LOADBALANCER

DEPLOYMENT GUIDE

Load Balancing Ceph Object Gateways

Version 1.3.1

Table of Contents

1. About this Guide

This guide details the steps required to configure a load balanced Ceph Object Gateway environment utilizing Loadbalancer.org appliances. It covers the configuration of the load balancers and also any Ceph 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 Administration Manual.

2. Loadbalancer.org Appliances Supported

All our products can be used with Ceph Object Gateway. For full specifications of available models please refer to https://www.loadbalancer.org/products.

Some features may not be available or fully supported in all cloud platforms due to platform specific limitations. For more details, please refer to the "Main Differences to our Standard (Non-Cloud) Product" section in the appropriate cloud platform Quick Start Guide or check with Loadbalancer.org support.

3. Software Versions Supported

3.1. Loadbalancer.org Appliance

- V8.9.1 and later

Note	The screenshots used throughout this document aim to track the latest Loadbalancer.org software version. If you're using an older version, or the very latest, the screenshots presented here may not match your WebUI exactly.

3.2. Ceph

- All versions

4. Ceph

Ceph is a free and open-source object storage solution. It provides for the building of massively scalable and decentralised storage clusters, up to petabytes or exabytes of data, using standard commodity hardware.

At the heart of Ceph is its Reliable Autonomic Distributed Object Store (RADOS) which provides the underlying storage functionality. Ceph provides three interfaces for accessing its RADOS storage cluster: the Ceph File System (CephFS), which presents storage as a traditional POSIX-compatible file system; the Ceph Block Device (RBD), which allows storage to be mounted as a block device; and the Ceph Object Gateway (RGW), which provides S3 and Swift-compatible API access and is the subject of this document.

The Ceph project is backed by a variety of organisations, with contributions and broader community involvement notably from Canonical, CERN, Fujitsu, Intel, Red Hat, SanDisk, and SUSE. Commercial support and professional services for Ceph are available from Red Hat, as ***Red Hat Ceph Storage***, SUSE, as ***SUSE Enterprise Storage***, and Canonical, as "Ceph storage on Ubuntu".

5. Load Balancing Ceph

> **Note** It's highly recommended that you have a working Ceph environment with working Ceph Object Gateways first before implementing the load balancer.

5.1. Load Balancing & HA Requirements

For high availability and scalability, multiple Ceph Object Gateways must be deployed as part of a Ceph deployment. Additional gateways can be deployed over time to increase capacity to meet demand as needed.

5.2. Persistence (aka Server Affinity)

Ceph Object Gateways do not require session affinity at the load balancing layer.

5.3. Virtual Service (VIP) Requirements

To provide load balancing and HA for Ceph Object Gateways, one VIP is required:

- Ceph RGW

Optionally, an additional VIP can be used to make the Ceph Dashboard highly available from a single IP address, if the optional dashboard module is in use:

- Ceph Dashboard

5.4. Port Requirements

Ceph is extremely flexible and its user facing services can be configured to use HTTP and/or HTTPS over whichever ports are desired.

The following table is illustrative only: it shows a range of services using a mixture of default ports and likely real-world port values. A specific deployment may well use different ports. The services that are likely to be load balanced are:

Port	Protocols	Use
80	TCP/HTTP	Ceph Object Gateway Access
443	TCP/HTTPS	Secure Ceph Object Gateway Access (HTTPS)
8080	TCP/HTTP	Ceph Dashboard Access
8443	TCP/HTTPS	Secure Ceph Dashboard Access (HTTPS)

5.5. TLS/SSL Termination

Default/Recommended Option: No TLS/SSL Termination (TLS/SSL Pass-through)

If secure HTTPS services are in use, it is highly recommended to **leave all TLS/SSL termination being performed on the Ceph Object Gateways**. The gateway servers are best positioned to perform this function and, crucially, it's easy to provision additional gateway servers as needed to handle a growing computational load from increased

HTTPS use.

Legacy Support Option: Performing TLS/SSL Termination on the Load Balancer

One exception to the above recommendation is when load balancing older Ceph deployments. Ceph version 11.0.1, Kraken, introduced support for TLS/SSL using the CivetWeb HTTP frontend on Ceph Object Gateways. Versions prior to that supported HTTP, plain text access to Ceph Object Gateways only.

To use HTTPS on an older deployment that doesn't have native HTTPS support, a TLS/SSL termination can be configured on the load balancer. This allows clients to connect to the load balancer using HTTPS and the load balancer can then send plain text HTTP to the Ceph Object Gateways, as normal.

To configure the load balancer in this way, first run through the instructions in Appliance Configuration for Ceph Object Gateways – Using Layer 7 SNAT Mode (HAProxy). Once complete, then proceed to run through the instructions in Legacy Support Option: Performing TLS/SSL Termination on the Load Balancer.

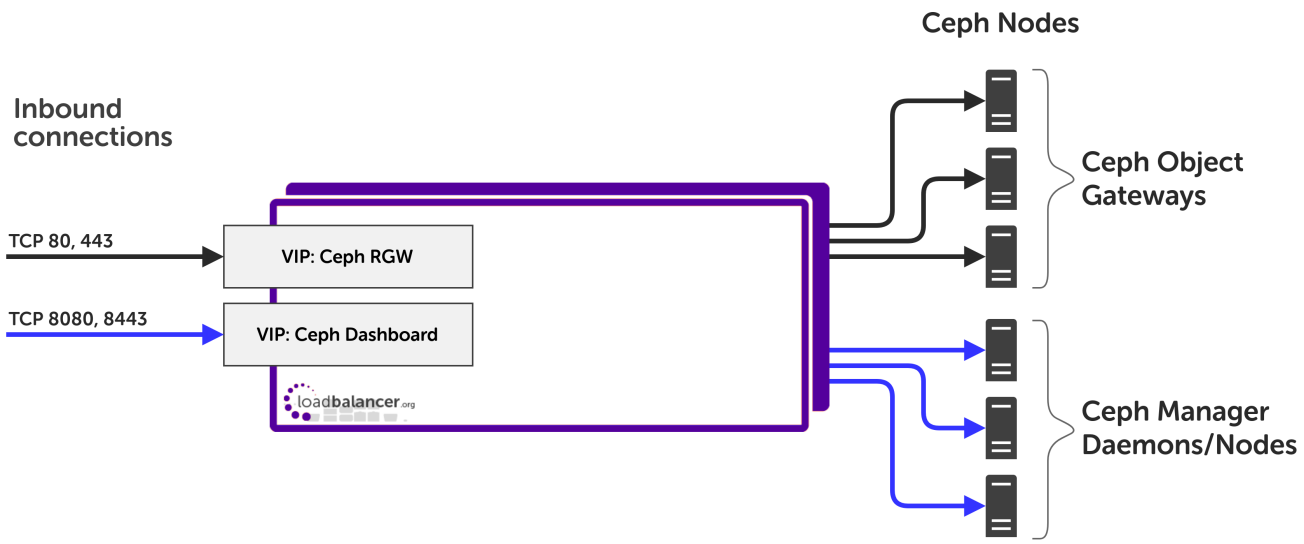
> **Note** Layer 4 DR Mode is not compatible with the "legacy support", TLS/SSL termination option.

5.6. GSLB / Location Affinity

For multi-site Ceph 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 Ceph Object Gateway service and/or load balancers are offline then local clients are automatically directed to a functioning Ceph Object Gateway service at another site.

A full explanation and instructions on setting up this optional feature can be found in Configuring GSLB / Location Affinity.

6. Deployment Concept



VIP = **V**irtual **IP** Address

> **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 Configuring HA - Adding a Secondary Appliance for more details on configuring a clustered pair.

7. Load Balancer Deployment Methods

The load balancer can be deployed in 4 fundamental ways: ***Layer 4 DR mode***, ***Layer 4 NAT mode***, ***Layer 4 SNAT mode,*** and ***Layer 7 SNAT mode***.

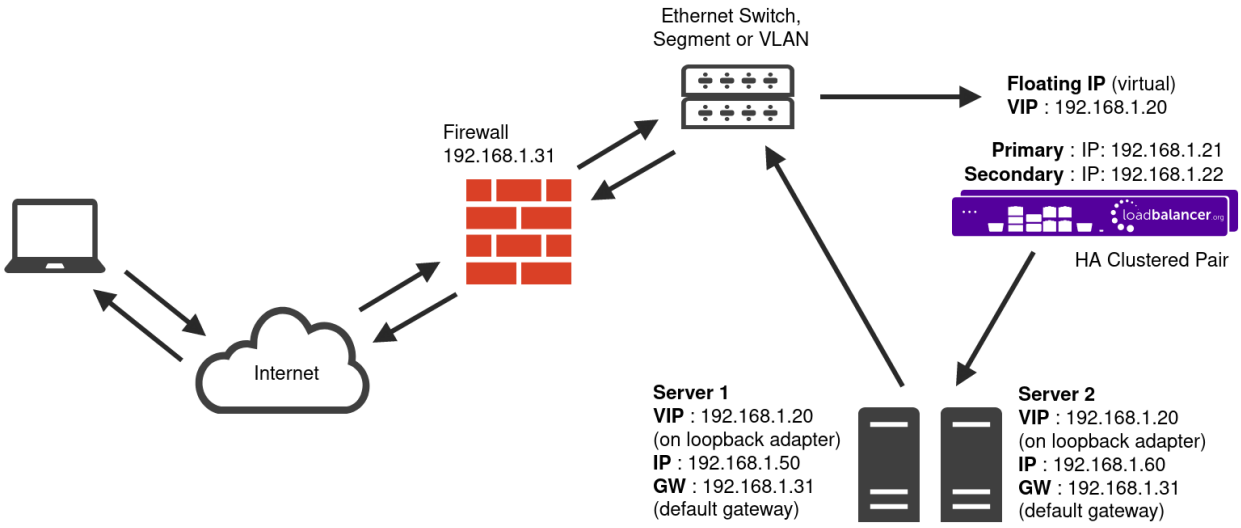
When load balancing Ceph Object Gateways, using layer 4 DR mode or layer 7 SNAT mode is recommended. It is also possible to use layer 4 TUN mode or layer 4 NAT mode in specific circumstances, however these load balancing methods come with many caveats and should only be considered if both layer 4 DR mode and layer 7 SNAT mode have been discounted.

Layer 4 DR mode and layer 7 SNAT mode are both described below and are used for the configurations presented in this guide. For configuring using DR mode please refer to Appliance Configuration for Ceph Object Gateways – Using Layer 4 DR Mode (LVS/DR) and for configuring using layer 7 SNAT mode refer to Appliance Configuration for Ceph Object Gateways – Using Layer 7 SNAT Mode (HAProxy).

7.1. Layer 4 DR Mode

Layer 4 DR (Direct Routing) mode is a very high performance solution that requires little change to your existing infrastructure.

> **Note** Kemp, Brocade, Barracuda & A10 Networks call this ***Direct Server Return*** and F5 call it ***nPath***.



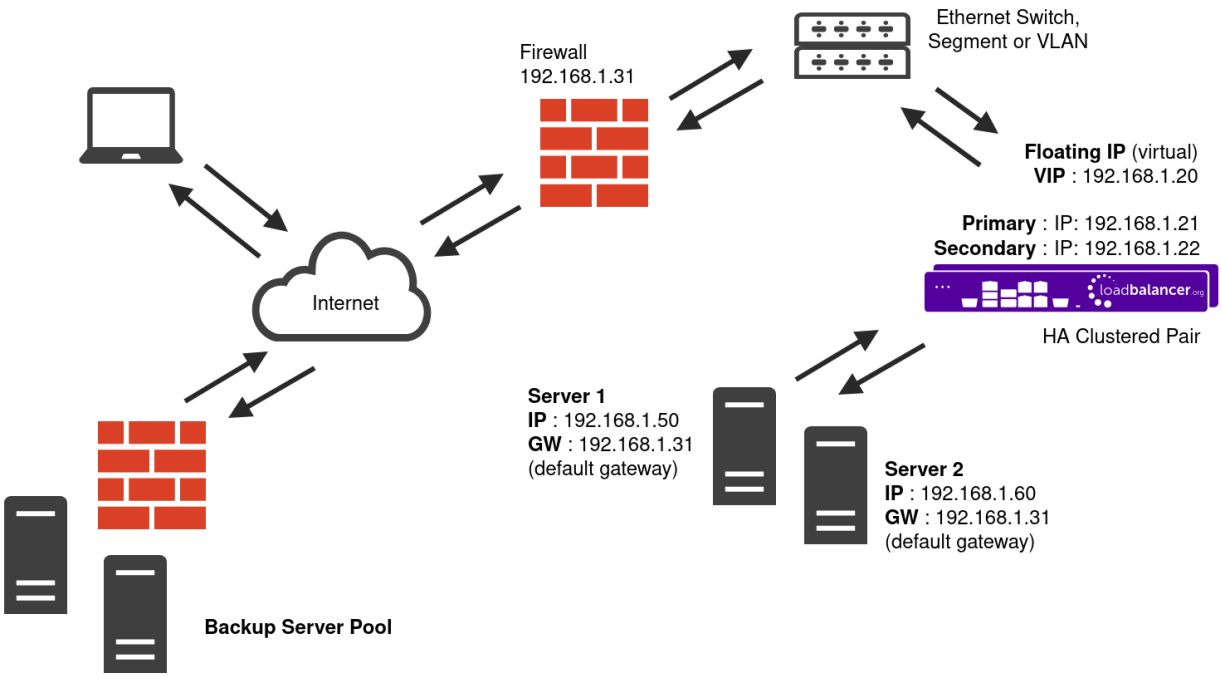
- DR mode works by changing the destination MAC address of the incoming packet to match the selected Real Server on the fly which is very fast.
- When the packet reaches the Real Server it expects the Real Server to own the Virtual Services IP address (VIP). This means that each Real Server (and the load balanced application) must respond to both the Real Server's own IP address and the VIP.
- The Real Server should not respond to ARP requests for the VIP. Only the load balancer should do this. Configuring the Real Server in this way is referred to as "Solving the ARP Problem". For more information

please refer to DR Mode Considerations.

- On average, DR mode is 8 times quicker than NAT mode for HTTP and much faster for other applications such as Remote Desktop Services, streaming media and FTP.
- The load balancer must have an interface in the same subnet as the Real Servers to ensure layer 2 connectivity which is required for DR mode to operate.
- The VIP can be brought up on the same subnet as the Real Servers or on a different subnet provided that the load balancer has an interface in that subnet.
- Port translation is not possible with DR mode, e.g. VIP:80 → RIP:8080 is not supported.
- DR mode is transparent, i.e. the Real Server will see the source IP address of the client.

7.2. Layer 7 SNAT Mode

Layer 7 SNAT mode uses a proxy (HAProxy) at the application layer. Inbound requests are terminated on the load balancer and HAProxy generates a new corresponding request to the chosen Real Server. As a result, Layer 7 is typically not as fast as the Layer 4 methods. Layer 7 is typically chosen when either enhanced options such as SSL termination, cookie based persistence, URL rewriting, header insertion/deletion etc. are required, or when the network topology prohibits the use of the layer 4 methods.



