## Contents

1. About this Guide ................................................................................................................................. 4
2. Loadbalancer.org Appliances Supported........................................................................................... 4
3. Loadbalancer.org Software Versions Supported................................................................................. 4
4. Cloudian HyperStore Software Versions Supported............................................................................ 4
5. Cloudian HyperStore ............................................................................................................................ 5
6. Load Balancing Cloudian HyperStore................................................................................................ 5
   - Load Balancing & HA Requirements ............................................................................................... 5
   - Persistence (aka Server Affinity) ..................................................................................................... 5
   - Virtual Service (VIP) Requirements ............................................................................................... 5
   - Port Requirements ............................................................................................................................ 5
   - Health Checks .................................................................................................................................... 6
   - GSLB / Location Affinity .................................................................................................................... 6
   - Alternative Load Balancing Method for Read-Intensive Deployments (Direct Routing).............. 6
7. Performance and Sizing for a Virtual Load Balancer Deployment with Cloudian HyperStore .......... 6
8. Deployment Concept ............................................................................................................................. 7
9. Loadbalancer.org Appliance – the Basics ........................................................................................... 8
   - Virtual Appliance Download & Deployment.................................................................................... 8
   - Initial Network Configuration ........................................................................................................... 8
   - Accessing the Web User Interface (WebUI) ....................................................................................... 9
   - HA Clustered Pair Configuration ...................................................................................................... 10
10. Appliance Configuration for Cloudian HyperStore ........................................................................ 11
    - Enabling Multithreaded Load Balancing ...................................................................................... 11
    - Configuring VIP 1 – Cloudian Management Console ................................................................. 11
      - Configuring the Virtual Service (VIP) ......................................................................................... 11
      - Defining the Real Servers (RIPs) ................................................................................................. 13
    - Configuring VIP 2 – S3 Client Requests (HTTP) .......................................................................... 13
      - Configuring the Virtual Service (VIP) ......................................................................................... 13
    - Configuring VIP 3 – S3 Client Requests (HTTPS) ........................................................................ 15
      - Configuring the Virtual Service (VIP) ......................................................................................... 15
    - Configuring VIP 4 – Admin API Requests ..................................................................................... 16
      - Configuring the Virtual Service (VIP) ......................................................................................... 16
    - Finalizing the Configuration ........................................................................................................... 17
11. Testing & Verification .......................................................................................................................... 17
    - Using System Overview ................................................................................................................ 17
12. Technical Support ............................................................................................................................... 18
13. Further Documentation ....................................................................................................................... 18
14. Conclusion ........................................................................................................................................... 18
15. Appendix ............................................................................................................................................ 19
    - 1 – Configuring GSLB / Location Affinity .................................................................................... 19
      - Conceptual Overview ................................................................................................................. 19
      - Appliance Configuration .............................................................................................................. 20
      - DNS Server Configuration .......................................................................................................... 22
    - 2 – Alternative Load Balancing Method for Read-Intensive Deployments (Direct Routing) .... 23
Caveats .................................................................................................................................................. 23
Appliance Configuration for Cloudian HyperStore – Using Layer 4 DR Mode (Direct Routing) .................. 24
3 – Using the PROXY Protocol to Retain Client IP Addresses ................................................................. 26
Configuring the HyperStore Nodes to Accept the PROXY Protocol .......................................................... 26
Appliance Configuration .......................................................................................................................... 28
4 – Clustered Pair Configuration – Adding a Slave Unit ............................................................................ 30
16. Document Revision History .................................................................................................................. 32
1. About this Guide
This guide details the steps required to configure a load balanced Cloudian HyperStore environment utilizing Loadbalancer.org appliances. It covers the configuration of the load balancers and also any Cloudian HyperStore configuration changes that are required to enable load balancing.

For more information about initial appliance deployment, network configuration and using the Web User Interface (WebUI), please also refer to the relevant Administration Manual:

- v7 Administration Manual
- v8 Administration Manual

2. Loadbalancer.org Appliances Supported
The complete list of our products that are supported for load balancing Cloudian HyperStore is shown below:

<table>
<thead>
<tr>
<th>Discontinued Models</th>
<th>Current Models *</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Enterprise 10G</td>
</tr>
<tr>
<td></td>
<td>Enterprise 40G</td>
</tr>
<tr>
<td></td>
<td>Enterprise VA MAX</td>
</tr>
</tbody>
</table>

* For full specifications of these models please refer to: http://www.loadbalancer.org/products/hardware

** Some features may not be supported, please check with Loadbalancer.org support

3. Loadbalancer.org Software Versions Supported

- V8.4.1 and later

4. Cloudian HyperStore Software Versions Supported

- Cloudian HyperStore – all versions
5. Cloudian HyperStore

Cloudian is a file and object storage company specialising in S3 (Simple Storage Service) API storage systems. The technology allows companies of all sizes realise the benefits of object storage in their own data centres. The Cloudian HyperStore Operating Environment software provides scalable enterprise object storage, with 100% native Amazon S3-API support.

Cloudian HyperStore architecture supports High Availability (HA) clustering by putting a load balancer in front of it. Load balancers monitor and perform health checks on a node to ensure traffic is routed correctly to healthy nodes. Without the use of a load balancer, an offline or failed node would still receive traffic, causing failures.

