not support symbolic links for data directories.

If some of your hosts have data directory mount point lists that differ from the cluster default, in the installation staging directory create a *<hostname>_fslist.txt* file for each such host. For example, along with the default *fslist.txt* file that specifies the mount points that most of your hosts use, you could also have a *cloudian-node11_fslist.txt* file and a *cloudian-node12_fslist.txt* file that specify mount points for two non-standard nodes that have hostnames *cloudian-node11* and *cloudian-node12*.

5.4.6. Reducing Reserved Space to 0% for HyperStore Data Disks

By default Linux systems reserve 5% of file system space for root user and system services. On modern large-capacity disks this can be a waste of a considerable amount of storage space. Cloudian recommends that you set the reserved space to 0% for each drive on which you will store HyperStore data (S3 object data).

For each HyperStore data drive do the following.

```
# Check current “Reserved block count”:

[root]# tune2fs -l <device>

# Set Reserved block count to 0%:

[root]# tune2fs -m 0 <device>

# For example:

[root]# tune2fs -m 0 /dev/sdc1
```

5.5. Cluster Survey File (survey.csv)

During the **"Installing HyperStore Prerequisites"** (page 12) task you use the *system_setup.sh* script to create a cluster survey file which by default is named *survey.csv*. This file resides in your installation staging directory for the life of your HyperStore system. The survey file is automatically updated by the system if you subsequently use the CMC to add more nodes to your cluster; and it is automatically copied to your new installation staging directory when you execute a HyperStore version upgrade.

> **Note** The survey file must be kept in the installation staging directory, not in a different directory. Do not delete or move the survey file.

The survey file contains one line for each HyperStore host in your cluster (including the Puppet Master host), with each line using the format below.

```
<regionname>,<hostname>,<ip4-address>,<datacenter-name>,<rack-name>[,<internal-interface>]
```

- *<region-name>* — The HyperStore system supports having multiple service regions with each region having its own independent storage cluster and own independent S3 object inventory, and with S3 clients able to choose a storage region when they create storage buckets. Even if you will have only one region you must give it a name. The maximum allowed length is 52 characters. The only allowed character types are lower case ASCII alphanumerical characters and dashes (a-z0-9 and dashes). **Make sure the region name matches the region string that you use in your S3 endpoints in your "DNS Set-Up"** (page 4). For more information on regions see "Service Regions Feature Overview" in the "Major Features" section of the *HyperStore Administrator's Guide*.
- *<hostname>* — Hostname of a host machine on which you are installing HyperStore software. Be sure to include in your survey file one line for each node on which you are installing HyperStore. You can install to all your intended nodes at the same time, even nodes in a different data center or region. In the hostnames you can use periods (such as in FQDNs), dashes, and underscores -- but not spaces or special characters.
- *<ip4-address>* — IP address (v4) that the hostname resolves to. Do not use IPv6. This should be the IP address associated with the host's default, external interface -- not an internal interface.
- *<datacenter-name>* — Name of the data center in which the host machine is located. The maximum allowed length is 256 characters. The only allowed character types are ASCII alphanumerical characters and dashes (A-Za-z0-9 and dashes).
- *<rack-name>* — Name of the server rack in which the host machine is located. The maximum allowed length is 256 characters. The only allowed character types are ASCII alphanumerical characters and dashes (A-Za-z0-9 and dashes).

  > **Note:** Within a data center, use the same "rack name" for all of the nodes, even if some nodes are on different physical racks than others. For example, if you have just one data center, all the nodes must use the same rack name. And if you have two data centers named DC1 and DC2, all the nodes in DC1 must use the same rack name as the other nodes in DC1; and all the nodes in DC2 must use the same rack name as the other nodes in DC2.

- *[<internal-interface>]* — Use this field only for hosts that will use a different network interface for internal cluster traffic than the rest of the hosts in the cluster do. For example, if most of your hosts will use "eth1" for internal cluster traffic, but two of your hosts will use "eth2" instead, use this field to specify "eth2" for each of those two hosts, and leave this field empty for the rest of the hosts in your survey file. (Later in the installation procedure you will have the opportunity to specify the default internal interface for the hosts in your cluster -- the internal interface used by all hosts for which you do not specify the *internal-interface* field in your survey file.) If all of your hosts use the same internal network interface — for example if all hosts use "eth1" for internal network traffic — then leave this field empty for all hosts in the survey file.

Note: Cassandra, Redis, and the HyperStore Service are among the services that will utilize the internal interface for intra-cluster communications.

The example survey file below is for a single-node HyperStore installation:

```
region1,arcturus,65.10.2.1,DC1,RAC1
```

This second example survey file is for a three-node HyperStore cluster with just one service region, one data center, and one rack:

```
tokyo,cloudian-vm7,65.10.1.33,DC1,RAC1
tokyo,cloudian-vm8,65.10.1.34,DC1,RAC1
tokyo,cloudian-vm9,65.10.1.35,DC1,RAC1
```

This third example survey file below is for a HyperStore installation that spans two regions, with the first region comprising two data centers and the second region comprising just one data center. Two of the hosts use a different network interface for internal network traffic than all the other hosts do.

```
boston,hyperstore1,65.1.0.1,DC1,RAC1
boston,hyperstore2,65.1.0.2,DC1,RAC1
boston,hyperstore3,65.1.0.3,DC1,RAC1
boston,hyperstore4,66.2.0.1,DC2,RAC1
boston,hyperstore5,66.2.0.2,DC2,RAC1
chicago,hyperstore6,68.3.0.1,DC3,RAC1
chicago,hyperstore7,68.3.0.2,DC3,RAC1
chicago,hyperstore8,68.3.2.1,DC3,RAC1,eth2
chicago,hyperstore9,68.3.2.2,DC3,RAC1,eth2
```