- Because layer 7 SNAT mode is a full proxy, any server in the cluster can be on any accessible subnet including across the Internet or WAN.
- Layer 7 SNAT mode is not transparent by default, i.e. the Real Servers will not see the source IP address of the client, they will see the load balancer's own IP address by default, or any other local appliance IP address if preferred (e.g. the VIP address). This can be configured per layer 7 VIP. If required, the load balancer can be configured to provide the actual client IP address to the Real Servers in 2 ways. Either by inserting a header that contains the client's source IP address, or by modifying the Source Address field of the IP packets and replacing the IP address of the load balancer with the IP address of the client. For more information on these methods please refer to Transparency at Layer 7.

- Layer 7 SNAT mode can be deployed using either a one-arm or two-arm configuration. For two-arm deployments, **eth0** is normally used for the internal network and **eth1** is used for the external network although this is not mandatory.
- Requires no mode-specific configuration changes to the load balanced Real Servers.
- Port translation is possible with Layer 7 SNAT mode, e.g. VIP:80 → RIP:8080 is supported.
- You should not use the same RIP:PORT combination for layer 7 SNAT mode VIPs and layer 4 SNAT mode VIPs because the required firewall rules conflict.

7.3. Our Recommendation

The load balancing method to use depends on a number of factors. To help choose, presented below is a brief summary of each method.

Layer 4 DR Mode (LVS/DR)

This mode delivers the **best raw performance and throughput**. It requires making a few simple changes on the Ceph Object Gateways (see Configuring Ceph Object Gateways for Layer 4 DR Mode (LVS/DR) for details).

This is **particularly well suited to read-intensive storage scenarios** because the bandwidth of the return traffic is maximised by completely bypassing the load balancer. This is also known as 'direct server return'.

This mode has the caveat that the load balancer ***must*** reside on the same network segment as the Ceph Object Gateways. If that condition cannot be met then layer 7 SNAT mode should be used.

Layer 7 SNAT Mode (HAProxy)

This is the **simplest and most flexible mode to configure** as no changes are required on the Ceph Object Gateways. The servers can be located anywhere in relation to the load balancer, provided that traffic can be routed between the load balancer and the servers.

In this mode, the load balancer uses HAProxy and acts as a full reverse proxy between the users and the Ceph Object Gateways.

8. Configuring Ceph for Load Balancing

8.1. Configuring Ceph Object Gateways for Layer 4 DR Mode (LVS/DR)

If Layer 4 DR mode is used then the 'ARP problem' must be solved. This involves configuring each Real Server (each Ceph Object Gateway in this context) to be able to receive traffic destined for the VIP address, and ensuring that each Real Server does not respond to ARP requests for the VIP address – only the load balancer should do this.

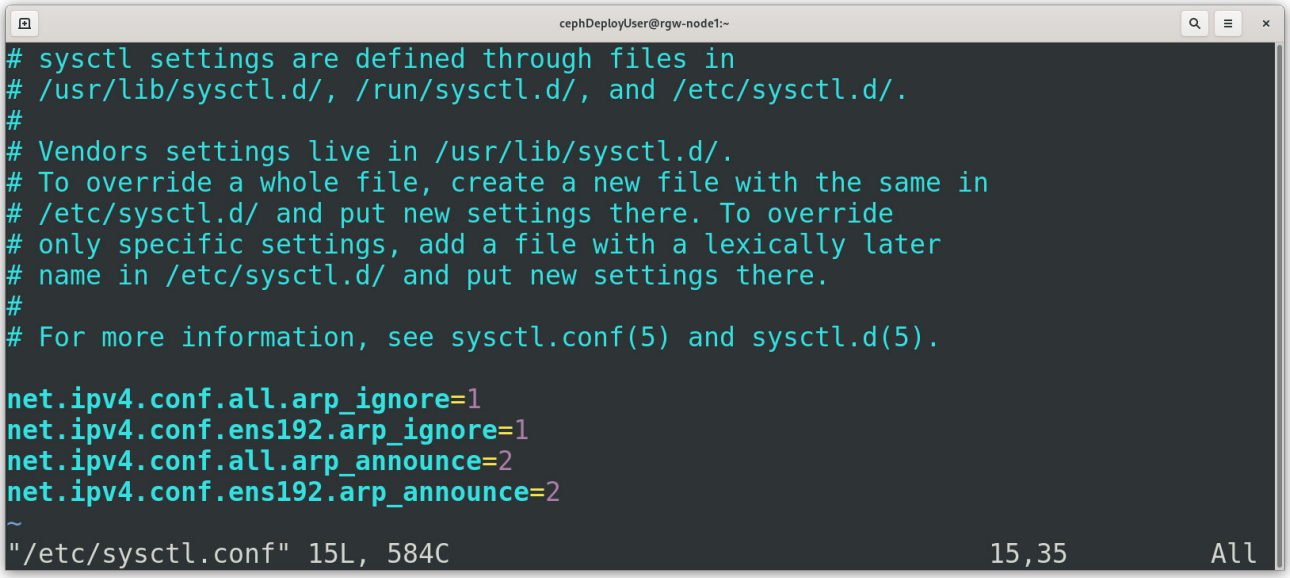
The steps below are for Linux servers. For instructions for FreeBSD servers, please refer to DR Mode Considerations.

Step 1: Re-configure ARP on the Real Servers

To do this, add the following lines to **/etc/sysctl.conf**:

```
net.ipv4.conf.all.arp_ignore=1
net.ipv4.conf.ens192.arp_ignore=1
net.ipv4.conf.all.arp_announce=2
net.ipv4.conf.ens192.arp_announce=2
```

For example:



> **Note** Adjust the commands shown above to suit the network configuration of your server, i.e. change **ens192** to match the name of your server's network interface.

Step 2: Re-configure DAD on the Real Servers (this step is only necessary if using IPv6 addresses)

To do this add the following lines to /etc/sysctl.conf:

```
net.ipv6.conf.lo.dad_transmits=0
net.ipv6.conf.lo.accept_dad=0
```

Step 3: Apply these settings

Either reboot the Real Server or run the following command to apply these settings:

```
/sbin/sysctl -p
```

```
cephDeployUser@rgw-node1:~
[cephDeployUser@rgw-node1 ~]$ sudo /sbin/sysctl -p
net.ipv4.conf.all.arp_ignore = 1
net.ipv4.conf.ens192.arp_ignore = 1
net.ipv4.conf.all.arp_announce = 2
net.ipv4.conf.ens192.arp_announce = 2
[cephDeployUser@rgw-node1 ~]$
```

Step 4: Add the Virtual Service's IP address to the loopback adapter

To temporarily add the VIP address, e.g. for an initial test, run the following command for the VIP address:

```
ip addr add dev lo <IPv4-VIP>/32
```

For IPv6 addresses use:

```
ip addr add dev lo <IPv6-VIP>/128
```

```
cephDeployUser@rgw-node1:~
[cephDeployUser@rgw-node1 ~]$ sudo ip addr add dev lo 172.26.11.231/32
[cephDeployUser@rgw-node1 ~]$
```

To make this new address permanent and persist across server reboots, add the command to an appropriate startup script, such as /etc/rc.local. For example, on CentOS 7 / RHEL 7:

```
cephDeployUser@rgw-node1:~
#!/bin/bash
# THIS FILE IS ADDED FOR COMPATIBILITY PURPOSES
#
# It is highly advisable to create own systemd services or udev rules
# to run scripts during boot instead of using this file.
#
# In contrast to previous versions due to parallel execution during boot
# this script will NOT be run after all other services.
#
# Please note that you must run 'chmod +x /etc/rc.d/rc.local' to ensure
# that this script will be executed during boot.

touch /var/lock/subsys/local

ip addr add dev lo 172.26.11.231/32
~
~
"/etc/rc.local" 15L, 510C                                    15,35         All
```

As described in the help text, on CentOS 7 / RHEL 7 it is necessary to also execute

```
chmod +x /etc/rc.d/rc.local
```

to make the rc.local script executable during boot, otherwise the VIP address will not be added at boot.

Note

As an *alternative* to using a startup script such as rc.local (which, on some Linux distributions, may be considered a legacy option or is offered "for compatibility purposes" only), the VIP address can be added to the loopback adaptor by modifying the OS appropriate NIC configuration file.

As an example, on CentOS 7 / RHEL 7 the loopback adaptor configuration file at /etc/sysconfig/network-scripts/ifcfg-lo can be appended to in order to add the additional IP address, e.g. by appending:

```
IPADDR1=172.26.11.231
NETMASK1=255.255.255.255
```

```
cephDeployUser@rgw-node1:~
DEVICE=lo
IPADDR=127.0.0.1
NETMASK=255.0.0.0
NETWORK=127.0.0.0
# If you're having problems with gated making 127.0.0.0/8 a martian,
# you can change this to something else (255.255.255.255, for example)
BROADCAST=127.255.255.255
ONBOOT=yes
NAME=loopback
IPADDR1=172.26.11.231
NETMASK1=255.255.255.255
~
~
"/etc/sysconfig/network-scripts/ifcfg-lo" 11L, 301C          1,1           All
```

The network service would then need to be restarted to put the new configuration and IP

address into use, e.g. by executing:

```
systemctl restart network
```

8.2. Configuring the Dashboard Module for Load Balancing

The Ceph Dashboard module is an optional component of Ceph. If desired, an additional virtual service can be used to make the Ceph Dashboard highly available from a single IP address.

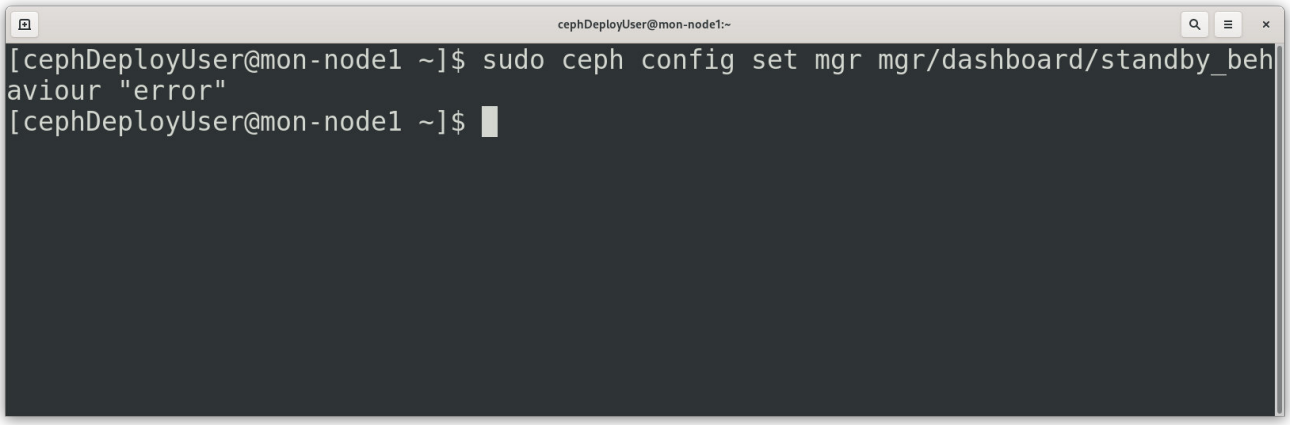
Two changes should be made to the dashboard module before attempting to load balance the service.

Disable the Redirection

The default behaviour of passive manager nodes is to send an HTTP 303 response status code when a client attempts to access the dashboard from them. When the dashboard service is being load balanced, there is a possibility that a connecting client could be sent an unresolvable URL redirect in the event that a fail over from one manager to another is in progress. As such, the default redirection behaviour should be disabled when the dashboard service is being load balanced.

From any Ceph node in the cluster, execute the following command (sudo privileges may be required):

```
ceph config set mgr mgr/dashboard/standby_behaviour "error"
```



Configure the Error Status Code

Once redirection has been disabled, passive manager nodes will send an HTTP 500 'internal server error' status code if a client attempts to access the dashboard from them. It is recommended by the Ceph project, and is good practice, to use the more descriptive HTTP 503 'service unavailable' status code.

From any Ceph node in the cluster, execute the following command (sudo privileges may be required):

```
ceph config set mgr mgr/dashboard/standby_error_status_code 503
```

```
cephDeployUser@mon-node1:~
[cephDeployUser@mon-node1 ~]$ sudo ceph config set mgr mgr/dashboard/standby_err
or_status_code 503
[cephDeployUser@mon-node1 ~]$
```

9. Loadbalancer.org Appliance – the Basics

9.1. Virtual Appliance

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, XEN and Nutanix AHV and has been optimized for each Hypervisor. By default, the VA is allocated 2 vCPUs, 4GB of RAM and has a 20GB 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 Virtual Appliance Installation and the ReadMe.txt text file included in the VA download for additional information on deploying the VA using the various Hypervisors.

Note	The VA has 4 network adapters. For VMware only the first adapter (**eth0**) is connected by default. For HyperV, KVM, XEN and Nutanix AHV all adapters are disconnected by default. Use the network configuration screen within the Hypervisor to connect the required adapters.

9.2. Initial Network Configuration

After boot up, follow the instructions on the appliance console to configure the management IP address, subnet mask, default gateway, DNS servers and other network and administrative settings.

Important	Be sure to set a secure password for the load balancer, when prompted during the setup routine.

9.3. Accessing the Appliance WebUI

The WebUI is accessed using a web browser. By default, users are authenticated using Apache authentication. Users can also be authenticated against LDAP, LDAPS, Active Directory or Radius - for more information, please refer to External Authentication.

> **Note** There are certain differences when accessing the WebUI for the cloud appliances. For details, please refer to the relevant Quick Start / Configuration Guide.

1. Using a browser, navigate to the following URL:

   https://<IP-address-configured-during-the-network-setup-wizard>:9443/lbadmin/

   > **Note** You'll receive a warning about the WebUI's SSL certificate. This is due to the default self signed certificate that is used. If preferred, you can upload your own certificate - for more information, please refer to Appliance Security Features.

