LOADBALANCER

DEPLOYMENT GUIDE

Load Balancing WEKA S3

Version 1.0.0

Table of Contents

1. About this Guide

This guide details the steps required to configure a load balanced WEKA S3 environment utilizing Loadbalancer.org appliances. It covers the configuration of the load balancers and also any WEKA S3 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 for load balancing WEKA S3. For full specifications of available models please refer to https://www.loadbalancer.org/products/enterprise.

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 or 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. WEKA S3

- All supported versions

4. WEKA S3

The WEKA S3 service is a scalable, resilient service that provides multi-protocol access to data.

The S3 service is implemented by specifying a set of storage hosts that you want to run the S3 protocol on and then creating a logical S3 cluster to expose the S3 service. As you define many hosts that serve the S3 protocol the S3 cluster scales to higher performance.

The WEKA S3 service works on top of the WekaFS file service. Buckets are mapped to (top-level) directories, and objects are mapped to files. Then, the same data can be exposed with either of the WEKA-supported protocols.

5. Load Balancing WEKA S3

> **Note** It's highly recommended that you have a working WEKA S3 environment first before

implementing the load balancer.

5.1. Load Balancing & HA Requirements

The function of the load balancer is to ensure that connections to a WEKA S3 cluster are distributed across healthy WEKA S3 nodes. This is done to provide a highly available and scalable service that allows the WEKA system to service thousands of clients.

To provide HA for the load balancer, Loadbalancer.org recommends that 2 appliances are deployed as an HA clustered pair.

5.2. Load Balancer Configuration

Load balancing a WEKA S3 deployment requires no Virtual Services (VIPs). Instead, both load balancers are configured as smart DNS name servers for the FQDN of the WEKA S3 cluster address in question (e.g. **weka-s3.company.com**). The is achieved using the load balancer's built-in GSLB service and by using DNS delegation.

5.3. Connection Distribution

The GSLB service uses weighted round-robin load balancing to distribute inbound client connections across the healthy WEKA S3 nodes. The weights for the nodes can be set as required depending on relative node performance.

