Network Setup Guide:
Deploying your Cloudian Hyperstore Hybrid Storage Service

Contents

Intended Audience 2
DNS, Networking and Firewall connections 2
Networking Preparation 3
Hostnames and IP Addresses 3
Network Interfaces and Listening Ports 4
DNS Set-Up 6
Outbound Internet Access 9

Intended Audience

The purpose of this document is to help a new user deploy a 3-node Cloudian storage cluster in your datacenter for use with the Cloudian HyperStore Hybrid Cloud Service from AWS Marketplace.

DNS, Networking and Firewall connections

Cloudian HyperStore is an IP accessible service and careful consideration must be paid to your network configuration. It is important that all services and ports are accessible across your network infrastructure (DNS, Firewalls etc). This document provides a guide to confirm service connections and port access.

Networking Preparation

This section on networking requirements for HyperStore covers the following topics:

- Hostnames and IP Addresses
- Network Interfaces and Listening Ports
- DNS Set-Up
- Outbound Internet Access

Hostnames and IP Addresses

For all hosts on which you will install HyperStore software, make sure that the hostname:

- Does not contain upper case letters. Must be lower case only.
- Is not set to *localhost*.

  To confirm, at the Linux prompt type *hostname* and verify that the command does not return *localhost*. In the example below, the hostname is *cloudian-machine*.

  ```
  root# hostname
  cloudian-machine
  ```

- Is not mapped to the loopback address (127.0.0.1) in */etc/hosts*.

  To confirm, check the contents of */etc/hosts*. In the example below, the hostname *cloudian-machine* is mapped to IP address 10.0.1.20, not to the loopback address.

  ```
  root# cat /etc/hosts
  127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
  ::1          localhost localhost.localdomain localhost6 localhost6.localdomain6
  10.0.1.20    cloudian-machine
  ...
  ```

- Is assigned a static IPv4 address. Do not use DHCP. Do not use IPv6.

  To confirm that a host is not using DHCP, check the contents of */etc/sysconfig/network-scripts/ifcfg-eth0* and verify that *BOOTPROTO=none*.

  ```
  root# cat /etc/sysconfig/network-scripts/ifcfg-eth0
  DEVICE=eth0
  BOOTPROTO=none
  ...
  ```

  If you are using a network interface device other than *eth0*, check the *ifcfg-<interface>* file for the interface that you're using to confirm that *BOOTPROTO=none*.

Network Interfaces and Listening Ports

When you are installing HyperStore you will specify the name of the dedicated internal network interface used by each of your host machines (if your host machines have multiple NICs and you want to designate an interface for internal cluster traffic). The system supports either of two configuration scenarios for internal service network interface:

- Default configuration: All of your HyperStore hosts use the same network interface for internal network traffic (for example all hosts use "eth1" for internal network traffic)

  OR

- Different hosts use different interfaces for internal network traffic (for example most of your hosts use "eth1" for internal network traffic while other hosts use "eth2" for internal network traffic).

HyperStore does not support using interfaces that have a period in their name, such as "bond1.1234". If you have a period in the names of any of interfaces that HyperStore will use, please rename them. Underscores are allowed, such as "bond1_1234".

HyperStore does not support specifying interfaces on a per-service level. For example you cannot configure your system so that Redis listens on "bond1" while Cassandra listens on "bond2" and the HyperStore Service listens on "bond3".

The following services will all listen on the dedicated internal network interface for intra-cluster communications:

- Redis Credentials Service and Redis QoS Service
- Redis Monitor
- Cassandra Service
- HyperStore Service

From the table below the only services that should be exposed to the external network are:

- S3 Service
- Cloudian Management Console (CMC), if you want to allow regular users to access the CMC. The web-based CMC includes a GUI for interfacing with the data store (for example uploading or downloading objects). When regular users access the CMC they see only the CMC's end user oriented functions.

Do not publicly expose any of the other services in the table below.