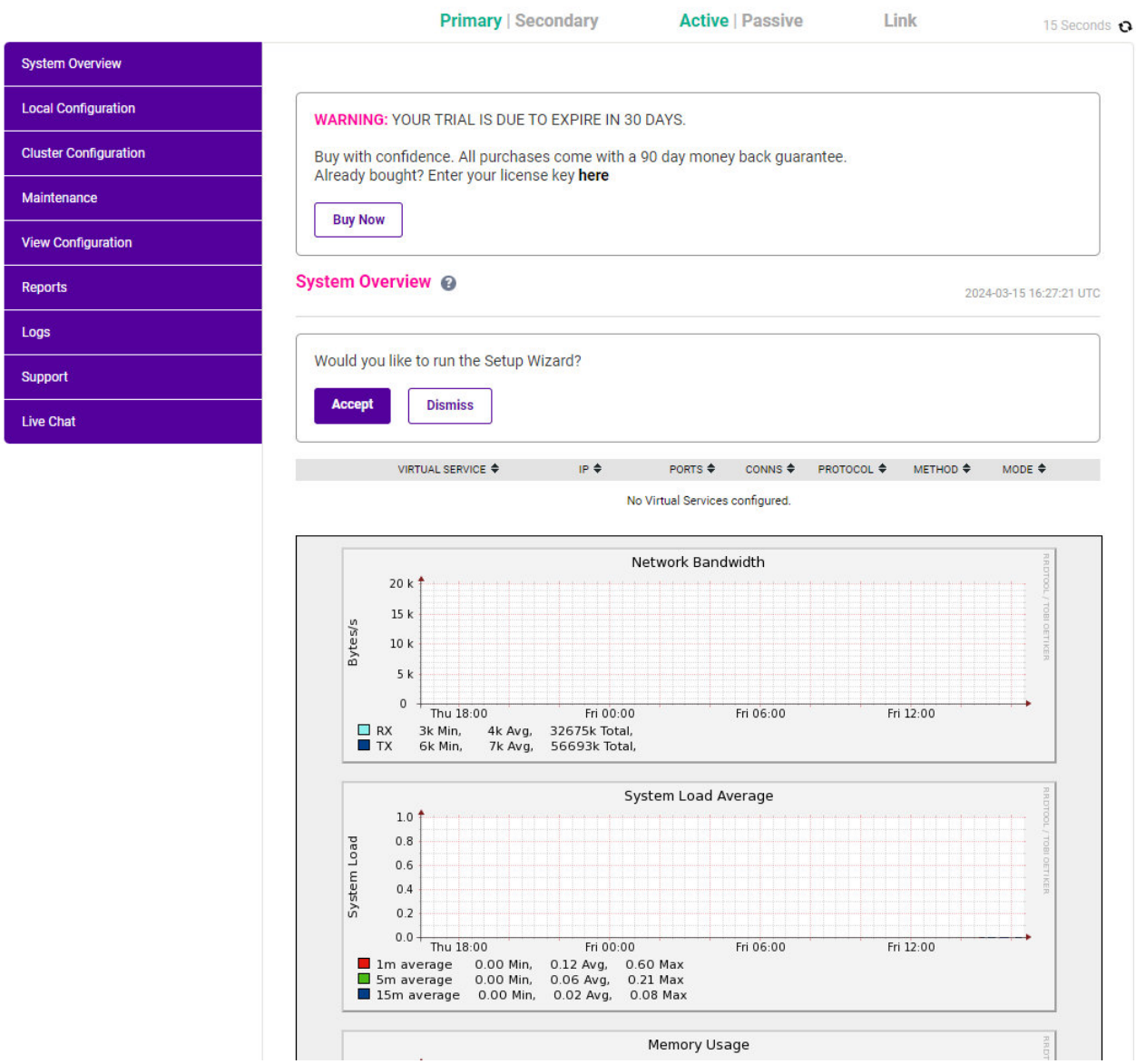
   > **Note** If you need to change the port, IP address or protocol that the WebUI listens on, please refer to Service Socket Addresses.

2. Log in to the WebUI using the following credentials:

   Username: loadbalancer
   Password: <configured-during-network-setup-wizard>

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

   Once logged in, the WebUI will be displayed as shown below:



3. You'll be asked if you want to run the Setup Wizard which can be used to configure layer 7 services. Click **Dismiss** if you're following a guide or want to configure the appliance manually or click **Accept** to start the wizard.

Main Menu Options

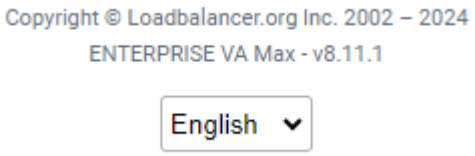
System Overview - Displays a graphical summary of all VIPs, RIPs and key appliance statistics
Local Configuration - Configure local host settings such as IP address, DNS, system time etc.
Cluster Configuration - Configure load balanced services such as VIPs & RIPs
Maintenance - Perform maintenance tasks such as service restarts and taking backups
View Configuration - Display the saved appliance configuration settings
Reports - View various appliance reports & graphs
Logs - View various appliance logs
Support - Create a support download, contact the support team & access useful links
Live Chat - Start a live chat session with one of our Support Engineers

9.4. Appliance Software Update

To ensure that the appliance(s) are running the latest software version, we recommend a software update check is performed.

Determining the Current Software Version

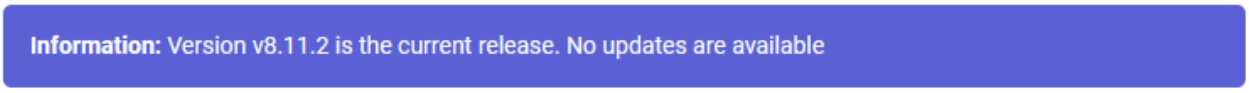
The software version is displayed at the bottom of the WebUI as shown in the example below:



Checking for Updates using Online Update

> **Note** By default, the appliance periodically contacts the Loadbalancer.org update server and checks for updates. An update check can also be manually triggered as detailed below.

1. Using the WebUI, navigate to: ***Maintenance > Software Update***.
2. Select **Online Update**.
3. If the latest version is already installed, a message similar to the following will be displayed:



4. If an update is available, you'll be presented with a list of new features, improvements, bug fixes and security related updates.
5. Click **Online Update** to start the update process.

> **Note** Do not navigate away whilst the update is ongoing, this may cause the update to fail.

6. Once complete (the update can take several minutes depending on download speed and upgrade version) the following message will be displayed:

7. If services need to be reloaded/restarted or the appliance needs a full restart, you'll be prompted accordingly.

Using Offline Update

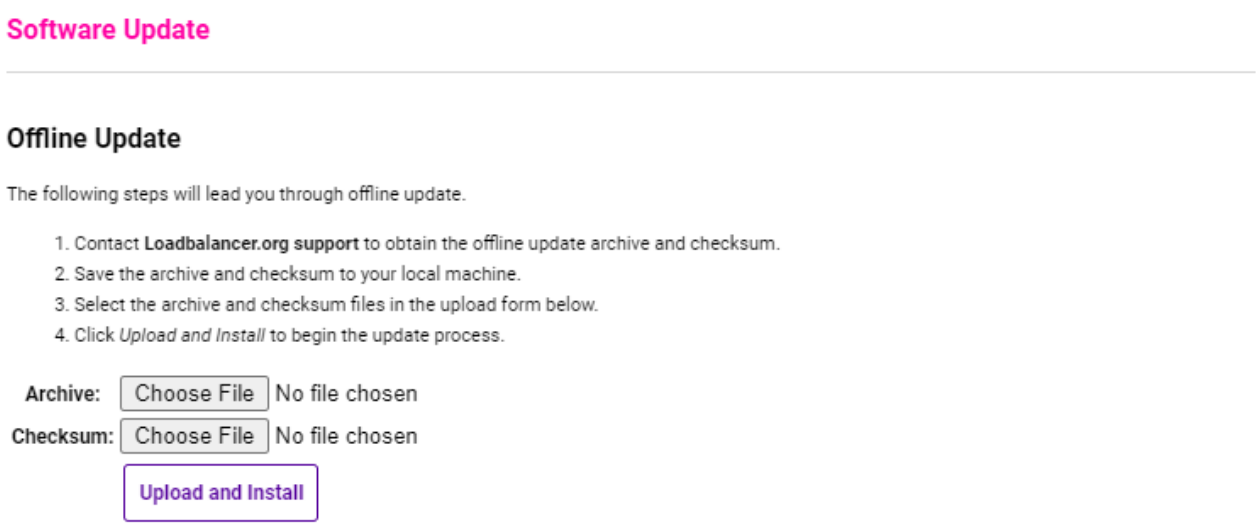
If the load balancer does not have access to the Internet, offline update can be used.

> **Note** Please contact support@loadbalancer.org to check if an update is available and obtain the latest

offline update files.

To perform an offline update:

1. Using the WebUI, navigate to: *Maintenance > Software Update*.
2. Select **Offline Update**.
3. The following screen will be displayed:



4. Select the *Archive* and *Checksum* files.
5. Click **Upload and Install**.
6. If services need to be reloaded/restarted or the appliance needs a full restart, you'll be prompted accordingly.

9.5. Ports Used by the Appliance

By default, the appliance uses the following TCP & UDP ports:

Protocol	Port	Purpose
TCP	22 *	SSH
TCP & UDP	53 *	DNS / GSLB
TCP & UDP	123	NTP
TCP & UDP	161 *	SNMP
UDP	6694	Heartbeat between Primary & Secondary appliances in HA mode
TCP	7778	HAProxy persistence table replication
TCP	9000 *	Gateway service (Centralized/Portal Management)
TCP	9080 *	WebUI - HTTP (disabled by default)
TCP	9081 *	Nginx fallback page
TCP	9443 *	WebUI - HTTPS

Protocol	Port	Purpose
TCP	25565 *	Shuttle service (Centralized/Portal Management)

> **Note** The ports used for SSH, GSLB, SNMP, the WebUI, the fallback page, the gateway service and the shuttle service can be changed if required. For more information, please refer to Service Socket Addresses.

9.6. 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 unit is covered in Configuring HA - Adding a Secondary Appliance.

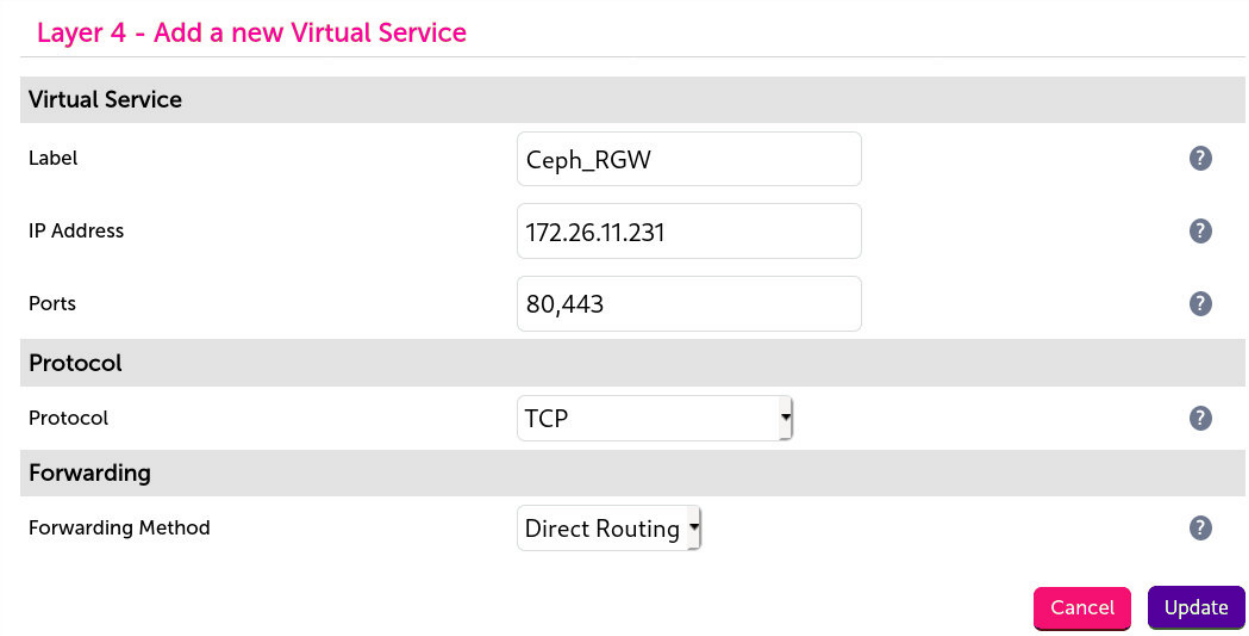
10. Appliance Configuration for Ceph Object Gateways – Using Layer 4 DR Mode (LVS/DR)

The following instructions assume that the steps in Configuring Ceph Object Gateways for Layer 4 DR Mode (LVS/DR) have already been followed.

10.1. Configuring VIP 1 – Ceph Object Gateways

Configuring the Virtual Service (VIP)

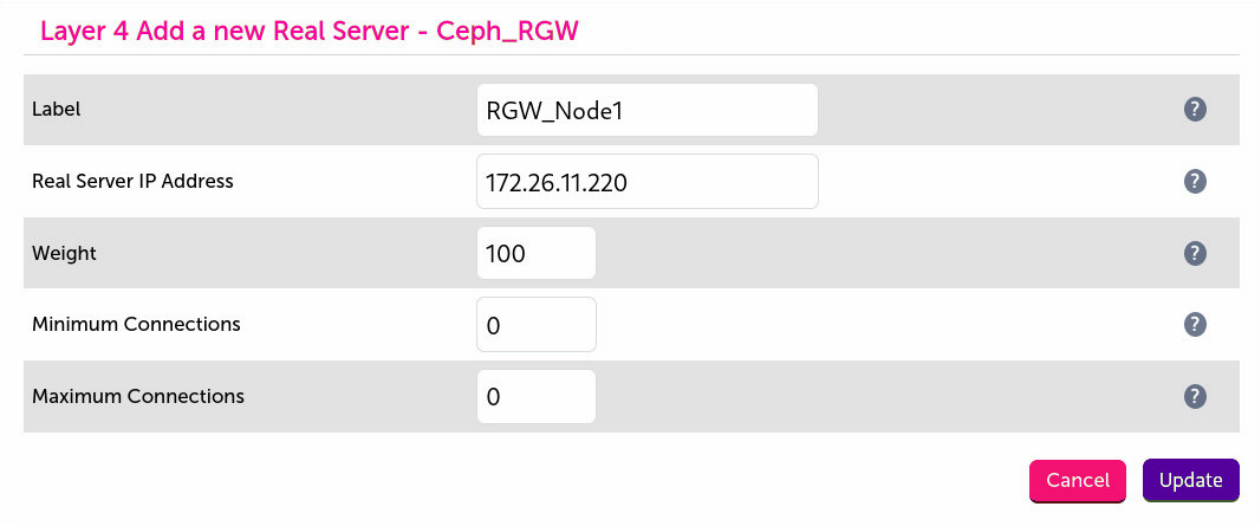
1. Using the web user interface, navigate to *Cluster Configuration > Layer 4 – Virtual Services* and click on **Add a new Virtual Service**.
2. Define the *Label* for the virtual service as required, e.g. **Ceph_RGW**.
3. Set the *Virtual Service IP Address* field to the required IP address, e.g. **172.26.11.231**.
4. Set the *Ports* field to the ports that are in use by the Ceph Object Gateways, e.g. **80,443**.
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. Set the *Balance Mode* to **Weighted Round Robin**.
10. Ensure that the *Persistence Enable* checkbox is not checked.
11. 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. **RGW_Node1**.
3. Set the *Real Server IP Address* field to the required IP address, e.g. **172.26.11.220**.
4. Click *Update*.
5. Repeat these steps to add the remaining Ceph Object Gateways.



10.2. Configuring VIP 2 – Ceph Dashboard

A layer 7 solution (which uses HAProxy) should be used for this optional virtual service. To configure it, follow the instructions in Configuring VIP 2 – Ceph Dashboard followed by the instructions that immediately follow it - Finalizing the Layer 7 Configuration.

11. Appliance Configuration for Ceph Object Gateways – Using Layer 7 SNAT Mode (HAProxy)

11.1. Enabling Multithreaded Load Balancing

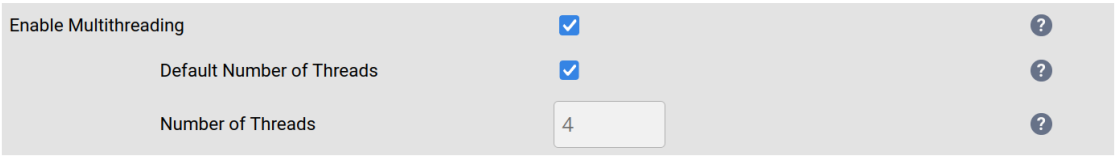
> **Note**
>
> Multithreading is enabled by default for ***new*** load balancers starting from version 8.5.1 and does not require changing.
>
> ***If upgrading an older appliance*** then ensure that the multithreading configuration is set correctly, as described below.