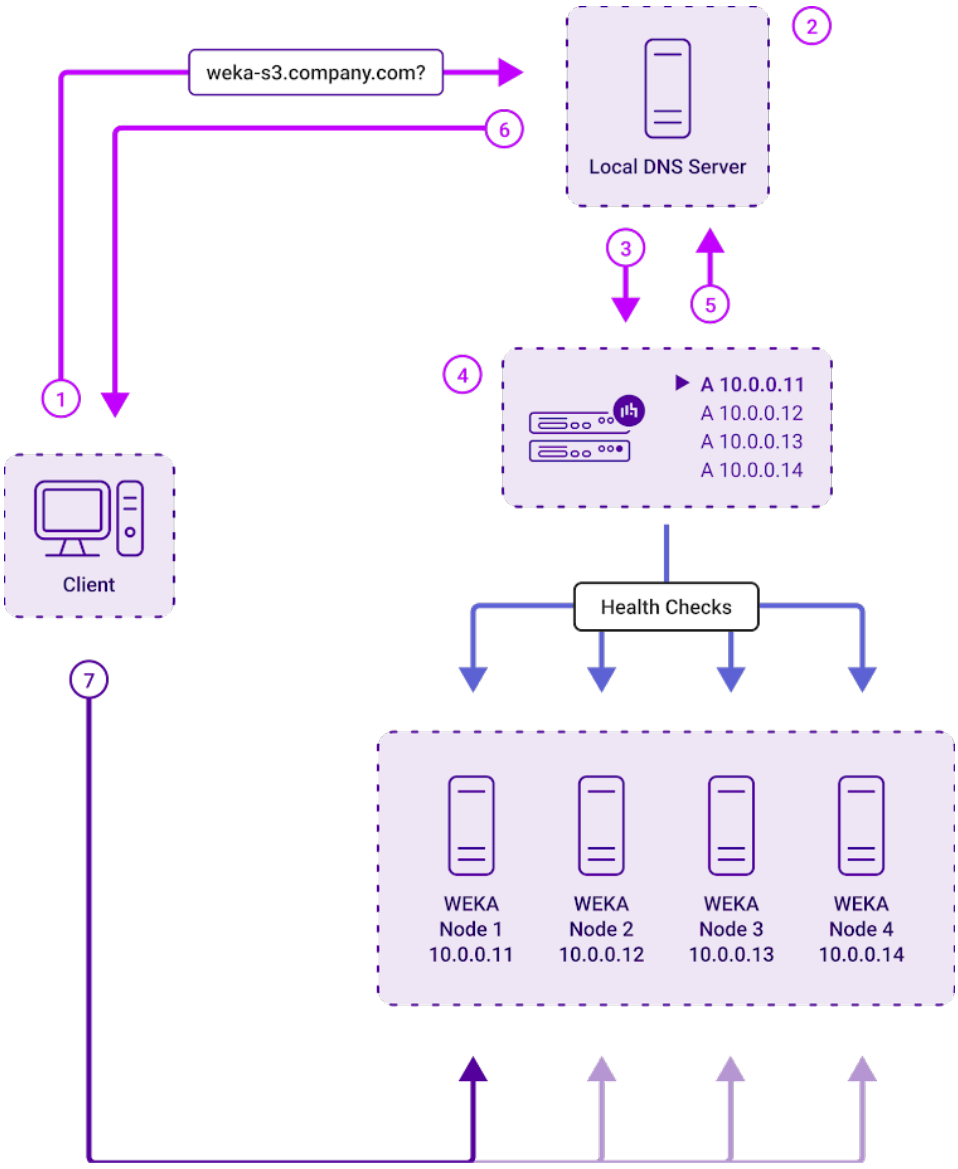
5.4. Health Checks

Each WEKA S3 node is regularly health-checked by each load balancer and this information is used when providing the smart DNS response to inbound DNS queries.

> **Note** For more details on configuring DNS delegation please refer to DNS Server Configuration.

6. Deployment Concept

The diagram below illustrates how the load balancer is deployed.



> **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 an HA clustered pair.

6.1. Explanation

1. The client sends a DNS query for the service FQDN, e.g. **weka-s3.company.com** to the local DNS server.
2. The local DNS server has the sub domain delegated to both load balancers (both load balancers are configured as name servers for the sub domain).
3. One of the load balancers receives the delegated DNS query.
4. The load balancer selects a healthy WEKA S3 node based on the GSLB health checks and the round robin algorithm used.
5. The load balancer returns the IP address of the selected WEKA S3 node to the DNS server.
6. The DNS server returns the IP address of the selected WEKA S3 node to the client.
7. The client connects directly to the WEKA S3 node.

7. Loadbalancer.org Appliance – the Basics

7.1. 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.

7.2. 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.