A variety of load balancing methods are currently supported by Cloudian HyperStore, dependent on customer infrastructure, including layer 4, layer 7 and Geo GSLB/location affinity. The HyperStore services that should be balanced are: S3, Cloudian Management Console (CMC) and Admin-API.

6. Load Balancing Cloudian HyperStore

Note: It’s highly recommended that you have a working Cloudian HyperStore environment first before implementing the load balancer.

Load Balancing & HA Requirements

The function of the load balancer is to distribute inbound connections across a cluster of Cloudian HyperStore nodes, to provide a highly available and scalable service. Four virtual services are used to load balance the different aspects of HyperStore.

Persistence (aka Server Affinity)

The CMC VIP requires source IP persistence to be enabled in order to function correctly. For all other VIPs, client persistence is not required and should not be enabled.

Virtual Service (VIP) Requirements

To provide load balancing for Cloudian HyperStore, the following VIPs are required:

- **CMC**: for Cloudian Management Console requests
- **S3-HTTP**: handles requests from S3 client applications via HTTP
- **S3-HTTPS**: handles requests from S3 client applications via HTTPS
- **API**: handles API requests via HTTPS

Port Requirements

The following table shows the ports that are load balanced:

<table>
<thead>
<tr>
<th>Port</th>
<th>Protocols</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>TCP/HTTP</td>
<td>Requests from S3 client applications</td>
</tr>
</tbody>
</table>
Health Checks
The S3-HTTP service uses the "Negotiate HTTP (HEAD)" health check, while the S3-HTTPS and API services both use the "Negotiate HTTPS (HEAD)" health check.

The CMC service uses the "Negotiate HTTPS (OPTIONS)" health check.

The health check for the API virtual service should be configured with the credentials for the sysadmin user so that it can authenticate against the API service in order to successfully check its health. This is described fully in section Configuring VIP 4 – Admin API Requests.

GSLB / Location Affinity
For multi-site HyperStore deployments, it is possible to use the load balancer’s GSLB functionality to provide high availability and location affinity across multiple sites. Using this optional, DNS based feature, in the event that a site’s HyperStore service and/or load balancers are offline then local clients are automatically directed to a functioning HyperStore cluster at another site.

A full explanation and instructions on setting up this optional feature can be found in section 1 of the appendix, Configuring GSLB / Location Affinity, on page 19.

Alternative Load Balancing Method for Read-Intensive Deployments (Direct Routing)
For deployments that are read-intensive, it is possible to use an alternative load balancing method known as direct routing. This allows reply traffic to flow directly from the back end servers to the clients, thus removing the load balancer as a potential bottleneck for reply traffic. Direct routing can benefit read-intensive deployments with a large reply traffic to request traffic ratio.

A more detailed explanation of this alternative load balancing method can be found in section 2 of the appendix, Alternative Load Balancing Method for Read-Intensive Deployments (Direct Routing), on page 23.

7. Performance and Sizing for a Virtual Load Balancer Deployment with Cloudian HyperStore
The Loadbalancer.org appliance can be deployed as a virtual appliance.

To achieve the best level of performance and throughput when load balancing a Cloudian HyperStore deployment, the Loadbalancer.org appliance should be configured to actively use multiple CPU cores for the load balancing process. This must be considered when initially deploying and sizing virtual appliances.

A virtual host should be allocated a minimum of 4 vCPUs.
8. Deployment Concept

VIPs = Virtual IP Addresses

Note: The load balancer can be deployed as a single unit, although Loadbalancer.org recommends a clustered pair for resilience & high availability. Please refer to section 4 in the appendix on page 30 for more details on configuring a clustered pair.
9. Loadbalancer.org Appliance – the Basics

Virtual Appliance Download & Deployment
A fully featured, fully supported 30 day trial is available if you are conducting a PoC (Proof of Concept) deployment. The VA is currently available for VMware, Virtual Box, Hyper-V, KVM and XEN and has been optimized for each Hypervisor. By default, the VA is allocated 1 CPU, 2GB of RAM and has an 8GB virtual disk. The Virtual Appliance can be downloaded [here](#).

Note: The same download is used for the licensed product, the only difference is that a license key file (supplied by our sales team when the product is purchased) must be applied using the appliance’s WebUI.

Note: Please refer to the [Administration Manual](#) and the ReadMe.txt text file included in the VA download for more detailed information on deploying the VA using various Hypervisors.

Initial Network Configuration
The IP address, subnet mask, default gateway and DNS settings can be configured in several ways as detailed below:

**Method 1 - Using the Network Setup Wizard at the console**
After boot up, follow the instructions on the console to configure the IP address, subnet mask, default gateway and DNS settings.