For the full layer 7 load balancing scenario (not necessary if using a layer 7 virtual service for the dashboard element only), 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 deployment of Ceph Object Gateways at layer 7.

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

To enable multithreaded mode from the WebUI:

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

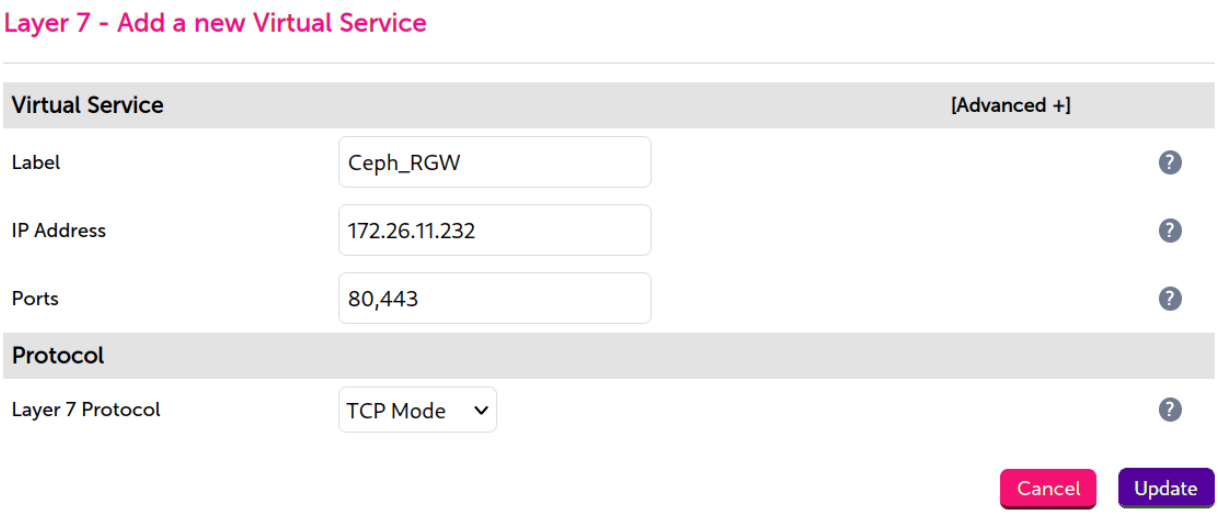


11.2. Configuring VIP 1 – Ceph Object Gateways

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. **Ceph_RGW**.
3. Set the *Virtual Service IP Address* field to the required IP address, e.g. **172.26.11.232**.

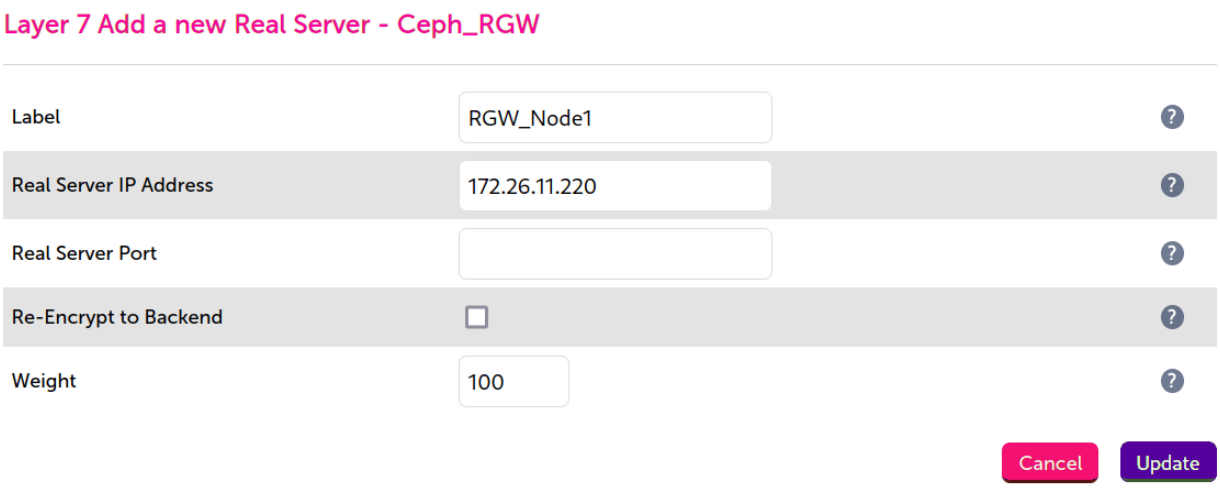
4. Set the *Ports* field to the ports that are in use by the Ceph Object Gateways, e.g. **80,443**.
5. Set the *Layer 7 Protocol* to **TCP Mode**.
6. Click **Update** to create the virtual service.



7. Click **Modify** next to the newly created VIP.
8. Set the *Balance Mode* to **Weighted Round Robin**.
9. Set *Persistence Mode* to **None**.
10. 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. **RGW_Node1**.
3. Set the *Real Server IP Address* field to the required IP address, e.g. **172.26.11.220**.
4. Click **Update**.
5. Repeat these steps to add the remaining Ceph Object Gateways.

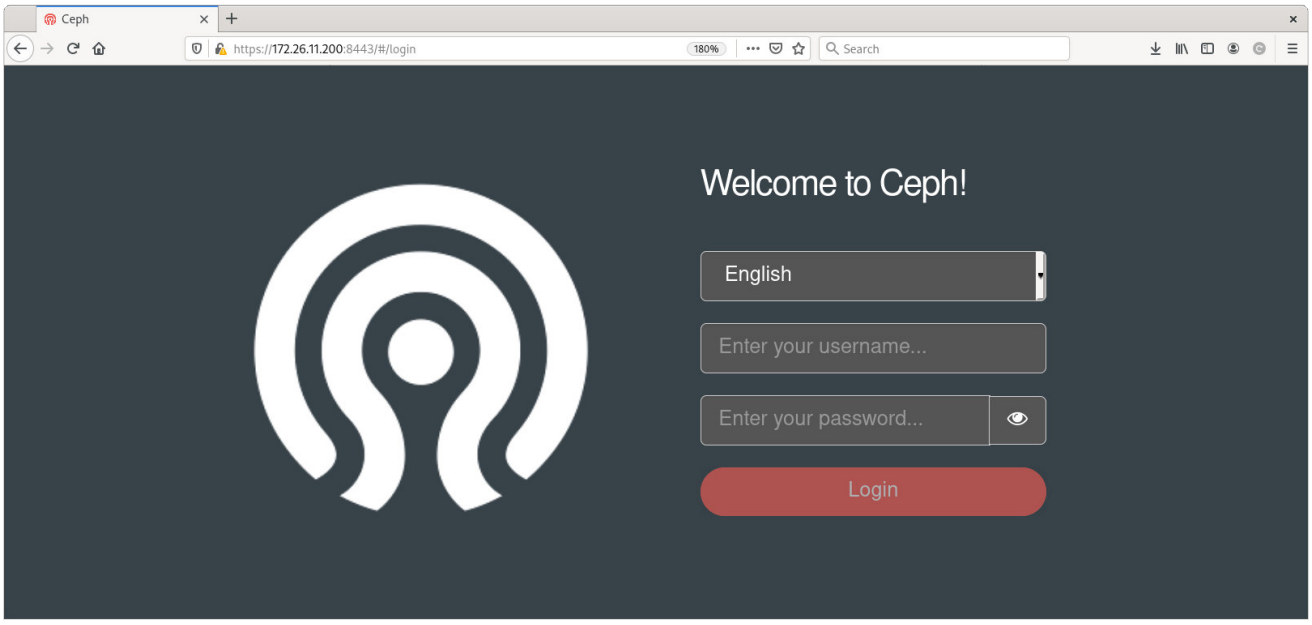


11.3. Configuring VIP 2 – Ceph Dashboard

The Ceph Dashboard module is an optional component of Ceph. In a deployment with multiple manager nodes, only one manager node is active at a given time. Only the active manager node hosts the web-based dashboard service.

If desired, an additional virtual service can be used on the load balancer to ensure that the Ceph Dashboard is highly available and always accessible from the same IP address.

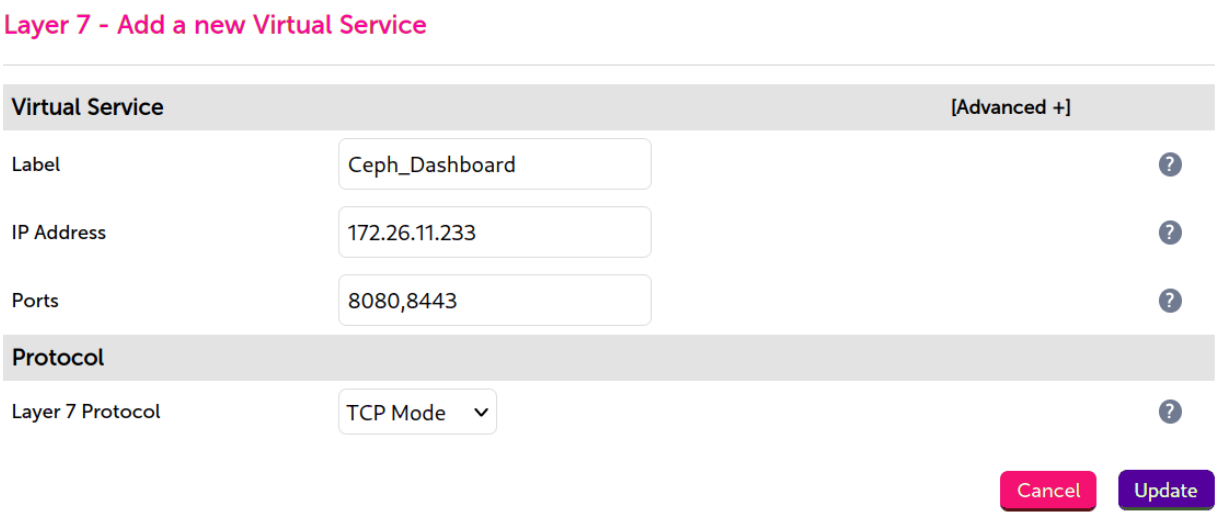
The dashboard should have been configured for load balancing before following the instructions presented below (please refer to Configuring the Dashboard Module for Load Balancing).



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. **Ceph_Dashboard**.

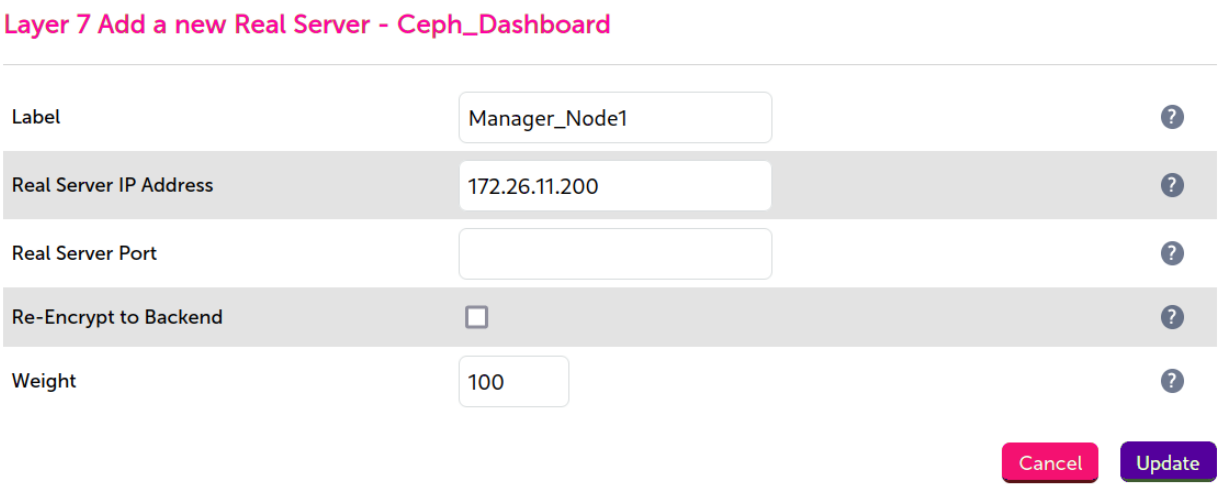
3. Set the *Virtual Service IP Address* field to the required IP address, e.g. **172.26.11.233**.
4. Set the *Ports* field to the ports that are in use by the Ceph Dashboard, e.g. **8080,8443**.
5. Set the *Layer 7 Protocol* to **TCP Mode**.
6. Click **Update** to create the virtual service.



7. Click **Modify** next to the newly created VIP.
8. In the *Health Checks* section click **Advanced** to expand the menu.
9. Set *Health Checks* as appropriate for the dashboard configuration. For example:
   - If the dashboard service is configured to use HTTP then a valid option is to select **Negotiate HTTP (GET)** and set the *Check Port* as required, e.g. **8080**.
   - If the dashboard service is configured to use HTTPS then a valid option is to select **Negotiate HTTPS (GET)** and set the *Check Port* as required, e.g. **8443**.
10. In the *Other* section click **Advanced** to expand the menu.
11. Check the **Timeout** checkbox.
12. Set *Client Timeout* to **50s** (for 50 seconds).
13. Set *Real Server Timeout* to **450s**.
14. 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. **Manager_Node1**.
3. Set the *Real Server IP Address* field to the required IP address, e.g. **172.26.11.200**.
4. Click **Update**.
5. Repeat these steps to add the remaining manager nodes.



> **Note** Unlike with most load balancing deployments, it is **required** to **add *all* manager nodes** to the dashboard virtual service. This is because ***any*** manager node could be the single active node at a given time. For example, if a manager node becomes the active node but is ***not*** defined under the dashboard virtual service then the virtual service will fail.

11.4. Finalizing the Layer 7 Configuration

To apply the new settings, HAProxy must be reloaded. This can be done using the button in the "Commit changes" box at the top of the screen or by using the *Restart Services* menu option:

1. Using the WebUI, navigate to: *Maintenance > Restart Services*.
2. Click **Reload HAProxy**.

12. Testing & Verification

> **Note** For additional guidance on diagnosing and resolving any issues you may have, please also refer to Diagnostics & Troubleshooting.

12.1. Using System Overview

The System Overview can be viewed in the WebUI. It shows a graphical view of all VIPs & RIPs (i.e. the Ceph nodes) and shows the state/health of each server as well as the state of the cluster as a whole.