If you use any type of firewall — on the nodes in your HyperStore cluster, make sure that there are **no restrictions on internal communication between nodes**. HyperStore nodes sometimes communicate with each other via JMX, and when they do, after initial connection establishment on the designated JMX port (see detail in the table below) a random port is used for continued communication. Therefore, there cannot be any port restrictions on internal communication between HyperStore nodes. **Only external communication should be restricted**.

Service	Listening Port	Purpose
S3 Service	80	Requests from S3 client applications via HTTP
	443	Requests from S3 client applications via HTTPS
	19080	JMX access
Cloudian Management Console (CMC)	8888	Requests from administrators' or end users' browsers via HTTP
	8443	Requests from administrators' or end users' browsers via HTTPS
Admin Service	18081	Admin API requests from the CMC via HTTP
	19443	Admin API requests from the CMC via HTTPS
	19081	JMX access
Redis Monitor	9078	Communication between primary and backup Redis Monitor instances
	19083	JMX access
HyperStore Service	19090	Data operation requests from the S3 Service
	19050	Communication between HyperStore Service instances
	19082	JMX access
Redis DBs	6379	Requests to the Redis Credentials DB from the S3, HyperStore, or Admin Services; and communication between Redis Credentials instances
	6380	Requests to the Redis QoS DB from the S3, HyperStore, or Admin Services; and communication between Redis Credentials instances
Cassandra	9160	Data operations requests from the S3, HyperStore, or Admin Services
	7000	Communication between Cassandra instances
	7199	JMX access
Cloudian Monitoring Agent	19070	Requests from the Cloudian Monitoring Data Collector
Puppet Master	8140	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

SSH	22	The HyperStore installer accesses this SSH port on each node on which you are installing HyperStore software

DNS Set-Up

For HyperStore to function properly, the HyperStore service endpoints (service URIs) must be resolvable. You have two options for making HyperStore service endpoints resolvable:

- The HyperStore product package includes an open source lightweight domain resolution utility called dnsmasq. When you launch the HyperStore installation script (as described later in this document), you can optionally have the script install *dnsmasq* and automatically configure it to resolve all HyperStore service domains. If you use this option, then no further domain resolution set-up is necessary. **This option is not appropriate for production environments**. However, it may be convenient if you're only installing one or a few HyperStore nodes in order to do some simple testing and initial evaluation of the system.

  OR

- On your name servers, configure DNS records for the HyperStore service endpoints. This is the recommended method for production environments or a rigorous evaluation.

The table below shows the DNS entries that you must configure on your name servers, to resolve HyperStore service endpoints. By default, the HyperStore system derives the endpoint values from your organization's top level 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 S3 or CMC endpoint formats, the HyperStore system allows you to specify custom endpoint values during the installation. If you intend to create custom endpoints rather than accepting the default endpoints, configure DNS entries to resolve those custom endpoint values rather than the default-formatted endpoint values shown below. Make a note of the custom endpoints for which you have configured DNS entries, so that later you can correctly specify those endpoints when you perform the HyperStore interactive installation.

DNS Entry	Default Format and Example	Description
S3 service endpoint (one per service region)	*s3-<region>.<your-domain>* *s3-tokyo.enterprise.com*	This is the service endpoint to which S3 client applications will submit requests. The *<region>* segment indicates the HyperStore service region. You must choose a service region name for your HyperStore installation, even if you intend to have only one service region. The region name must be lower case with no dots, dashes, underscores, or spaces. You will supply this region name again when you perform the HyperStore installation — make sure that the region name you supply when doing the install matches the region name you used in your DNS configuration. 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*. For more information about service regions see Service Regions in the *Cloudian HyperStore Administrator's Guide*. If you want to use a **custom S3 endpoint** that does not include a region string, HyperStore allows you to do so. Note however that if you have a multi-region system, using S3 endpoints that lack region strings means that you will miss out on some of the benefits of AWS Signature Version 4 authentication for S3 requests (specifically, the region validation aspect).
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 a specific storage bucket (which is nearly all S3 requests). By default S3 clients (including the Cloudian Management Console) submit S3 requests that include a bucket name in the HTTP Host header, in the form *<bucketname>.<s3-service-endpoint>*. For example *Host: bucket1.s3-tokyo.enterprise.com*.
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 whole system)	*s3-admin.<your-domain>* *s3-admin.enterprise.com*	This is the service endpoint for HyperStore's Admin API. The Cloudian Management Console accesses this RESTful HTTP API, and you can also access the API directly with a command line tool such as *cURL* or a client application of your own creation. Unlike the S3 and CMC endpoints, the installer does not support customizing the Admin Service endpoint. Configure a DNS entry for the default-formatted Admin Service endpoint (*s3-admin.<your-domain>*).
Cloudian Management Console (CMC) domain (one per whole 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) or performing system provisioning and administration tasks.