7.3. 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

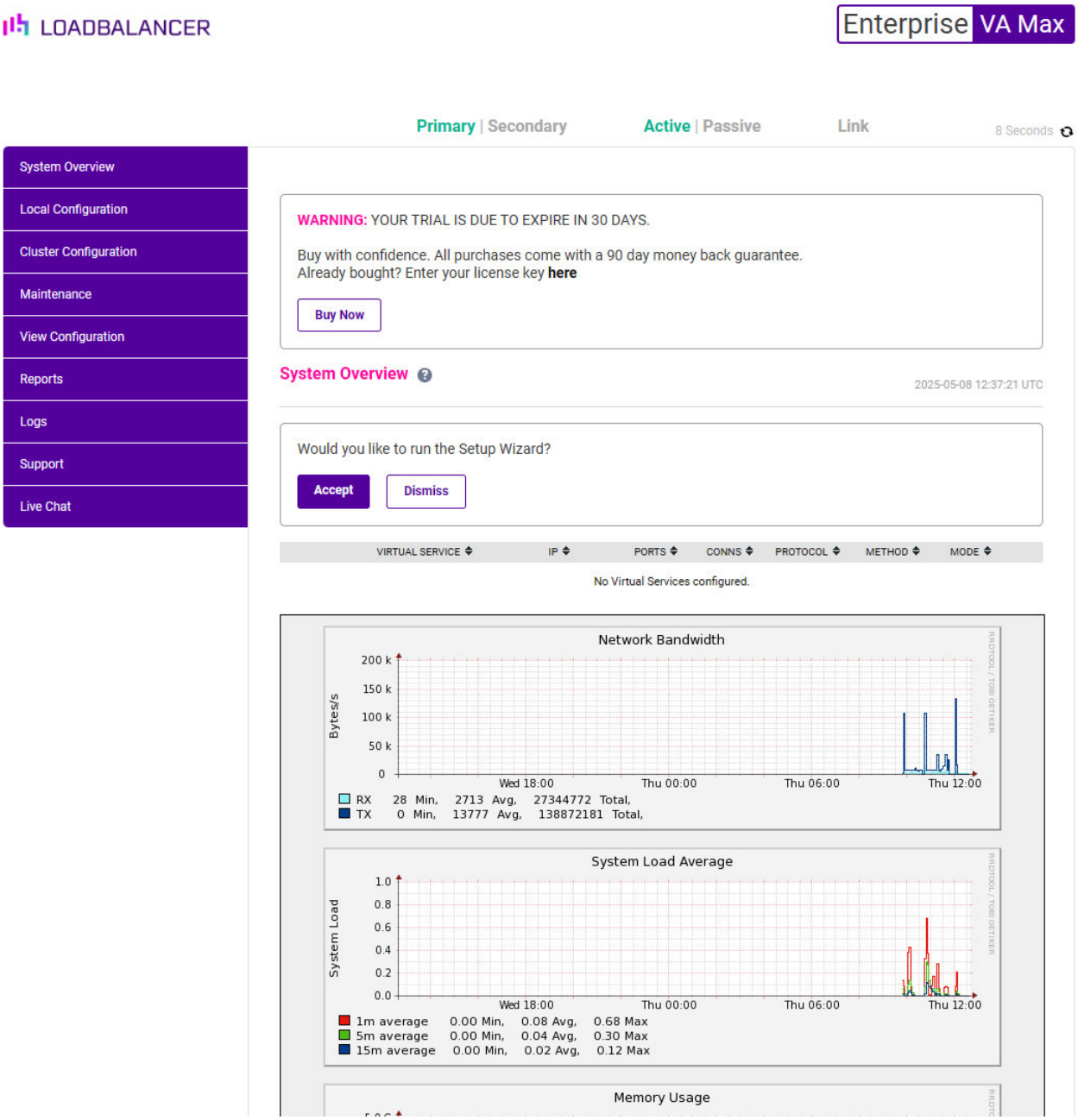
refer to Service Socket Addresses.

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. Click **Dismiss** if you're following a guide or want to configure the appliance manually. Click **Accept** to start the Setup Wizard.

> **Note** The Setup Wizard can only be used to configure Layer 7 services.

7.3.1. 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 creating 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

7.4. Appliance Software Update

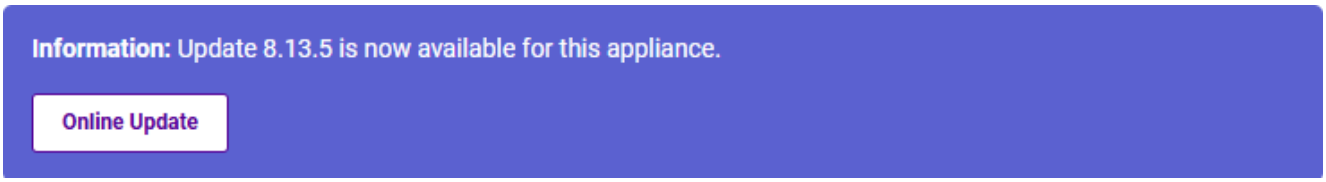
We recommend that the appliance is kept up to date to ensure that you benefit from the latest bug fixes, security updates and feature improvements. Both online and offline update are supported.

> **Note** For full details, please refer to Appliance Software Update in the Administration Manual.

> **Note** Services may need to be restarted/reloaded after the update process completes or in some cases a full appliance restart may be required. We therefore recommend performing the update during a maintenance window.

7.4.1. Online Update

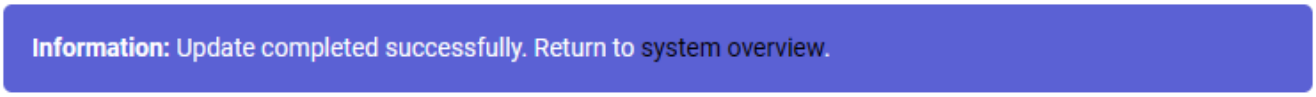
The appliance periodically contacts the Loadbalancer.org update server (**update.loadbalancer.org**) and checks for updates. This is the default behavior and can be disabled if preferred. If an update is found, a notification similar to the example below will be displayed at the top of the WebUI:



Click **Online Update**. A summary of all new features, improvements, bug fixes and security updates included in the update will be displayed. Click **Update** at the bottom of the page to start the update process.