**Method 2 - Using the WebUI**
Using a browser, connect to the WebUI on the default IP address/port: [https://192.168.2.21:9443](https://192.168.2.21:9443)
To set the IP address & subnet mask, use: `Local Configuration > Network Interface Configuration`
To set the default gateway, use: `Local Configuration > Routing`
To configure DNS settings, use: `Local Configuration > Hostname & DNS`
Accessing the Web User Interface (WebUI)
The WebUI can be accessed via HTTPS at the following URL: https://192.168.2.21:9443/lbadmin
* Note the port number → 9443

(replace 192.168.2.21 with the IP address of your load balancer if it’s been changed from the default)

Login using the following credentials:

  **Username:** loadbalancer  
  **Password:** loadbalancer

Note: To change the password, use the WebUI menu option: Maintenance > Passwords.

Once logged in, the WebUI will be displayed as shown on the following page:
HA Clustered Pair Configuration

Loadbalancer.org recommend that load balancer appliances are deployed in pairs for high availability. In this guide a single unit is deployed first, adding a secondary slave unit is covered in section 4 of the appendix on page 30.
10. Appliance Configuration for Cloudian HyperStore

**Enabling Multithreaded Load Balancing**

The Loadbalancer.org appliance should be configured to actively use multiple CPU cores for the load balancing process. This is required to achieve the high level of performance and throughput required when load balancing a Cloudian HyperStore deployment.

Note: A virtual host should be allocated a minimum of 4 vCPUs.

A **minimum of 4 threads** should be defined. The number of threads can be set as high as the number of threads available to the system (setting the value even higher than that will not increase performance).

To enable multithreaded mode from the WebUI:

1. Navigate to Cluster Configuration > Layer 7 - Advanced Configuration
2. Check the **Enable Multithreading** checkbox
3. Set **Number of Threads** to a minimum of 4
4. Click **Update** to apply the changes

**Configuring VIP 1 – Cloudian Management Console**

**Configuring The Virtual Service (VIP)**

1. Using the web user interface, navigate to Cluster Configuration > Layer 7 – Virtual Services and click on **Add a new Virtual Service**
2. Define the **Label** for the virtual service as required, e.g. `cmc.cloudian-hyperstore`
3. Set the **Virtual Service IP Address** field to the required IP address, e.g. `192.168.87.67`
4. Set the **Ports** field to `8888,8443`
5. Set the **Layer 7 Protocol** to **TCP Mode**
6. Click **Update** to create the virtual service
7. Click **Modify** next to the newly created VIP
8. Set **Persistence Mode** to **Source IP**
9. Set **Health Checks** to **Negotiate HTTPS (OPTIONS)**
10. Set **Request to send** to `/Cloudian/login.htm`
11. Click the **Advanced** button to expand the **Health Checks** menu
12. Set **Check Port** to **8443**

Note: The **Host Header** field should be set if appropriate, such as with your S3 endpoint name, for example ‘s3-region1.domain’.

13. Click **Update**
Defining The Real Servers (RIPs)

1. Using the web user interface, navigate to Cluster Configuration > Layer 7 – Real Servers and click on Add a new Real Server next to the newly created VIP.
2. Define the Label for the real server as required, e.g. cloudian-node1.
3. Set the Real Server IP Address field to the required IP address, e.g. 10.10.10.11.
4. Click Update.
5. Repeat these steps to add additional HyperStore nodes as real servers as required.

![Layer 7 Add a new Real Server - cmc.cloudian-hyperstore](image)

Configuring VIP 2 – S3 Client Requests (HTTP)

Configuring The Virtual Service (VIP)

1. Using the web user interface, navigate to Cluster Configuration > Layer 7 – Virtual Services and click Modify next to the previously created CMC VIP.
2. Click Duplicate Service and confirm when prompted.
3. Define the Label for the new virtual service as required, e.g. s3.cloudian-hyperstore.
4. Set the Virtual Service IP Address field to the required IP address, e.g. 192.168.87.67.
5. Set the Ports field to 80.
6. Set Persistence Mode to **None**
7. Set Health Checks to **Negotiate HTTP (HEAD)**
8. Set Request to send to **/healthCheck**
9. In the Health Checks section click **Advanced** to expand the menu
10. Clear the Check Port field to leave it empty

![Layer 7 - Modify Virtual Service]

**Health Checks**

- Health Checks: **Negotiate HTTP (HEAD)**
- Request to send: **/healthCheck**
- Check Port: **-**
- Username:
- Host Header:
- Password *

Note: The Host Header field should be set if appropriate, such as with your S3 endpoint name, for example 's3-region1.domain'.

11. Click **Update**

Note: If a HyperStore deployment requires the true source IP addresses of clients to be logged for S3 requests, for example so that S3 bucket policies or billing whitelisting can be used, then the PROXY protocol can be used to achieve this.
Configuring VIP 3 – S3 Client Requests (HTTPS)

Configuring The Virtual Service (VIP)

1. Using the web user interface, navigate to Cluster Configuration > Layer 7 – Virtual Services and click Modify next to the previously created S3 VIP
2. Click Duplicate Service and confirm when prompted

3. Define the Label for the new virtual service as required, e.g. https.s3.cloudian-hyperstore
4. Set the Virtual Service IP Address field to the required IP address, e.g. 192.168.87.67
5. Set the Ports field to 443

6. Set Health Checks to Negotiate HTTPS (HEAD)

Note: Clicking on Advanced reveals a Host Header field, which should be set if appropriate, such as with your S3 endpoint name, for example ‘s3-region1.domain’.
7. Click **Update**