5.6. cloudianInstall.sh Command Line Options

The HyperStore installation script *cloudianInstall.sh* resides in your installation staging directory on your Puppet Master node. Typically you would launch the script either like this:

```
[7.2]# ./cloudianInstall.sh -s survey.csv
```

Or like this if you are not using your DNS environment to resolve HyperStore service endpoints and you want to use the bundled tool dnsmasq instead (which is not appropriate for production systems):

```
[7.2]# ./cloudianInstall.sh -s survey.csv configure-dnsmasq
```

However the script does support additional command line options. The syntax is as follows:

```
[7.2]# ./cloudianInstall.sh [-s <survey-filename>]
[-k <ssh-private-key-filename>] [-d] [-h] [no-hosts] [configure-dnsmasq]
[no-firewall] [force]
```

Note If you use multiple options, on the command line place options that start with a "-" (such as *-s <survey-filename>* or *-d*) before options that do not (such as *no-hosts* or *configure-dnsmasq*).

After using the installer for product installation or subsequently for system configuration tasks, exit the installer when you’re done. Do not leave it running. Certain automated system tasks invoke the installer and cannot do so if it is already running.

- *[-s <survey-filename>]* — Name of your cluster survey file. If you do not specify the survey filename argument, the script will prompt you for the file name during installation.
- *[-k <ssh-private-key-filename>]* — The Puppet master employs SSH for secure communication with the rest of your HyperStore installation nodes. Use the *-k <ssh-private-key-filename>* option if you want to use your own existing SSH authentication key pair rather than having the HyperStore install tool generate a key pair for you. If you want to use this option, then before running the *cloudianInstall.sh* script:
  - Copy your private key into the installation staging directory (where the install script resides).
  - Copy your public key to each host on which you plan to install HyperStore (see standard SSH set-up documentation for guidance); or copy your public key to the installation staging directory and the install script will copy the public key to your target nodes automatically.
- *[-d]* — Turn on debugging output.
- *[-h]* — Display usage information for the install tool. This option causes the tool to print a usage message and exit.

> **Note:** This usage information mentions more command line options than are described here in this Help topic. This is because the usage information includes installer options that are meant for HyperStore internal system use, such as options that are invoked by the CMC when you use the CMC to add nodes to your cluster or remove nodes from your cluster. You should perform such operations through the CMC, not directly through the installer. The CMC implements automations and sanity checks beyond what is provided by the install script alone.

- *[no-hosts]* — Use this option if you do not want the install tool to append entries for each HyperStore host on to the */etc/hosts* file of each of the other HyperStore hosts. By default the tool appends to these files so that each host is resolvable to the other hosts by way of the */etc/hosts* files.
- *[configure-dnsmasq]* — Use this option if you want the install tool to install and configure **dnsmasq**, a lightweight utility that can provide domain resolution services for testing a small HyperStore system. If you use this option the installer installs *dnsmasq* and automatically configures it for resolution of HyperStore service domains. If you did not create DNS entries for HyperStore service domains as described in **"DNS Set-Up"** (page 4), then you must use the *configure-dnsmasq* option in order for the system to be functional when you complete installation. Note that using *dnsmasq* is not appropriate in a production environment.

> **Note:** If you do not have the installer install *dnsmasq* during HyperStore installation, and then later you decide that you do want to use *dnsmasq* for your already installed and running HyperStore system, do not use the *configure-dnsmasq* command line option when you re-launch the installer. Instead, re-launch the installer with no options and use the Installer Advanced Configuration Options menu to enable *dnsmasq* for your system. (See the System Configuration chapter of the HyperStore Administrator's Guide for instructions.)

- [*no-firewall*] — If this option is used, the HyperStore firewall will **not** be enabled upon HyperStore installation. By default the HyperStore firewall will be enabled upon completion of a fresh HyperStore installation. For more information about the HyperStore firewall see "Configuring the HyperStore Firewall" in the System Configuration section of the *HyperStore Administrator's Guide*.
- *[force]* — By default the installer performs certain prerequisite checks on each node on which you are installing HyperStore and aborts the installation if any of your nodes fails a check. By contrast, if you use the *force* option when you launch the installer, the installer will output warning messages to the terminal if one or more nodes fails a prerequisite check but the installation will continue rather than

aborting. The prerequisite checks that this feature applies to are:

- CPU has minimum of 8 cores
- RAM is at least 16GB
- System Architecture is x86 64-bit
- SELinux is disabled
- firewalld is disabled
- iptables is not running
- Directory permissions are as needed
- Hostnames of installation hosts are resolvable or installer is allowed to append to hosts' */etc/hosts* files
- S3 service and Admin service endpoints (which you will specify during the interactive install) are resolvable -- which they should be so long as you either create DNS entries for HyperStore service endpoints or else use the *configure-dnsmasq* option when launching the installer

5.7. system_setup.sh Command Line Options

For basic information about *system_setup.sh* command line options, change to the directory in which the script is located (typically your installation staging directory) and run the following command:

```
[7.2]# ./system_setup.sh --help
```