Below is an example set of DNS entries for HyperStore services, including the needed wildcard entries. Typically one or more virtual IP addresses (VIPs) would be used here, with the VIPs being the addresses of load balancers which would in turn distribute traffic among the HyperStore nodes.

```
s3-tokyo.enterprise.com IN A 123.123.123.123
*.s3-tokyo.enterprise.com IN A  123.123.123.123
s3-website-tokyo.enterprise.com IN A 123.123.123.123
*.s3-website-tokyo.enterprise.com IN A 123.123.123.123
s3-admin.enterprise.com IN A 123.123.123.124
cmc.enterprise.com IN A 123.123.123.125
```

The example above is for a single-region HyperStore system. If you are deploying a multi-region HyperStore system, each region will have its own S3 service endpoint and S3 static website endpoint, and DNS entries are necessary for each region (including the wildcard entries for resolving bucket-specific S3 requests).

For some configuration notes regarding setting up a load balancer to work with Hyperstore, see Load Balancing in the *Cloudian HyperStore Administrator's Guide*.

Using load balancers and VIPs is the recommended method for balancing request load across multiple HyperStore nodes. Alternatively you can use round-robin DNS to balance load. Note though that **with round-robin DNS, if a node goes down, some requests will still get routed to that node and those requests will fail**.

Here is an example of a round-robin DNS configuration for a three-node HyperStore system:

s3-tokyo.enterprise.com IN A	123.123.456.787
	123.123.456.788
	123.123.456.789
*.s3-tokyo.enterprise.com IN A	123.123.456.787
	123.123.456.788
	123.123.456.789
s3-website-tokyo.enterprise.com IN A	123.123.456.787
	123.123.456.788
	123.123.456.789
*.s3-website-tokyo.enterprise.com IN A	123.123.456.787
	123.123.456.788
	123.123.456.789
s3-admin.enterprise.com IN A	123.123.456.787
	123.123.456.788
	123.123.456.789
cmc.enterprise.com IN A	123.123.456.787
	123.123.456.788
	123.123.456.789

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:

- **Phone Home** — The Phone Home feature (also known as "Smart Support") 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 are *mts.properties.erb: phonehome.proxy.**).
- **Auto-Tiering** — In support of HyperStore's auto-tiering feature — for automatically transferring locally stored objects to Amazon S3 on a pre-defined schedule — the S3 Service running on each of your HyperStore nodes requires outbound internet access (if you want to use the auto-tiering feature). This feature doesn't support configuring an explicit forward proxy, but you can use transparent proxying if you wish.
- **Pre-Configured *ntpd*** — Accurate, synchronized time across the cluster is vital to HyperStore service. Two of your HyperStore nodes are automatically configured to act as local NTP servers (with one configured as the primary and the other as the secondary). All the other HyperStore nodes are automatically configured as clients to the two local NTP servers. The two nodes that act as local NTP servers are configured to connect to external NTP servers — the default public servers from the *pool.ntp.org* project. In order to connect to the public NTP servers at *pool.ntp.org*, the two local NTP servers must be allowed outbound internet access. This feature doesn't support configuring an explicit forward proxy, but you can use transparent proxying if you wish.

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