The example below shows a **layer 4 DR mode** configuration load balancing a pair of Ceph Object Gateways, where both nodes are healthy and available to accept connections, and a layer 7 virtual service for the Ceph Dashboard, with the active manager node showing as available to accept connections:

System Overview

2020-03-16 15:25:46 UTC

VIRTUAL SERVICE	IP	PORTS	CONNS	PROTOCOL	METHOD	MODE
Ceph_RGW	172.26.11.231	80,443	0	TCP	Layer 4	DR
REAL SERVER	*IP*	*PORTS*	*WEIGHT*	*CONNS*		
RGW_Node1	*172.26.11.220*	*80,443*	*100*	*0*	*Drain*	*Halt*
RGW_Node2	*172.26.11.221*	*80,443*	*100*	*0*	*Drain*	*Halt*
Ceph_Dashboard	172.26.11.233	8080,8443	0	TCP	Layer 7	Proxy
REAL SERVER	*IP*	*PORTS*	*WEIGHT*	*CONNS*		
Manager_Node1	*172.26.11.200*	*8080,8443*	*100*	*0*	*Drain*	*Halt*
Manager_Node2	*172.26.11.201*	*8080,8443*	*100*	*0*	*Drain*	*Halt*
Manager_Node3	*172.26.11.202*	*8080,8443*	*100*	*0*	*Drain*	*Halt*

The example below shows a **layer 7 SNAT mode** configuration load balancing a pair of Ceph Object Gateways, where both nodes are healthy and available to accept connections, and a layer 7 virtual service for the Ceph Dashboard, with the active manager node showing as available to accept connections:

System Overview

2020-03-16 15:22:09 UTC

VIRTUAL SERVICE	IP	PORTS	CONNS	PROTOCOL	METHOD	MODE
Ceph_RGW	172.26.11.232	80,443	0	TCP	Layer 7	Proxy
REAL SERVER	*IP*	*PORTS*	*WEIGHT*	*CONNS*		
RGW_Node1	*172.26.11.220*	*80,443*	*100*	*0*	*Drain*	*Halt*
RGW_Node2	*172.26.11.221*	*80,443*	*100*	*0*	*Drain*	*Halt*
Ceph_Dashboard	172.26.11.233	8080,8443	0	TCP	Layer 7	Proxy
REAL SERVER	*IP*	*PORTS*	*WEIGHT*	*CONNS*		
Manager_Node1	*172.26.11.200*	*8080,8443*	*100*	*0*	*Drain*	*Halt*
Manager_Node2	*172.26.11.201*	*8080,8443*	*100*	*0*	*Drain*	*Halt*
Manager_Node3	*172.26.11.202*	*8080,8443*	*100*	*0*	*Drain*	*Halt*

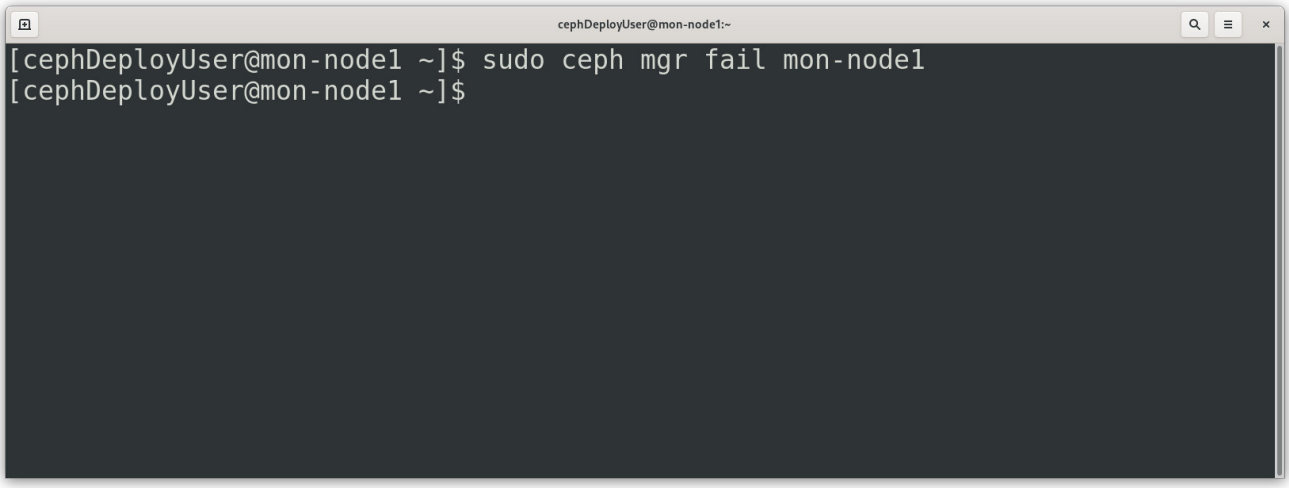
12.2. Testing Dashboard Fail Over

It is possible to issue a command to intentionally cause the active manager node to fail. This will cause one of the other manager nodes to become active and start serving the Ceph Dashboard service. From the load balancer's System Overview page, the active manager node should then turn 'red'/offline and another manager node should become 'green'/healthy.

From any Ceph node in the cluster, execute the following command to trigger a manager node failure (sudo privileges may be required):

```
ceph mgr fail <hostnameOfActiveManager>
```

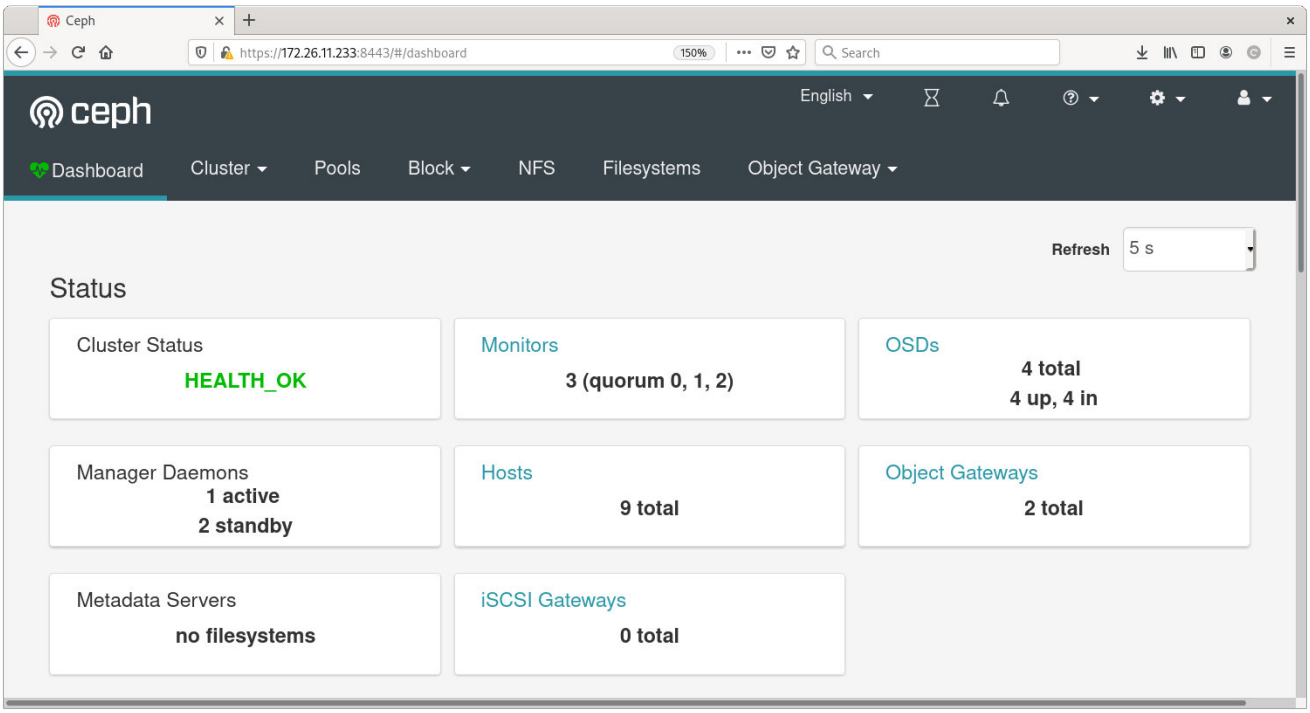
In the following example, the manager daemons all reside on monitor nodes. A failure is triggered on the active manager daemon, which resides on monitor node 1:



It is then observed that another manager node has become active:

Ceph_Dashboard	172.26.11.233	8080,8443	0	TCP	Layer 7	Proxy	
REAL SERVER	IP	PORTS	WEIGHT	CONNS			
Manager_Node1	172.26.11.200	8080,8443	100	0	Drain	Halt	
Manager_Node2	172.26.11.201	8080,8443	100	0	Drain	Halt	
Manager_Node3	172.26.11.202	8080,8443	100	0	Drain	Halt	

As a final test, the dashboard is successfully logged into using the VIP address, which is 172.26.11.233 in this example:



13. 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.

14. Further Documentation

For additional information, please refer to the Administration Manual.

15. Appendix

15.1. Legacy Support Option: Performing TLS/SSL Termination on the Load Balancer

Ceph versions prior to 11.0.1, Kraken, supported HTTP access to Ceph Object Gateways only. To use HTTPS on an older deployment which doesn't have native HTTPS support, a TLS/SSL termination can be configured on the load balancer. This allows clients to connect to the load balancer using HTTPS, and the load balancer can then send plain text HTTP to the Ceph Object Gateways.

The following instructions assume that the steps in Appliance Configuration for Ceph Object Gateways – Using Layer 7 SNAT Mode (HAProxy) have already been followed.

Uploading the Certificate

A self signed public certificate is included on the load balancer. This can be used for handling TLS/SSL connections, however, connecting clients will need to manually intervene to accept the self signed certificate when connecting to the load balanced HTTPS service. This is often acceptable for internal deployments. For public facing deployments, it is strongly advised to obtain a public certificate signed by a well known and trusted certificate authority.

Once obtained, the appropriate public certificate, including **both** the private key and public certificate parts, must be uploaded to the load balancer for TLS/SSL termination to work.

For detailed information on creating PEM certificate files and converting between certificate formats please refer to Creating a PEM File.

The process for uploading a certificate is as follows:

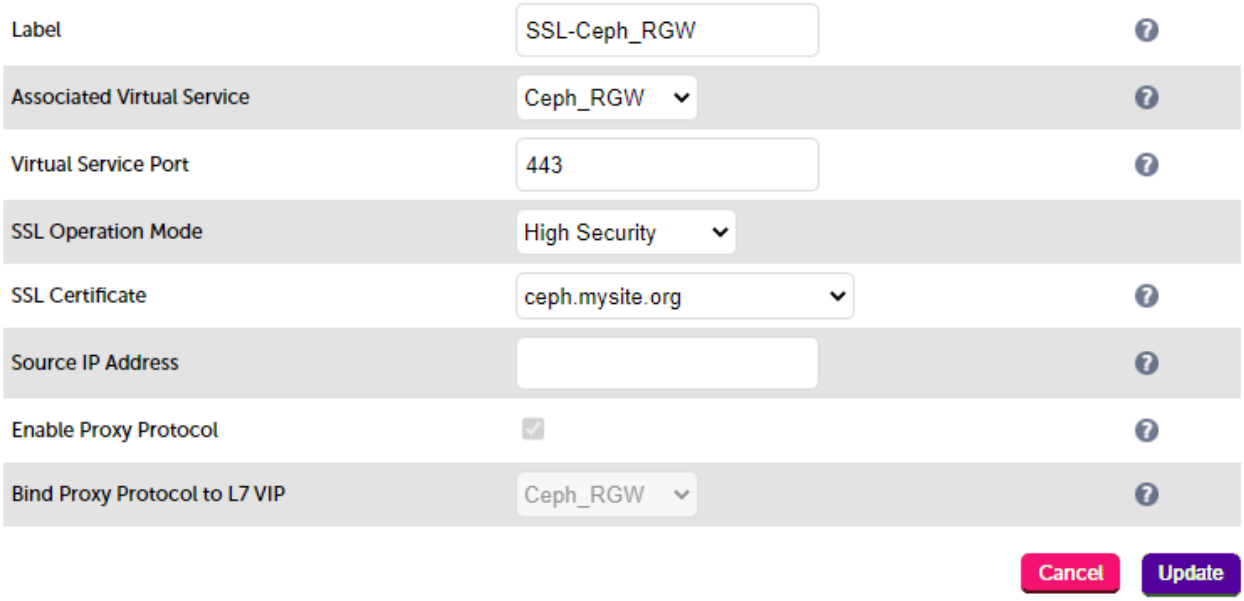
1. Using the web user interface, navigate to *Cluster Configuration > SSL Certificate* and click on **Add a new SSL Certificate**.
2. Press the *Upload prepared PEM/PFX file* radio button.
3. Define the *Label* for the certificate as required. It may make sense to use the domain that the certificate is associated to, e.g. **ceph.mysite.org**.
4. Click on **Browse** and select the appropriate PEM or PFX style certificate.
5. If uploading a PFX certificate, enter the certificate's password in the *PFX File Password* field.
6. Click **Upload certificate**.

Modifying the Existing Ceph Object Gateways Virtual Service

1. Using the web user interface, navigate to *Cluster Configuration > Layer 7 – Virtual Services* and click **Modify** next to the Ceph Object Gateways virtual service (*Ceph_RGW* in the example presented in this document).
2. Set the *Ports* field to the port that is in use by the Ceph Object Gateways for the HTTP service, e.g. **80**.
3. Click **Update**.

Creating the TLS/SSL Termination

1. Using the web user interface, navigate to *Cluster Configuration > SSL Termination* and click on **Add a new Virtual Service**.
2. From the *Associated Virtual Service* drop-down list, select the Ceph Object Gateways service that was created previously, e.g. **Ceph_RGW**.
3. Set the *Virtual Service Port* field to the port that will be used for the HTTPS service, e.g. **443**.
4. If using an uploaded public certificate, i.e. not the default self signed certificate, from the *SSL Certificate* drop-down list, select the certificate for the service in question, which in this example is **ceph.mysite.org**.
5. Click **Update** to create the TLS/SSL termination service.



Finalizing the Configuration

To apply the new settings, HAProxy and STunnel must both be reloaded. This can be done using the buttons in the "Commit changes" box at the top of the screen or by using the *Restart Services* menu option:

1. Using the WebUI, navigate to: *Maintenance > Restart Services*.
2. Click **Reload HAProxy**.
3. Click **Reload STunnel**.