Configuring VIP 4 – Admin API Requests

Configuring The Virtual Service (VIP)

1. Using the web user interface, navigate to **Cluster Configuration > Layer 7 – Virtual Services** and click **Modify** next to the previously created S3 HTTPS VIP
2. Click **Duplicate Service** and confirm when prompted

3. Define the **Label** for the new virtual service as required, e.g. `api.cloudflare-hyperstore`
4. Set the **Virtual Service IP Address** field to the required IP address, e.g. `192.168.87.67`
5. Set the **Ports** field to `19443`

6. In the **Health Checks** section click **Advanced** to expand the menu
7. Set **Username** to `sysadmin`
8. Set **Password** to `public`
9. Click Update

Finalizing the Configuration
To apply the new settings, HAProxy must be restarted as follows:

1. Using the WebUI, navigate to: Maintenance > Restart Services and click Reload HAProxy

11. Testing & Verification

Using System Overview

The System Overview can be viewed in the WebUI. It shows a graphical view of all VIPs & RIPv (i.e. the HyperStore Nodes) and shows the state/health of each server as well as the state of the each cluster as a whole. The example below shows that all HyperStore nodes are healthy and available to accept connections.
12. Technical Support
For more details about configuring the appliance and assistance with designing your deployment please don’t hesitate to contact the support team using the following email address: support@loadbalancer.org

13. Further Documentation

14. Conclusion
Loadbalancer.org appliances provide a very cost effective solution for highly available load balanced Cloudian HyperStore environments.
Appendix 1 – Configuring GSLB / Location Affinity

Conceptual Overview

For multi-site HyperStore deployments, it is possible to use the load balancer’s GSLB functionality to provide high availability and location affinity across multiple sites.

- Clients across multiple sites can use the same fully qualified domain name to access HyperStore services
- Under normal operation, clients are directed to their site’s local HyperStore cluster
- In the event that a site’s HyperStore service and/or load balancers are offline then local clients are automatically directed to a functioning HyperStore cluster at another site

For the sake of simplicity, the diagram presented below shows a two site setup. The principle can be extended to encompass as many sites as desired.
Explanation:

- **Start**: A client tries to access the S3 service by using the service’s fully qualified domain name, in this example s3-region1.domain.tld
- The client sends a DNS query for s3-region1.domain.tld to its local DNS server
- The local site's DNS server has the domain s3-region1.domain.tld delegated to the load balancers
- The DNS server sends a delegated DNS query for s3-region1.domain.tld to one of the load balancers
- The load balancer that received the delegated DNS query replies to the DNS server by serving up the appropriate, local VIP address. For example, if the query originated from the 10.0.0.0/24 subnet then the VIP in that subnet is served up. Likewise, if the query originated from the 172.16.0.0/24 subnet then the VIP in that subnet is served up. As such, clients are always directed to their local, on-site HyperStore instance, provided that the on-site instance is online and available
- The DNS server sends the delegated DNS answer to the client
- **Finish**: The client connects to the S3 service at s3-region1.domain.tld by using the local VIP address

Note: *In the event that the HyperStore cluster and/or load balancers at one site should completely fail* then local clients will be directed to the HyperStore cluster at the other site and the service will continue to be available.

This style of multi-site failover is possible because the load balancers’ GSLB functionality continuously health checks the service at each site. When the service at a site is observed to be unavailable then that site’s IP address is no longer served when responding to DNS queries.

Handling Multiple Subdomains

Cloudian HyperStore DNS configurations typically use the following DNS subdomains (or something similar):

- cmc.domain.tld
- s3-admin.domain.tld
- s3-<region/location>.domain.tld (e.g. s3-region1.domain.tld)

Due to the complexity of DNS delegation, it is recommend to delegate a single subdomain using the GSLB service and then using CNAME records to point everything else at the delegated subdomain. This makes it much easier to configure many DNS entries, including more complex options such as using wildcard entries, for example *s3-region1.domain.tld.

Appliance Configuration

The GSLB service must be configured on the master load balancer appliance at each site. The GSLB configuration must be identical across all sites. Configuration takes place in two locations in the WebUI:

- Polaris config: the main configuration for the Polaris service which handles GSLB
- Topology config: defines the network topology, mapping network subnets to sites

Polaris Config

Using the web user interface of the master appliance, navigate to Cluster Configuration > GSLB Configuration and select
Polaris Config from the drop-down list.

If GSLB has never been configured then the default example configuration will be displayed in the text box.

What follows is a Cloudian specific example configuration which covers three example sites. It can be copy and pasted to replace the default example and can be used as a basis for creating a deployment-specific configuration.

The elements presented in **boldface** should be customised to reflect the HyperStore deployment in question. Stanzas should be added or removed under the ‘members’ section as needed to reflect all sites that are to be used in the deployment.

Be sure to click the **Update** button when finished to write the configuration to disk.