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

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



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

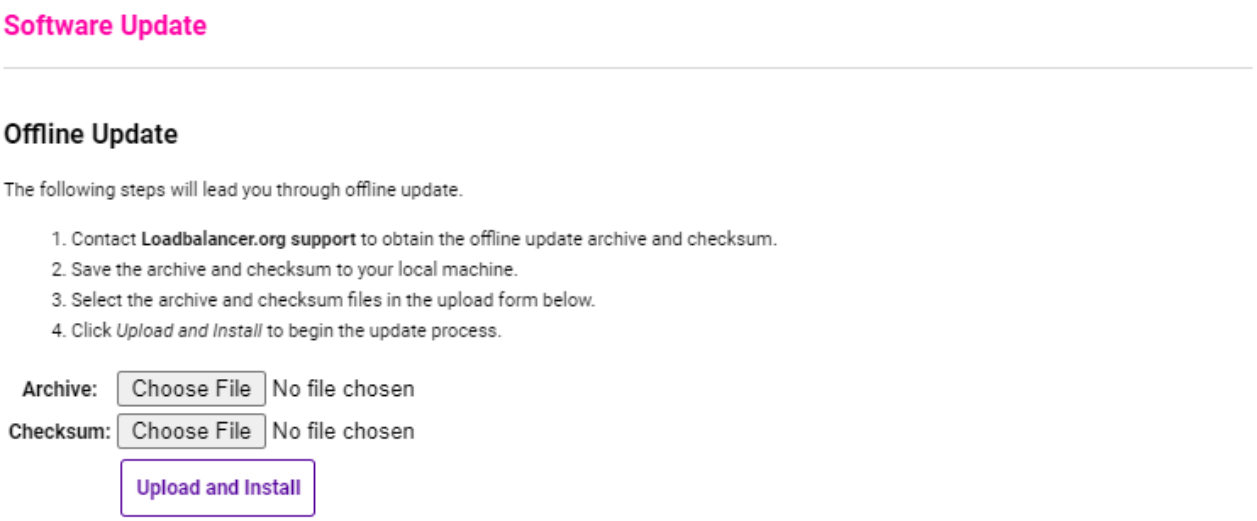
7.4.2. Offline Update

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

To check for the latest version, please refer to our product roadmap page available here. To obtain the latest offline update files contact support@loadbalancer.org.

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.

7.5. Ports Used by the Appliance

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

Protocol	Port	Purpose
TCP	22	SSH
TCP & UDP	53	GSLB
TCP & UDP	123	NTP

Protocol	Port	Purpose
TCP & UDP	161	SNMP
UDP	6694	Heartbeat between Primary & Secondary appliances in HA mode
TCP	7778	HAProxy persistence table replication
TCP	9000	Gateway service for ADC Portal comms
TCP	9080	WebUI - HTTP (disabled by default)
TCP	9081	Nginx fallback server
TCP	9443	WebUI - HTTPS
TCP	25565	Shuttle service for ADC Portal comms

> **Note** All ports listed above except port 123 (NTP) can be changed if required.

- To change the port used for heartbeat, refer to Configuring High Availability
- To change the port used for HAProxy replication, refer to Layer 7 - Advanced Configuration
- To change other ports, refer to Service Socket Addresses

8. Appliance Configuration for WEKA S3

8.1. Step 1 – Configure the HA Pair

If you intend to deploy 2 load balancers in order to configure an HA clustered pair (our recommended configuration) then the HA pair should be configured first before the GSLB configuration takes place. This simplifies the process since GSLB settings will then be automatically replicated to the paired appliance. This helps ensure that both appliances are correctly configured and ready for sub domain delegation (please refer to DNS Server Configuration).

Once the HA pair is configured, the remaining configuration should take place on the Primary unit, the Secondary unit will then be kept in sync automatically. For details on configuring an HA pair, please refer to Configuring HA - Adding a Secondary Appliance.