15.2. Configuring GSLB / Location Affinity

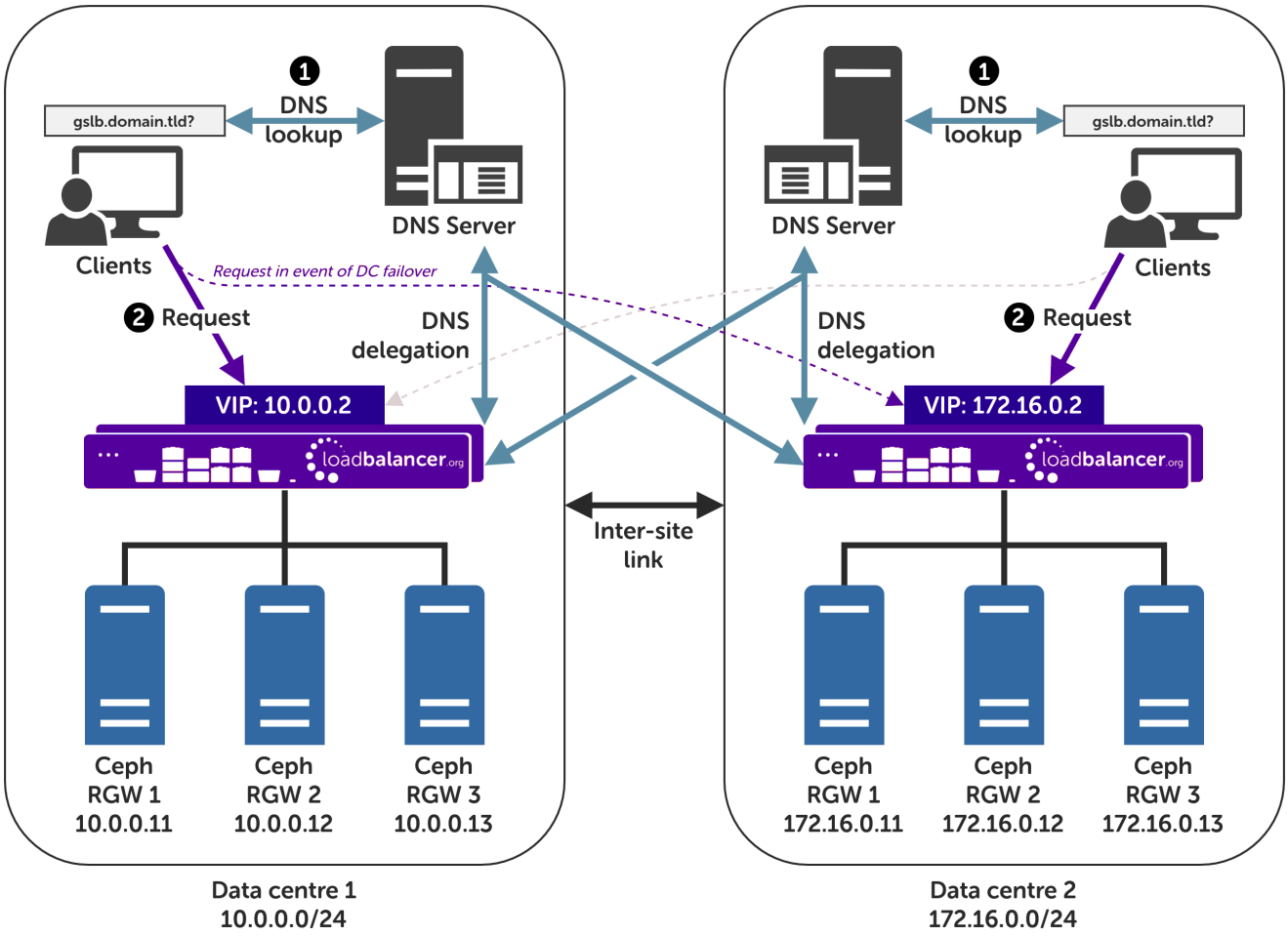
Conceptual Overview

For **multi-site Ceph deployments**, it is possible to use the load balancer's global server load balancing (GSLB) functionality to provide both high availability and location affinity across multiple sites.

- Clients across multiple sites use the same fully qualified domain name to access the Ceph Object Gateway service.
- **Under normal operation**: clients are directed to their local site's cluster of Ceph Object Gateways.

- **In the event of a local service failure**: clients are automatically directed to a functioning cluster of Ceph Object Gateways at another site. This would happen if the local site's Ceph Object Gateway cluster and/or load balancers were offline and unavailable.

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 Ceph Object Gateway service using the S3 protocol. To do this, the client uses the service's fully qualified domain name, in this example `gslb.domain.tld`
- The client sends a DNS query for `gslb.domain.tld` to its local DNS server.
- The DNS server has the domain `gslb.domain.tld` delegated to the load balancers.
- The DNS server sends a delegated DNS query for `gslb.domain.tld` to one of the load balancers.
- The load balancer that received the delegated DNS query replies to the DNS server. The load balancer answers with the IP address of the VIP (Ceph Object Gateway service) that is **local to the DNS server making the query**, and hence local to the original client.
  - An example: if the delegated query from the DNS server originated from the 10.0.0.0/24 subnet then the VIP in that subnet is served up. Likewise, if the delegated 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 Ceph Object Gateway cluster, provided that the local 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 `gslb.domain.tld` by using the local VIP address.

> **Note**
>
> **In the event that the cluster of Ceph Object Gateways and/or load balancers at one site should completely fail** then local clients will be directed to the cluster of Ceph Object Gateways 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.

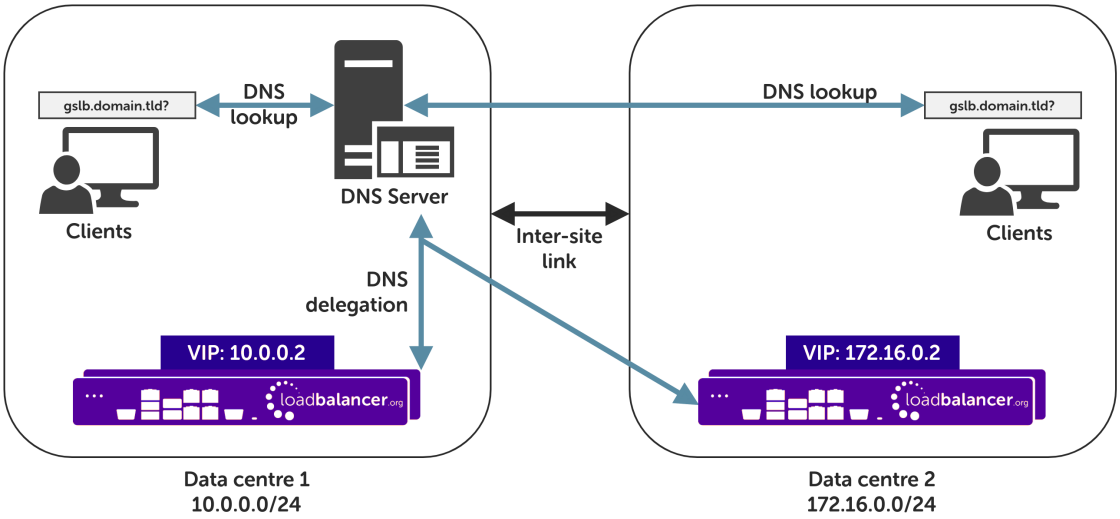
DNS Server Prerequisites

> **Important**
>
> Location affinity (ensuring clients 'stick' to their local site) **requires** a unique DNS server at each site.

For this setup to work and provide location affinity, a unique DNS server is required at each site, like the example deployment shown at the beginning of this section.

If multiple sites ***share*** a common DNS server then ***clients cannot be directed to their local, on-site Ceph Object Gateway cluster***.

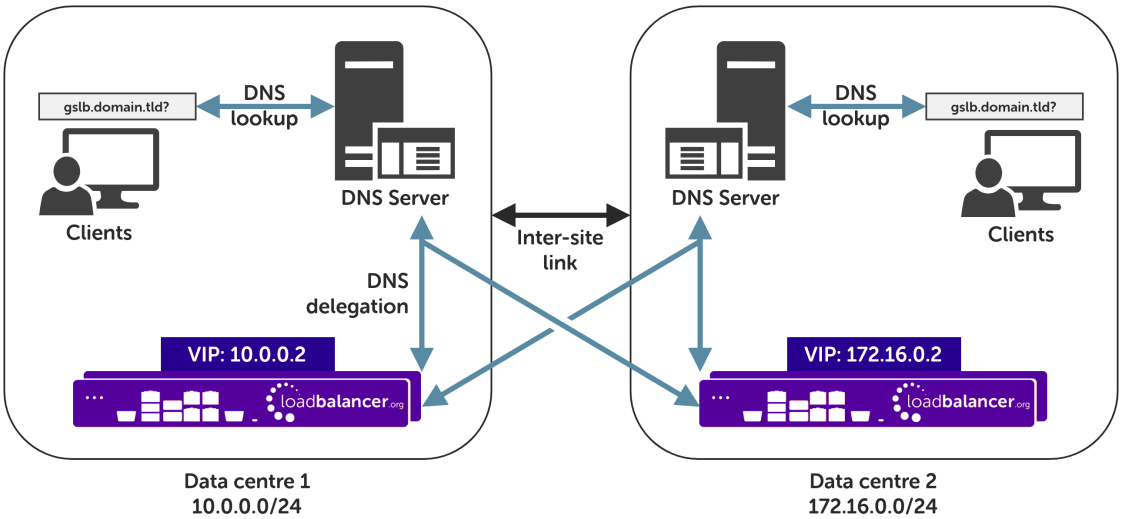
Example: Consider a two data centre deployment with a shared, common DNS server located at DC 1. From the perspective of a load balancer in this scenario, ***every*** delegated DNS request would be seen to come from the single, shared DNS server at DC 1. Specifically, the requests would all come from the DNS server's IP address, which would fall within DC 1's subnet.



A load balancer would have ***no way to distinguish between delegated requests for DC 1's clients and delegated requests for DC 2's clients***. All delegated requests would originate from within DC 1's subnet, therefore **all traffic would be directed to DC 1's Ceph Object Gateway cluster**.

To resolve such a situation, a DNS server would need to be deployed at DC 2. The load balancers could then

easily tell which site a given delegated DNS query has come from and, therefore, which site the client should be directed to.



If having unique DNS servers per-site and splitting up sites using a topology configuration is ***not*** possible then clients **will** bounce between different VIPs (and hence bounce between sites) in a round-robin fashion. If this behaviour is acceptable then it can theoretically be used without significant issue.

Handling Multiple Subdomains, Including Wildcard Subdomains

Scenario

Object storage-related DNS configurations may use various DNS subdomains, for example:

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

Some scenarios also require the use of wildcard DNS entries, for example to cover bucket specific subdomains like `app-instance-f57ac0.s3-region1.domain.tld`.

Solution

Configuring DNS delegation can be complex. As such, the supported solution is to:

- Delegate a single subdomain to the load balancer, e.g. `gslb.`
- Use CNAME records to point everything else at the delegated subdomain

For example, the subdomain `gslb.domain.tld` would be delegated and everything else would point to it. This would look like so:

`gslb.`	Delegate to the load balancer
`s3-<region>.`	CNAME to `gslb.domain.tld`
`*.s3-<region>.`	CNAME to `gslb.domain.tld`
`s3-admin-console.`	CNAME to `gslb.domain.tld`

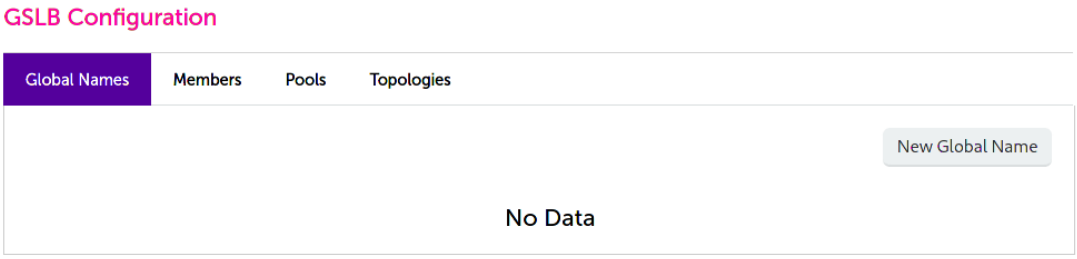
This approach simplifies DNS entry configuration, particularly when wildcard entries are involved.

Appliance Configuration

The GSLB service should be configured on the **primary** load balancer appliance at each site.

Note that **the GSLB configuration must be identical across all sites**: inconsistent configurations will lead to unexpected behaviour.

Configuration takes place in the WebUI under *Cluster Configuration > GSLB Configuration*:

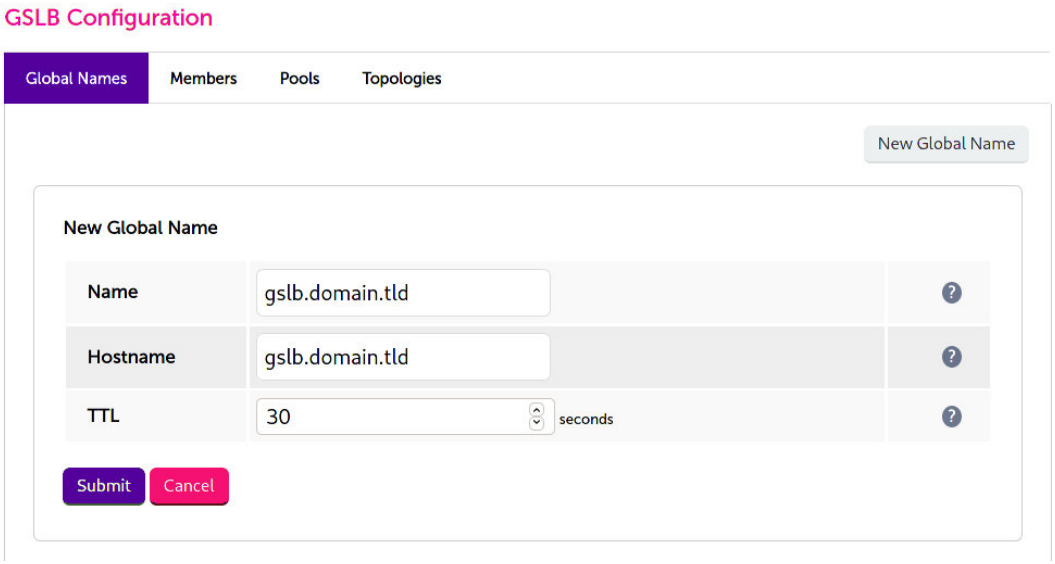


Step 1 – Configuring the Global Name

1. Using the WebUI on the primary appliance for the first site, navigate to *Cluster Configuration > GSLB Configuration*.
2. Select the **Global Names** tab.
3. Click the **New Global Name** button.
4. Define a friendly *Name* for the new hostname, which can just be the subdomain itself, e.g. **gslb.domain.tld**

> **Note** If only working with a *single* subdomain then it's perfectly acceptable to directly delegate the specific subdomain in question, e.g. `s3-region1.domain.tld`, rather than delegating a generic subdomain like `gslb.domain.tld`.

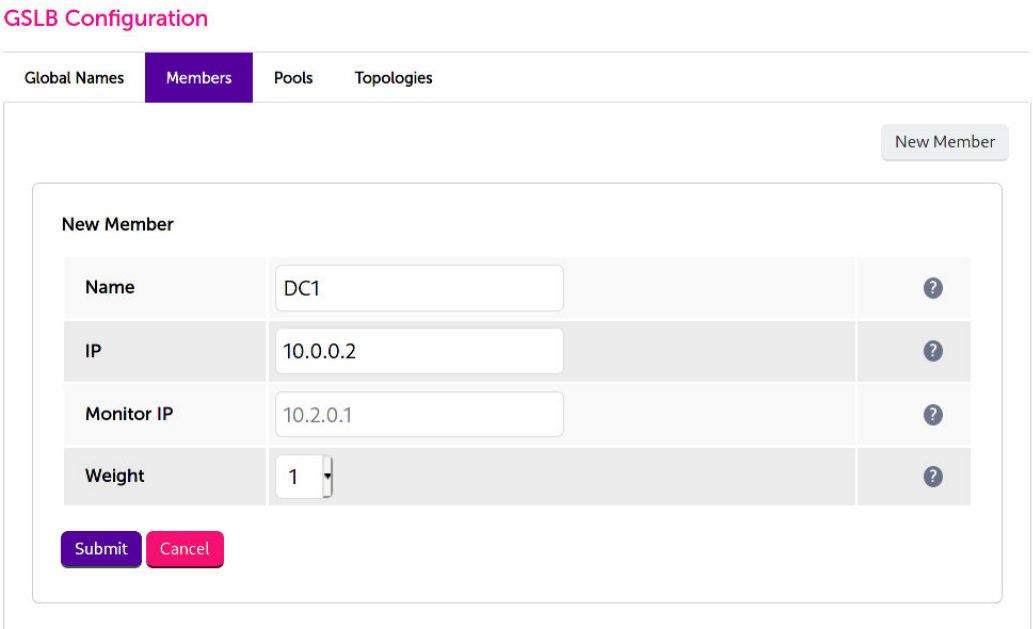
5. Define the *Hostname* of what will be the delegated subdomain, e.g. **gslb.domain.tld**
6. Click **Submit**.