Note: Indentation is important and **must** be preserved, otherwise the underlying Polaris service will throw an error. As many PDF readers discard leading whitespace, the below example can also be found at the following location as a plain text file, in an effort to preserve the indentation when copying:

https://pdfs.loadbalancer.org/Cloudian_HyperStore_Deployment_Guide-PolarisConfigExample.txt

```
globalnames:
   s3-region1.domain.tld:
      pool: s3-nodes
      ttl: 5

pools:
   s3-nodes:
      monitor: http
      monitor_params:
         use_ssl: true
         hostname: s3-region1.domain.tld
         url_path: /.healthCheck
      lb_method: twrr
      fallback: any
      members:
         - ip: 10.0.0.2
           name: node1-dc1
           weight: 1
         - ip: 172.16.0.2
           name: node2-dc2
           weight: 1
         - ip: 192.168.1.2
           name: node3-dc3
           weight: 1
```

Topology Config

Using the web user interface of the master appliance, navigate to **Cluster Configuration > GSLB Configuration** and select **Topology Config** from the drop-down list.

If GSLB has never been configured then the default example topology configuration will be displayed in the text box.

What follows is a Cloudian specific example topology configuration which covers the same three example sites from the example configuration in the previous section. It can be copy and pasted to replace the default topology example and can be used as a basis for creating a deployment-specific configuration.

The titles and IP subnets should be customised to reflect the HyperStore deployment in question. Stanzas should be
added or removed as needed to reflect all sites that are to be used in the deployment.

Be sure to click the **Update** button when finished to write the configuration to disk.

**datacenter1:**
- 10.0.0.0/24

**datacenter2:**
- 172.16.0.0/24

**datacenter3:**
- 192.168.0.0/18

**Applying the Configuration**

If updating the configuration of an **existing GSLB configuration** then press the **Reload GSLB** button when prompted.

If setting up the GSLB service for the **first time** then the service **must** undergo a full restart to be enabled and for the configuration to be applied. To do this, using the web user interface, navigate to **Maintenance > Restart Services** and click the **Restart GSLB** button. If using an HA pair of load balancers at a site then this procedure **must** also be carried out on the slave appliance. Once this procedure been carried out it never needs to be done again; the GSLB service only needs to be **reloaded** in the future when prompted, for example following a configuration change, something that only needs to be carried out on the master appliance.

**DNS Server Configuration**

Once the GSLB service has been configured on the master load balancer at every site, ensuring that the configuration is **identical across all sites**, the DNS server at each site must then be configured for GSLB.

The DNS server at each site must be configured to delegate DNS requests for the subdomain in question to the load balancers; the load balancers’ GSLB services will serve the appropriate IP addresses to the DNS servers. Using the example presented throughout this appendix, the DNS server at each site would be configured with a delegation for the domain s3-region1.domain.tld. The domain would be delegated to every load balancer across every site, which provides multi-site redundancy.

The exact steps for creating a DNS delegation vary between different DNS servers and are outside the scope of this document. For further information, a blog post that walks through creating a DNS delegation on a Microsoft DNS server in the context of setting up GSLB on our appliance can be found at [https://www.loadbalancer.org/blog/loadbalancer-org-releases-a-gslb/](https://www.loadbalancer.org/blog/loadbalancer-org-releases-a-gslb) (see the section titled “Delagating your subdomain to your GSLB’s using Microsoft’s DNS Server”).
Direct routing, also known as direct server return or DSR, is a method of load balancing. With direct routing, reply traffic flows directly from the back end servers to the clients. In this way, the load balancer is completely bypassed on the return journey for a given connection, thus removing the load balancer as a potential bottleneck for traffic on the return path.

This alternative method of load balancing can benefit read-intensive deployments which feature a large reply traffic to request traffic ratio. For example, consider the scenario where a typical client request is 10 kB in size while a typical reply is 10 GB in size (perhaps file retrieval or video streaming). Direct routing benefits such scenarios: the much larger volume of reply traffic bypasses the load balancer and is not limited by the load balancer’s network throughput. The reply traffic is instead limited by the total available network bandwidth between the servers and the clients, which is limited only by the underlying infrastructure.

Caveats

There are caveats for using the direct routing load balancing method which should be considered:

- The load balancers must be on the same network segment / switching fabric as the HyperStore nodes (due to the fact that this load balancing method works by rewriting MAC addresses, i.e. operates at layer 2 of the OSI model)
- Each HyperStore node must own the VIP address so that they can all accept and reply to the load balanced traffic. This address should be assigned to a loopback network adaptor
- Each HyperStore node must be configured to not reply to ARP requests for the VIP address or advertise that they
For guidance on configuring the HyperStore nodes for direct routing, in the context of the caveats described above, please consult with Cloudian Sales Engineering or Support.