8.2. Step 2 – Configure the GSLB Health Check

1. Using the WebUI on the Primary appliance, navigate to *Cluster Configuration > Health Check Scripts*.
2. Click **Add New Health Check**.

   [WEKA s3 check]

3. Specify an appropriate *Name* for the new check, e.g. **WEKA-S3-check**.
4. Set the *Type* to **GSLB**.
5. Remove the current contents of the *Primary Node Health Check Contents* text area.

6. Copy/paste the following script into the *Primary Node Health Check Contents* text area.

```
#!/bin/bash

## Configuration: below values are specified via WebUI
host="$1" # IP address of the server to check
hc_port="$2" # Port number

hc_protocol="https" # Health check protocol HTTP/ HTTPS only
hc_path="/wekas3api/health/ready?c=json" # Path for the HTTPS health check (e.g.,
/_s3_gslb_check).

## Configuration: advanced options, only adjust if required
hc_timeout=5 # Timeout for the HTTPS health check (seconds)

get_weight() {
local url="$hc_protocol://$host:$hc_port$hc_path" # Set health check URL
local response
local http_code
local json_data
local score

response=$(curl -k -s -m "$hc_timeout" -w "%{http_code}" "$url")
http_code=${response: -3}
json_data=${response%???}

if [ "$http_code" != "200" ]; then
  exit 1 # Exit with error on failed health check
fi

score=$(echo "$json_data" | jq .score)
scaled_weight=$(echo "10 - $score * 10" | bc)
scaled_weight=$(printf "%.0f" "$scaled_weight")
}

main() {
get_weight
printf "%s\n" "$scaled_weight"
exit 0
}

main
```

7. Click **Update**.

8.3. Step 3 – Configure GSLB

8.3.1. Handling Multiple Subdomains, Including Wildcard Subdomains

Scenario

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

```
weka-s3-<region/location>.domain.tld
```

e.g. `weka-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.weka-s3-region1.domain.tld`.

Solution

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

1. Delegate a single subdomain to the load balancer, e.g. `gslb.`
2. 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
`weka-s3-region1.`	CNAME to `gslb.domain.tld`
`*.weka-s3-region2.`	CNAME to `gslb.domain.tld`
`weka-s3-admin-console.`	CNAME to `gslb.domain.tld`

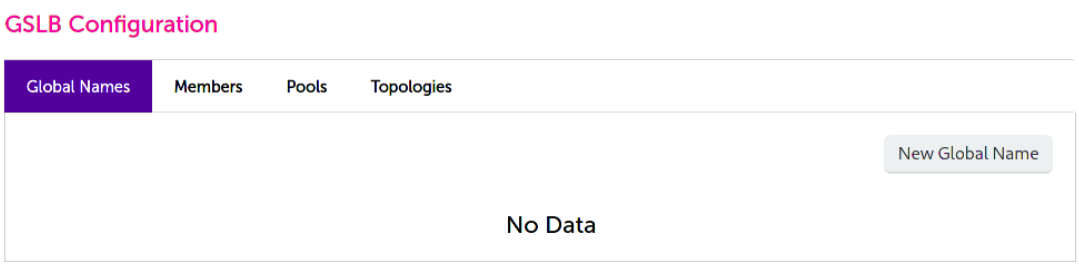
This approach simplifies DNS entry configuration, particularly when wildcard entries are involved.

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

8.3.2. Appliance Configuration

The GSLB service should be configured on the **Primary** load balancer appliance and should be configured at each site if a multi-site deployment is being configured.