Step 2 – Configure the Members

Each *member* can be thought of as a single site.

1. Select the **Members** tab.
2. Click the **New Member** button.
3. Enter a friendly *Name* for the member, e.g. **DC1**.
4. Specify an *IP* address for the member: in this context, this should be the VIP address of the site's Ceph Object Gateway service, e.g. **10.0.0.2**.
5. Ignore the example value in the *Monitor IP* field.
6. Click **Submit**.
7. Repeat these steps to add additional sites as members as required.

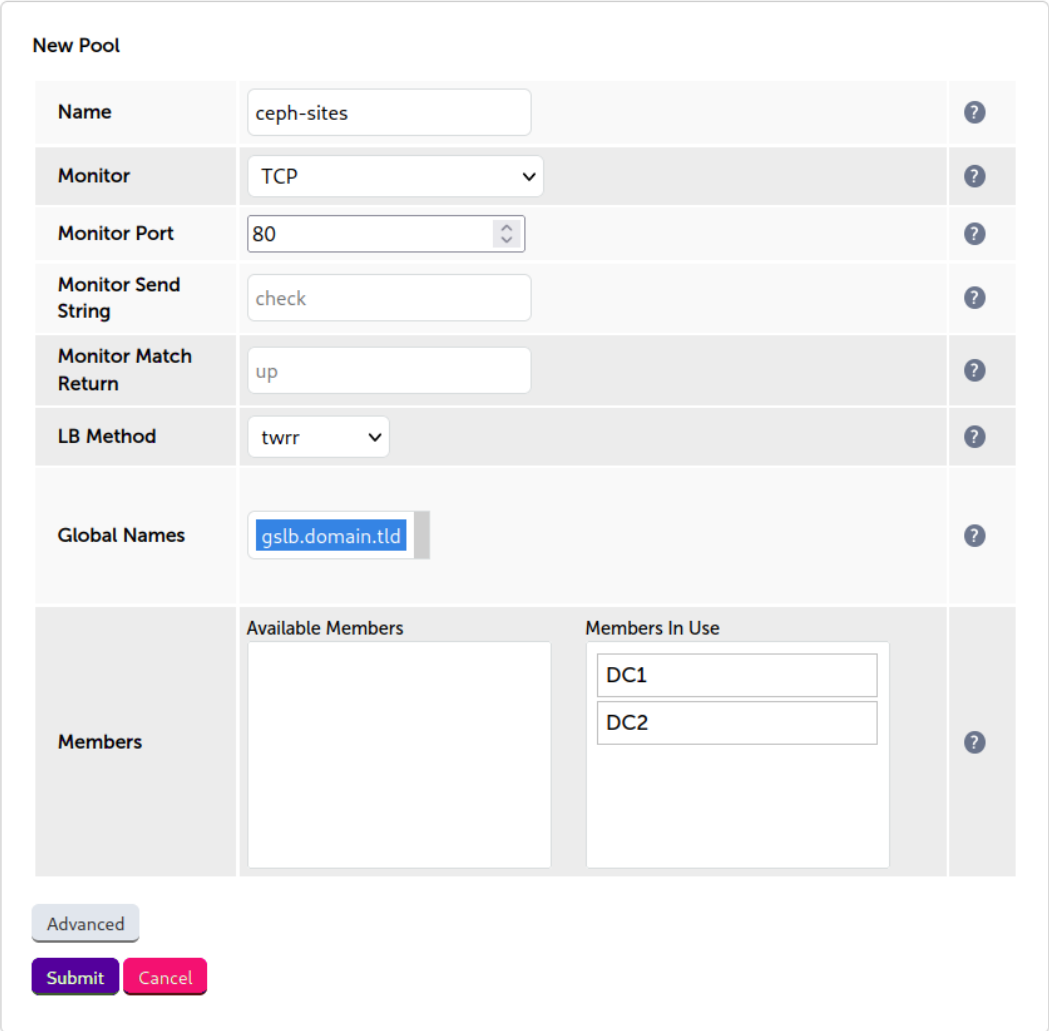


Step 3 – Configure the Pool

A pool must be created to link together a global name with the members that should serve traffic for that global name.

Continuing with the example presented in this section, both sites have a functional Ceph Object Gateway cluster ready for use. A pool would therefore be created linking the global name `gslb.domain.tld` with members (sites) DC1 and DC2, both of which should serve Ceph Object Gateway traffic.

1. Select the **Pools** tab.
2. Click the **New Pool** button.
3. Enter a friendly *Name* for the pool, e.g. **ceph-sites**.
4. Set the *Monitor* to **TCP**.
5. Set the *Monitor Port* to **80**.
6. Set *LB Method* to **twrr**.
7. From the *Global Names* list box, select the global name in question, e.g. **gslb.domain.tld**
8. In the *Members* section, drag the appropriate members (sites) from the *Available Members* box into the *Members In Use* box.
9. Click **Submit**.



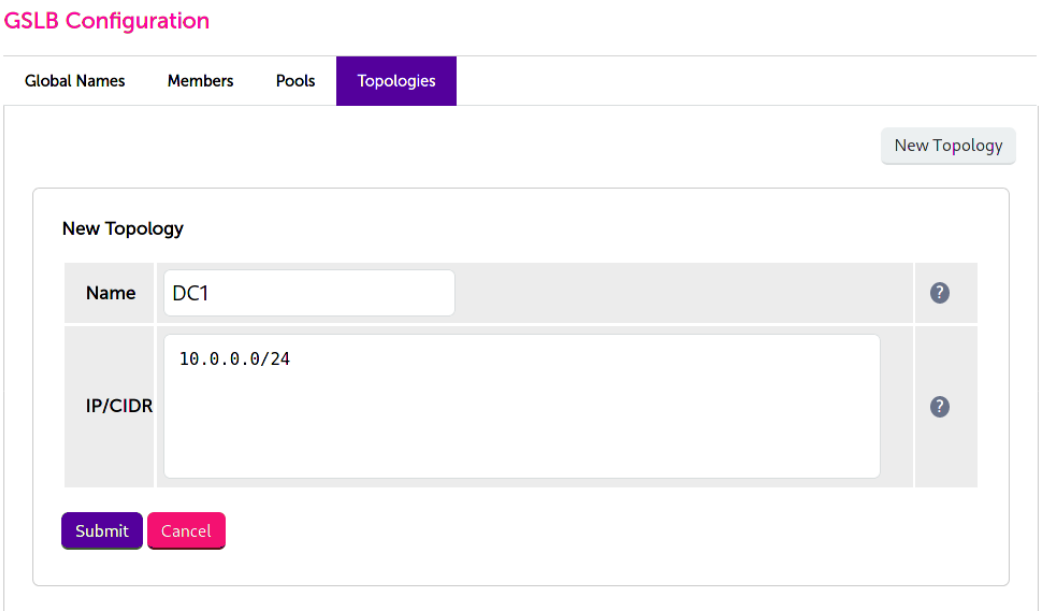
Step 4 – Configure the Topology

Topology configuration is used to map subnets to sites. This gives the solution its location awareness, allowing clients to be directed to their *local* Ceph Object Gateway instance instead of being bounced between every site which has been defined.

1. Select the **Topologies** tab.
2. Click the **New Topology** button.
3. Enter a friendly *Name* for the topology, e.g. **DC1**.
4. In the *IP/CIDR* text box, define the subnet(s) that covers the site in question, e.g. **10.0.0.0/24**.

   This can be a comma separated list of subnets and hosts, e.g. `10.0.0.0/24, 192.168.2.0/24, 192.168.17.57`. The key is that the site's DNS server *and* its Ceph Object Gateway VIP fall within the union of all subnets and hosts defined for the site. This is what allows DNS queries originating from the site to be matched up with that site's local VIP: the local VIP is then served as a DNS response for clients at that site.

5. Click **Submit**.
6. Repeat these steps to add additional topology configurations as required.



Step 5 – Finalising the Configuration

To apply the new settings, the GSLB service must be restarted as follows:

1. Using the WebUI, navigate to: *Maintenance > Restart Services* and click **Restart GSLB**.

Optional: Defining a Default Site for External Traffic (Handling DNS Requests from Unpredictable Source Addresses)

It is plausible that a Ceph Object Gateway GSLB deployment may be required to answer DNS queries sourced from outside of the subnets defined in the topology configuration.

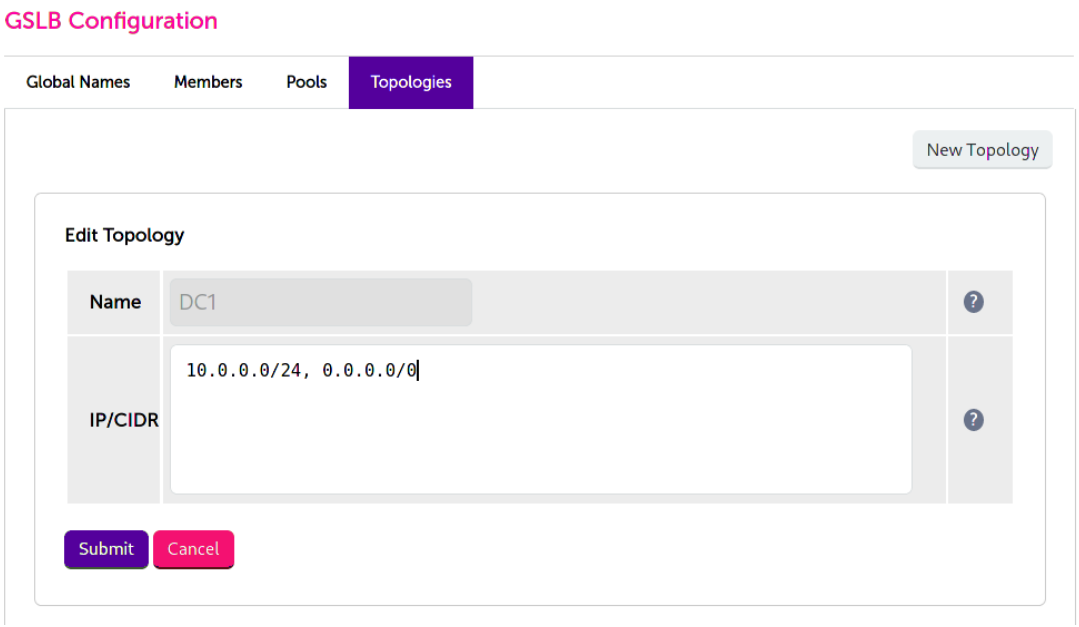
Consider a client on the public internet requesting a resource from the Ceph Object Gateway cluster. The DNS

query associated with the request may be sourced from a previously unseen, unpredictable public IP address. DNS queries from IP addresses that do not fall within the predefined network topology/subnets will be answered with DNS records pointing to ***any*** of the defined sites in a round-robin fashion.

An alternative is to define a ***default site***. All DNS queries from outside the predefined network topology will be answered with ***the same*** DNS record: a record pointing to the default site.

To configure this, add the widest possible subnet of 0.0.0.0/0 to the topology configuration of the site which is to be the 'default'. Any DNS query whose source IP address does not fall within one of the other, smaller subnets will be picked up by this new "catch all" subnet.

Following on from the previous example, setting data centre 1 to be the 'default' site would look like so:



DNS Server Configuration

Once the GSLB service has been configured on the primary load balancer at every site, 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 section, the DNS server at each site would be configured with a delegation for the domain `gslb.domain.tld`. The domain would be delegated to every load balancer across every site, which provides multi-site redundancy.

Steps walking through creating a DNS delegation on a Microsoft DNS server in the context of setting up GSLB on our appliance can be found in the appendix, in the section Microsoft DNS Server Configuration.

15.3. Microsoft DNS Server Configuration

Once the GSLB service has been fully configured on the primary load balancer at every site, as described in the previous sections, the DNS server at each site must be configured for GSLB.

The DNS server at each site must be configured to delegate DNS requests for the subdomain in question to the

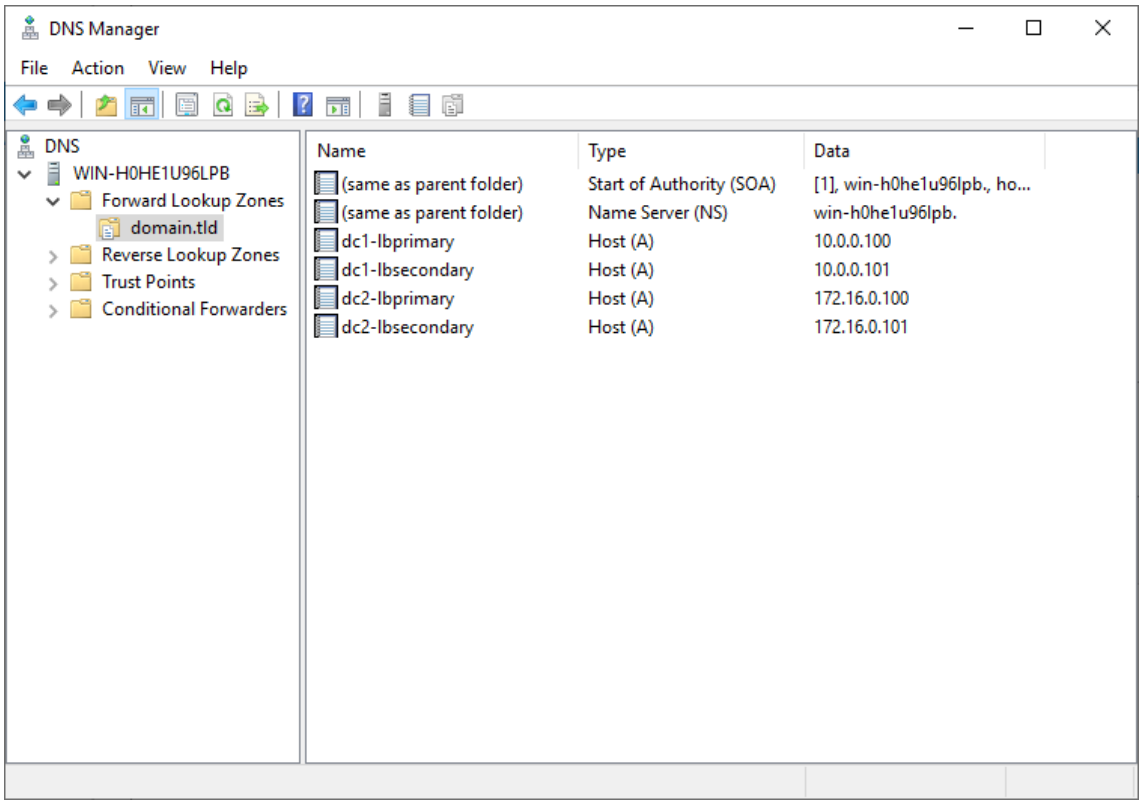
load balancers; the load balancers' GSLB services will serve the appropriate IP addresses to the DNS servers. Using the example presented throughout this document, the DNS server at each site would be configured with a delegation for the domain `gslb.domain.tld`. The domain would be delegated to every load balancer across every site, which provides multi-site redundancy.