Appliance Configuration For Cloudian HyperStore – Using Layer 4 DR Mode (Direct Routing)

Configuring VIP 1 – Cloudian Management Console

Configuring the Virtual Service (VIP)

1. Using the web user interface, navigate to Cluster Configuration > Layer 4 – Virtual Services and click on Add a new Virtual Service
2. Define the Label for the virtual service as required, e.g. cmc.cloudian-hyperstore
3. Set the Virtual Service IP Address field to the required IP address, e.g. 192.168.87.67
4. Set the Ports field to 8888,8443
5. Leave the Protocol set to TCP
6. Leave the Forwarding Method set to Direct Routing
7. Click Update to create the virtual service
8. Click Modify next to the newly created VIP
9. Ensure that the Persistence Enable checkbox is checked and that the Timeout is set to 1800
10. Set the Health Checks Check Type to Connect to port
11. Set the Check Port to 8443
12. Click Update

Defining the Real Servers (RIPs)

1. Using the web user interface, navigate to Cluster Configuration > Layer 4 – Real Servers and click on Add a new Real Server next to the newly created VIP
2. Define the Label for the real server as required, e.g. cloudian-node1
3. Set the Real Server IP Address field to the required IP address, e.g. 10.10.10.11
4. Click Update
5. Repeat these steps to add additional HyperStore nodes as real servers as required

Configuring VIP 2 – S3 Client Requests (HTTP)

Configuring the Virtual Service (VIP)

1. Using the web user interface, navigate to Cluster Configuration > Layer 4 – Virtual Services and click Modify next to the previously created CMC VIP
2. Click Duplicate Service and confirm when prompted
3. Define the Label for the new virtual service as required, e.g. s3.cloudian-hyperstore
4. Set the Virtual Service IP Address field to the required IP address, e.g. 192.168.87.67
5. Set the Ports field to 80
6. Un-check the Persistence Enable checkbox
7. Clear the Health Checks Check Port field leaving it empty
8. Click Update

Configuring VIP 3 – S3 Client Requests (HTTPS)

Configuring the Virtual Service (VIP)

1. Using the web user interface, navigate to Cluster Configuration > Layer 4 – Virtual Services and click Modify next to the previously created S3 VIP
2. Click Duplicate Service and confirm when prompted

3. Define the Label for the new virtual service as required, e.g. https.s3.cloudian-hyperstore
4. Set the Virtual Service IP Address field to the required IP address, e.g. 192.168.87.67
5. Set the Ports field to 443
6. Click Update

Configuring VIP 4 – Admin API Requests

Configuring the Virtual Service (VIP)

1. Using the web user interface, navigate to Cluster Configuration > Layer 4 – Virtual Services and click Modify next to the previously created S3 HTTPS VIP
2. Click Duplicate Service and confirm when prompted

3. Define the Label for the new virtual service as required, e.g. api.cloudian-hyperstore
4. Set the Virtual Service IP Address field to the required IP address, e.g. 192.168.87.67
5. Set the Ports field to 19443
6. Click Update

3 – Using the PROXY Protocol to Retain Client IP Addresses

The PROXY protocol is an HTTP-like protocol which allows a reverse proxy, such as a layer 7 load balancer, to pass along the original client’s source IP address. The PROXY protocol is not HTTP compliant and cannot be read by applications that are expecting to receive standard HTTP requests.

HyperStore 7.0 and later supports reading traffic that uses the PROXY protocol. When correctly configured, this makes the HyperStore nodes aware of the original client’s source IP address for S3 requests. This allows for the use of S3 bucket policies and the HyperStore billing whitelist feature, both of which require clients’ IP addresses to function. This also means that clients’ real IP addresses are recorded in the S3 request log, which can assist with troubleshooting and monitoring.

Configuring The HyperStore Nodes To Accept The PROXY Protocol

1. Log on to the Puppet master node (the HyperStore node on which the installation script was run)
2. In common.csv (the main HyperStore configuration file), set the value of s3_proxy_protocol_enabled to true

   Note: Step 2 can be carried out using the vi text editor. The common.csv file to be edited is located at:
   /etc/cloudian-<version>-puppet/manifests/extdata/common.csv
   
   The line
   s3_proxy_protocol_enabled,false
   should be amended to read
   s3_proxy_protocol_enabled,true
   and the file should then be saved.

   Alternatively, executing the following series of commands from the console will make the necessary change:
   find /etc -iname "common.csv" | xargs -n1 sed -i.bkup 's/s3_proxy_protocol_enabled,false/s3_proxy_protocol_enabled,true/g'

3. Push the configuration change to all HyperStore nodes in the cluster

   Note: The easiest way to carry out step 3 is to change into the installation staging directory at the command line and then run the HyperStore installation/configuration script. To run the script, execute:
   ./cloudianInstall.sh

   Using the script, enter 2 for Cluster Management:
Enter **b** for *Push Configuration Settings to Cluster*:

Press **Enter** at the prompt to select all nodes, and then wait for a success message to be displayed:

```
Run Puppet to configure agent nodes.

region region contains the following hosts: cloudlan1 cloudlan2 cloudlan3
Enter a comma-separated list of hosts in office to execute agents on? [empty for all] {}
Redirecting to /bin/puppetd --write-server.service
Configuring agent node cloudlan1.
Ready to run Puppet agent on host cloudlan1. This could take some time.
Configuring agent node cloudlan2.
Configuring agent node cloudlan3.
Ready to run Puppet agent on host cloudlan2. This could take some time.
Ready to run Puppet agent on host cloudlan3. This could take some time.
All Puppet agent runs completed successfully in office region.
Puppet agent daemons is now running on cloudlan1.
Puppet agent daemons is now running on cloudlan2.
Puppet agent daemons is now running on cloudlan3.
Puppet agent run ended for office.
Press any key to continue ... []
```
4. Restart the S3 service to apply the new configuration across all nodes

Note: From the Cluster Management menu of the installation/configuration script, enter `c` for Manage Services, enter 5 for S3 service, enter “restart” to trigger a cluster-wide restart of the service, and then wait for each success message to be displayed:

```
Service Management
0) All services
1) Royals Credentials
2) Royals OSS
3) Execdora
4) HyperStore service
5) S3 service
6) Royals Monitor
7) Cloudian Agent
8) PROXY
9) Cloudian Management Console (CMC)
P) Popout service (status only)
X) Quit
You can execute the following list of commands:
start, stop, status, restart, version, force-stop, node-stop, node-stop
Select a service to manage: 5
Enter command: {start,stop,status,restart,version} restart
Executing Cloudian S3 service command restart ...
On host cloudian1:
/etc/init.d/cloudian-s3 restart => Restarting cloudian-s3 (via systemctl): [ OK ]
On host cloudian2:
/etc/init.d/cloudian-s3 restart => Restarting cloudian-s3 (via systemctl): [ OK ]
On host cloudian4:
/etc/init.d/cloudian-s3 restart => Restarting cloudian-s3 (via systemctl): [ OK ]
Press any key to continue ...
```

Appliance Configuration

Configuring Additional VIP 1 – S3 Client Requests Using the PROXY Protocol (HTTP)

1. Using the web user interface, navigate to Cluster Configuration > Layer 7 – Virtual Services and click **Modify** next to the existing **S3** VIP, e.g. s3.cloudian-hyperstore
2. Click **Duplicate Service** and confirm when prompted

3. Define the **Label** for the new virtual service as required, e.g. pp_s3.cloudian-hyperstore
4. Set the **Ports** field to 81
5. In the **Other** section click **Advanced** to expand the menu
6. Set **Send Proxy Protocol** to **Send Proxy V1**
7. Click **Update**
Configuring Additional VIP 2 – S3 Client Requests Using the PROXY Protocol (HTTPS)

1. Using the web user interface, navigate to Cluster Configuration > Layer 7 – Virtual Services and click Modify next to the existing S3 HTTPS VIP, e.g. https.s3.cloudian-hyperstore
2. Click Duplicate Service and confirm when prompted

3. Define the Label for the new virtual service as required, e.g. pp_https.s3.cloudian-hyperstore
4. Set the Ports field to 4431
5. In the Other section click Advanced to expand the menu
6. Set Send Proxy Protocol to Send Proxy V1
7. Click Update

Finalizing the Configuration
To apply the new settings, HAProxy must be restarted as follows:

1. Using the WebUI, navigate to: Maintenance > Restart Services and click Reload HAProxy

The two new virtual services will now be ready for use and will send traffic to the HyperStore nodes using the PROXY protocol.
4 – Clustered Pair Configuration – Adding a Slave Unit
If you initially configured just the master unit and now need to add a slave - our recommended procedure, please refer to the relevant section below for more details:

Note: A number of settings are not replicated as part of the master/slave pairing process and therefore must be manually configured on the slave appliance. These are listed below:

- Hostname & DNS settings
- Network settings including IP addresses, bonding configuration and VLANs
- Routing configuration including default gateways and static routes
- Date & time settings
- Physical – Advanced Configuration settings including Internet Proxy IP address & port, Firewall table size, SMTP relay and Syslog server
- SNMP settings
- Graphing settings
- Firewall Script & Firewall Lockdown Script settings
- Software updates

Version 7:
Please refer to Chapter 8 – Appliance Clustering for HA in the v7 Administration Manual.

Version 8:
To add a slave node – i.e. create a highly available clustered pair:

- Deploy a second appliance that will be the slave and configure initial network settings
- Using the WebUI, navigate to: Cluster Configuration > High-Availability Configuration
- Specify the IP address and the loadbalancer users password (the default is 'loadbalancer') for the slave (peer) appliance as shown above
- Click **Add new node**
- The pairing process now commences as shown below:

![Create a Clustered Pair](image)

- Once complete, the following will be displayed:

![High Availability Configuration - Master](image)

- To finalize the configuration, restart heartbeat and any other services as prompted in the blue message box at the top of the screen

Note: Clicking the **Restart Heartbeat** button on the master appliance will also automatically restart heartbeat on the slave appliance.