The exact steps for creating a DNS delegation vary between different DNS servers. Presented below are steps that walk through creating a DNS delegation on a Microsoft DNS server in the context of setting up GSLB on our appliance.

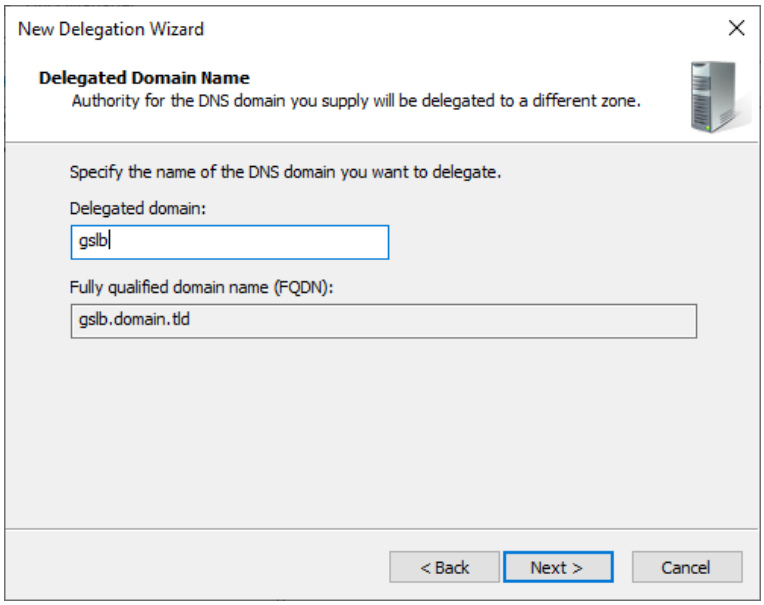
Microsoft DNS Server

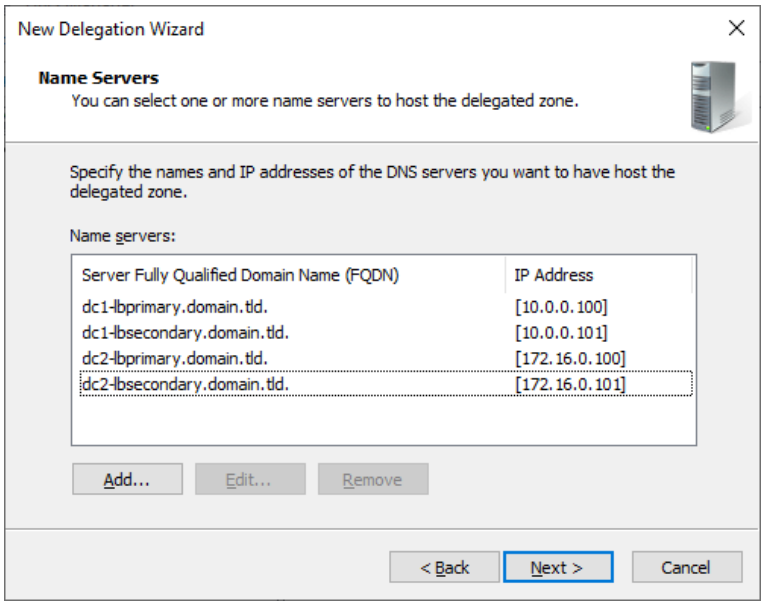
Delegating a subdomain in Microsoft DNS Manager is a short process.

1. Open **DNS Manager** and create A records for every load balancer at every site, using *Action > New Host* (e.g. `dc1-lbprimary.domain.tld`, `dc1-lbsecondary.domain.tld`, `dc2-lbprimary.domain.tld`, and `dc2-lbsecondary`).



2. Provided that the load balancer part of the GSLB configuration has been completed and is working, the **New Delegation** wizard should now be used to delegate the subdomain to the load balancers. The delegation will use the new FQDNs for the load balancers, as defined in the previous step. The delegation wizard is located at *Action > New Delegation*.





3. Test the delegation to make sure it is working as expected.

   From the Windows command line, the `nslookup` program can be used to send test DNS queries to the DNS server. The DNS server is located at IP address 10.0.0.50 in the example presented here.

   For the first test, use the `-norecurse` option to instruct the DNS server **not** to query another server for the answer. A successful test would see the DNS server respond and indicate that the subdomain in question is served by another server(s), giving the other server's details, like so:

```
C:\Users\me>nslookup -norecurse gslb.domain.tld 10.0.0.50
Server: UnKnown
Address: 10.0.0.50

Name:   gslb.domain.tld
Served by:
- dc1-lbprimary.domain.tld
          10.0.0.100
          gslb.domain.tld
```

```
- dc1-lbsecondary.domain.tld
          10.0.0.101
          gslb.domain.tld
- dc2-lbprimary.domain.tld
          172.16.0.100
          gslb.domain.tld
- dc2-lbsecondary.domain.tld
          172.16.0.101
          gslb.domain.tld
```

For the second test, execute the same command **without** the `-norecurse` option. This should see the DNS server fetch the answer from the load balancer and then serve up the 'fetched' answer in its response. A successful test would see the server reply with the IP address of one of the online sites/services, like so:

```
C:\Users\me>nslookup gslb.domain.tld 10.0.0.50
Server: UnKnown
Address: 10.0.0.50

Non-authoritative answer:
Name:   gslb.domain.tld
Address: 10.0.0.2
```

15.4. Configuring HA - Adding a Secondary Appliance

Our recommended configuration is to use a clustered HA pair of load balancers to provide a highly available and resilient load balancing solution. We recommend that the Primary appliance is fully configured first, then the Secondary appliance can be added to create an HA pair. Once the HA pair is configured, load balanced services must be configured and modified on the Primary appliance. The Secondary appliance will be automatically kept in sync.

> **Note** For Enterprise Azure, the HA pair should be configured first. For more information, please refer to the Azure Quick Start/Configuration Guide available in the documentation library

The clustered HA pair uses Heartbeat to determine the state of the other appliance. Should the active device (normally the Primary) suffer a failure, the passive device (normally the Secondary) will take over.

Non-Replicated Settings

A number of settings are not replicated as part of the Primary/Secondary pairing process and therefore must be manually configured on the Secondary appliance. These are listed by WebUI menu option in the table below:

WebUI Main Menu Option	Sub Menu Option	Description
Local Configuration	Hostname & DNS	Hostname and DNS settings
Local Configuration	Network Interface Configuration	Interface IP addresses, bonding configuration and VLANs
Local Configuration	Routing	Default gateways and static routes

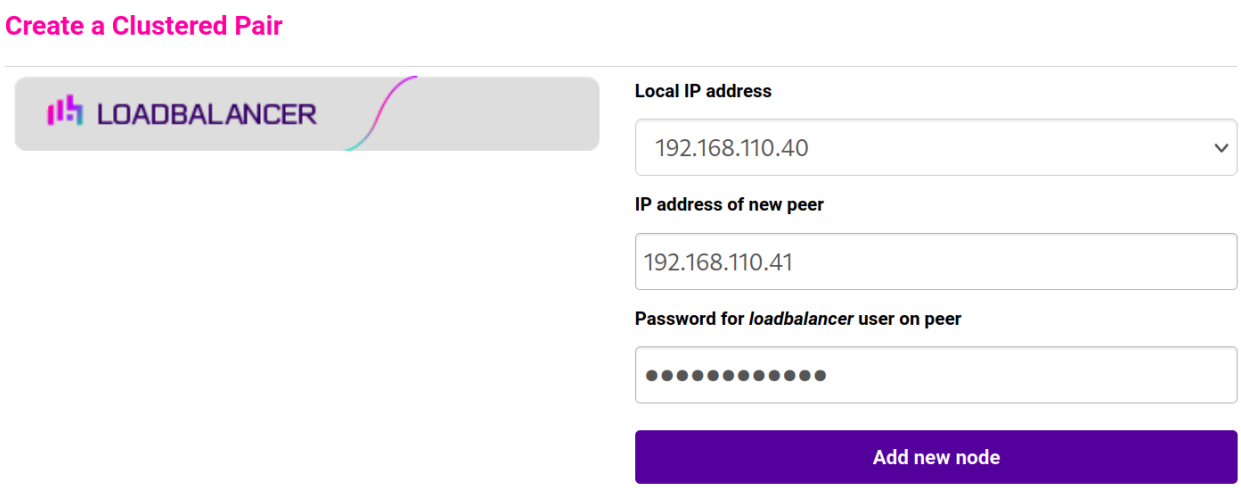
WebUI Main Menu Option	Sub Menu Option	Description
Local Configuration	System Date & time	Time and date related settings
Local Configuration	Physical – Advanced Configuration	Various appliance settings
Local Configuration	Portal Management	Portal management settings
Local Configuration	Security	Security settings
Local Configuration	SNMP Configuration	SNMP settings
Local Configuration	Graphing	Graphing settings
Local Configuration	License Key	Appliance licensing
Maintenance	Backup & Restore	Local XML backups
Maintenance	Software Updates	Appliance software updates
Maintenance	Fallback Page	Fallback page configuration
Maintenance	Firewall Script	Firewall (iptables) configuration
Maintenance	Firewall Lockdown Wizard	Appliance management lockdown settings

Important Make sure that where any of the above have been configured on the Primary appliance, they're also configured on the Secondary.

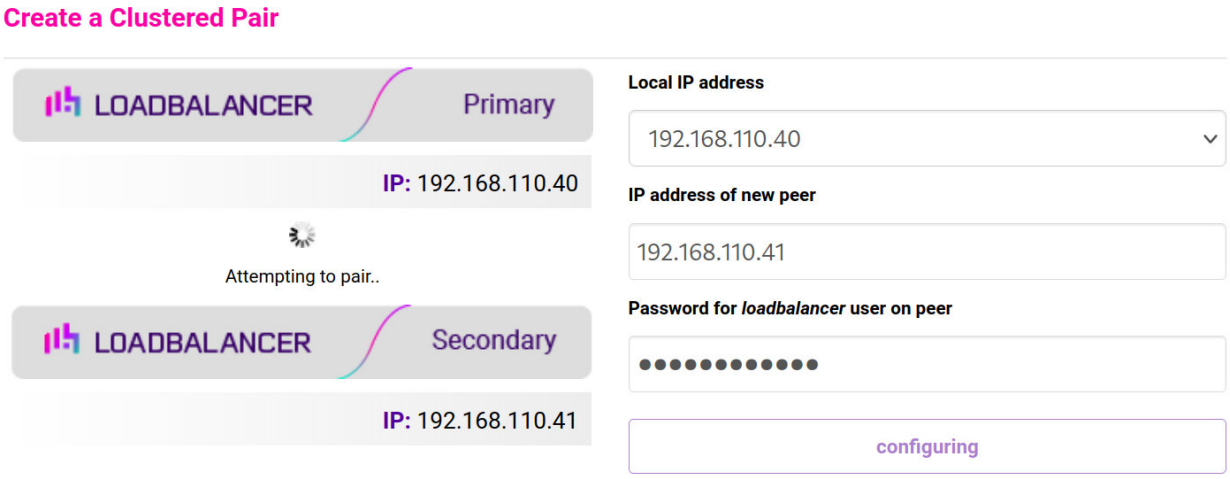
Configuring the HA Clustered Pair

Note If you have already run the firewall lockdown wizard on either appliance, you'll need to ensure that it is temporarily disabled on both appliances whilst performing the pairing process.

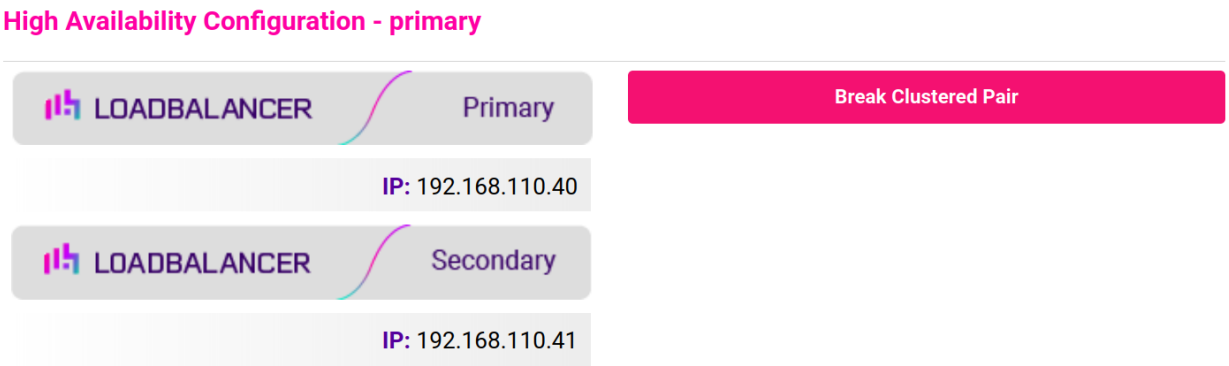
1. Deploy a second appliance that will be the Secondary and configure initial network settings.
2. Using the WebUI on the Primary appliance, navigate to: *Cluster Configuration > High-Availability Configuration*.



3. Specify the IP address and the *loadbalancer* user's password for the Secondary (peer) appliance as shown in the example above.
4. Click **Add new node**.
5. The pairing process now commences as shown below:



6. Once complete, the following will be displayed on the Primary appliance:



7. To finalize the configuration, restart heartbeat and any other services as prompted in the "Commit changes" message box at the top of the screen.

> **Note** Clicking the **Restart Heartbeat** button on the Primary appliance will also automatically restart heartbeat on the Secondary appliance.

> **Note** For more details on configuring HA with 2 appliances, please refer to Appliance Clustering for HA.

> **Note** For details on testing and verifying HA, please refer to Clustered Pair Diagnostics.

16. Document Revision History

Version	Date	Change	Reason for Change	Changed By
1.0.0	23 March 2020	Initial version		AH
1.0.1	1 September 2020	New title page Updated Canadian contact details	Branding update Change to Canadian contact details	AH
1.1.0	1 January 2022	Converted the document to AsciiDoc	Move to new documentation system	AH, RJC, ZAC
1.1.1	21 March 2022	Added new multithreading advice	Product change means multithreading is now enabled by default	AH
1.2.0	6 April 2022	Updated GSLB set up instructions to use GUI-driven GSLB configuration Updated DNS server configuration instructions	GSLB updates across all documentation Changed to use new, consistent common component	AH
1.2.1	22 April 2022	Updated SSL related content to reflect latest software version	New software release	RJC
1.2.2	28 September 2022	Updated layer 7 VIP and RIP creation screenshots	Reflect changes in the web user interface	AH

Version	Date	Change	Reason for Change	Changed By
1.2.3	5 January 2023	Combined software version information into one section Added one level of section numbering Added software update instructions Added table of ports used by the appliance Reworded 'Further Documentation' section Removed references to the colour of certain UI elements	Housekeeping across all documentation	AH
1.2.4	2 February 2023	Updated screenshots	Branding update	AH
1.2.5	7 March 2023	Removed conclusion section	Updates across all documentation	AH
1.3.0	24 March 2023	New document theme Modified diagram colours	Branding update	AH
1.3.1	29 June 2023	Updated multithreading advice	New default option in the web user interface	AH

Visit us: www.loadbalancer.org

Phone us: +44 (0)330 380 1064

Phone us: +1 833 274 2566

Email us: info@loadbalancer.org

Follow us: @loadbalancer.org

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.