Note: Please refer to chapter 9 – Appliance Clustering for HA in the *Administration Manual* for more detailed information on configuring HA with 2 appliances.
## 16. Document Revision History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Change</th>
<th>Reason for Change</th>
<th>Changed By</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0.0</td>
<td>4 April 2018</td>
<td>Initial version</td>
<td></td>
<td>AH</td>
</tr>
<tr>
<td>1.1.0</td>
<td>7 November 2018</td>
<td>Changed the &quot;Health Checks&quot; section to talk about the new built-in health checks and the rewritten custom health check Removed the two custom health checks from the appendix and replaced them with the updated new single script, along with a link to download it online Changed the VIP configuration instructions to refer to the new built-in health checks</td>
<td>Required updates</td>
<td>AH</td>
</tr>
<tr>
<td>1.1.1</td>
<td>6 December 2018</td>
<td>Added the new &quot;Company Contact Information&quot; page</td>
<td>Required updates</td>
<td>AH</td>
</tr>
<tr>
<td>1.1.2</td>
<td>18 December 2018</td>
<td>Modified the 'Loadbalancer.org Appliances Supported' section to state that only the Enterprise 10G, Enterprise 40G, and Enterprise VA MAX models are supported</td>
<td>Required updates</td>
<td>AH</td>
</tr>
<tr>
<td>1.1.3</td>
<td>31 January 2019</td>
<td>Added a clarifying note to the section on creating the CMC virtual service, explicitly stating that it requires the default setting of source IP persistence to be left enabled</td>
<td>Required updates</td>
<td>AH</td>
</tr>
<tr>
<td>1.2</td>
<td>3 April 2019</td>
<td>Changed Loadbalancer.org software versions supported to V8.3.6 and later, to account for the new authenticated health check in the WebUI Replaced the explanatory note regarding persistence on the CMC VIP with an explicit instruction to set the persistence option to HTTP cookie Added instructions for setting up a negotiate health check for the CMC VIP, and added a new explanatory note regarding the</td>
<td>Required updates</td>
<td>AH</td>
</tr>
<tr>
<td>Version</td>
<td>Date</td>
<td>Changes</td>
<td>Updates</td>
<td>Author</td>
</tr>
<tr>
<td>---------</td>
<td>------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------</td>
<td>--------</td>
</tr>
<tr>
<td>1.2.1</td>
<td>4 April 2019</td>
<td>Corrected the persistence mode for the CMC VIP to source IP, as it is a 'TCP mode' VIP. Removed the note for the CMC VIP that explained the rationale for checking against the TLS port, for simplicity.</td>
<td>Required updates</td>
<td>AH</td>
</tr>
<tr>
<td>1.2.2</td>
<td>25 April 2019</td>
<td>Added additional screenshots regarding the health checks, for clarity. Added notes regarding setting the health check host header if needed.</td>
<td>Required updates</td>
<td>AH</td>
</tr>
<tr>
<td>1.2.3</td>
<td>4 June 2019</td>
<td>Changed the health check for the Cloudian Management Console VIP, based on updated documentation from Cloudian.</td>
<td>Required updates</td>
<td>AH</td>
</tr>
<tr>
<td>1.2.4</td>
<td>5 July 2019</td>
<td>Changed the health check 'Request to send' fields from .healthCheck to /.healthCheck, and updated screen shots accordingly.</td>
<td>Required updates</td>
<td>AH</td>
</tr>
<tr>
<td>1.3.0</td>
<td>27 August 2019</td>
<td>Styling and layout. Changed the health check for the Cloudian Management Console.</td>
<td>General styling updates</td>
<td>AH</td>
</tr>
<tr>
<td>1.4.0</td>
<td>31 October 2019</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>----------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VIP to use an OPTIONS check</td>
<td>Required updates</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rewrote the instructions for setting up the virtual services to make use of the new ‘Duplicate Service’ function</td>
<td>Added support for the PROXY protocol at the request of Cloudian</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Moved the health check specific screenshots and note blocks above the ‘Click Update’ instructions for clarity</td>
<td>Added additional deployment options</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Added paragraph ‘GSLB / Location Affinity’ and associated appendix with configuration instructions</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Added paragraph ‘Alternative Load Balancing Method for Read-Intensive Deployments (Direct Routing)’ and associated appendix with configuration instructions</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Added a note at the end of the S3 VIP setup instructions pointing to a new appendix, ‘Using the PROXY Protocol to Retain Client IP Addresses’, containing configuration instructions</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Changed the Loadbalancer.org software version supported to 8.4.1 due to now needing health checks using the PROXY protocol</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Updated the advice for configuring multithreading in HAProxy to recommend a minimum of 4 threads and a maximum of the total number of available threads</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
About Loadbalancer.org

Loadbalancer.org's mission is to ensure that its clients' businesses are never interrupted. The load balancer experts ask the right questions to get to the heart of what matters, bringing a depth of understanding to each deployment. Experience enables Loadbalancer.org engineers to design less complex, unbreakable solutions - and to provide exceptional personalized support.

United Kingdom

Loadbalancer.org Ltd.
Compass House, North Harbour Business Park, Portsmouth, PO6 4PS
UK: +44 (0) 330 380 1064
sales@loadbalancer.org
support@loadbalancer.org

Canada

Loadbalancer.org Appliances Ltd.
300-422 Richards Street, Vancouver, BC V6B 2Z4, Canada
TEL: +1 302.213.0122
sales@loadbalancer.org
support@loadbalancer.org

United States

Loadbalancer.org, Inc.
4550 Linden Hill Road, Suite 201
Wilmington, DE 19808, USA
TEL: +1 833.274.2566
sales@loadbalancer.org
support@loadbalancer.org

Germany

Loadbalancer.org GmbH
Tengstraße 2780798,
München, Germany
TEL: +49 (0)89 2000 2179
sales@loadbalancer.org
support@loadbalancer.org