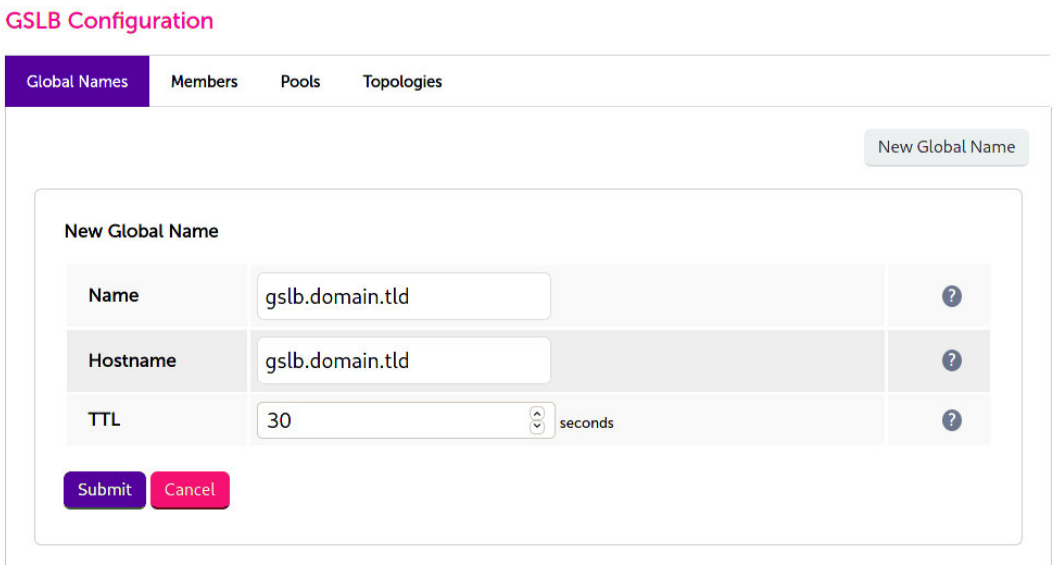
Configuration takes place in the WebUI under *Cluster Configuration > GSLB Configuration*:



Step 1 – Configuring the Global Name

1. Using the WebUI on the Primary appliance, navigate to *Cluster Configuration > GSLB Configuration*.

2. Select the **Global Names** tab.
3. Click the **New Global Name** button.

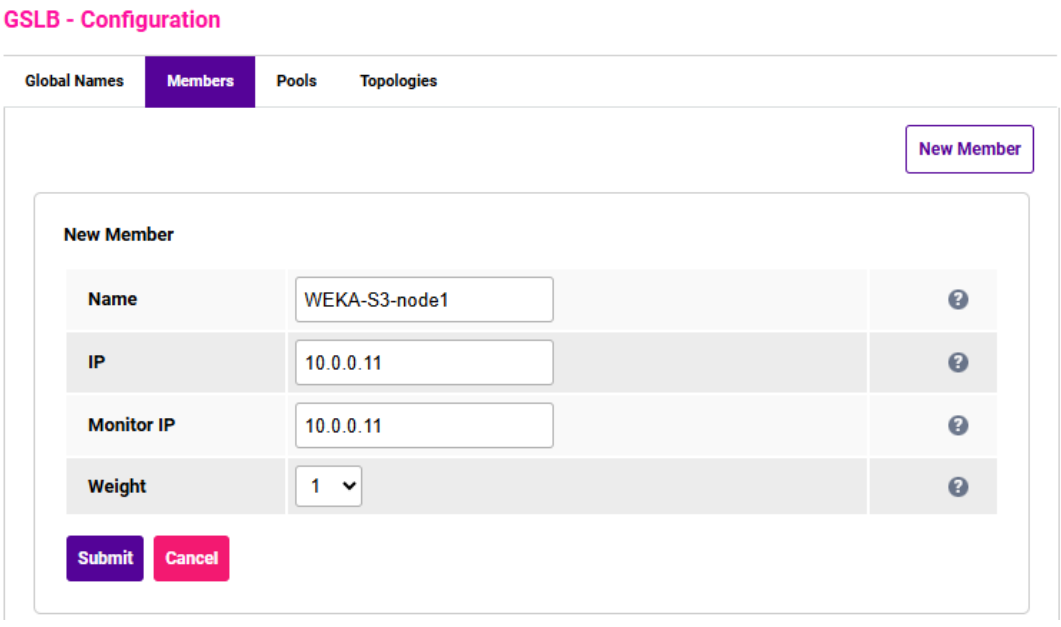


4. Specify an appropriate *Name* for the new hostname, which can just be the subdomain itself, e.g. **gslb.domain.tld**
5. Specify the *Hostname* of what will be the delegated subdomain, e.g. **gslb.domain.tld**
6. Specify a suitable *TTL* in seconds, e.g. **30**.
7. Click **Submit**.

Step 2 – Configure the Members

Each *member* is a single WEKA S3 node.

1. Select the **Members** tab.
2. Click the **New Member** button.



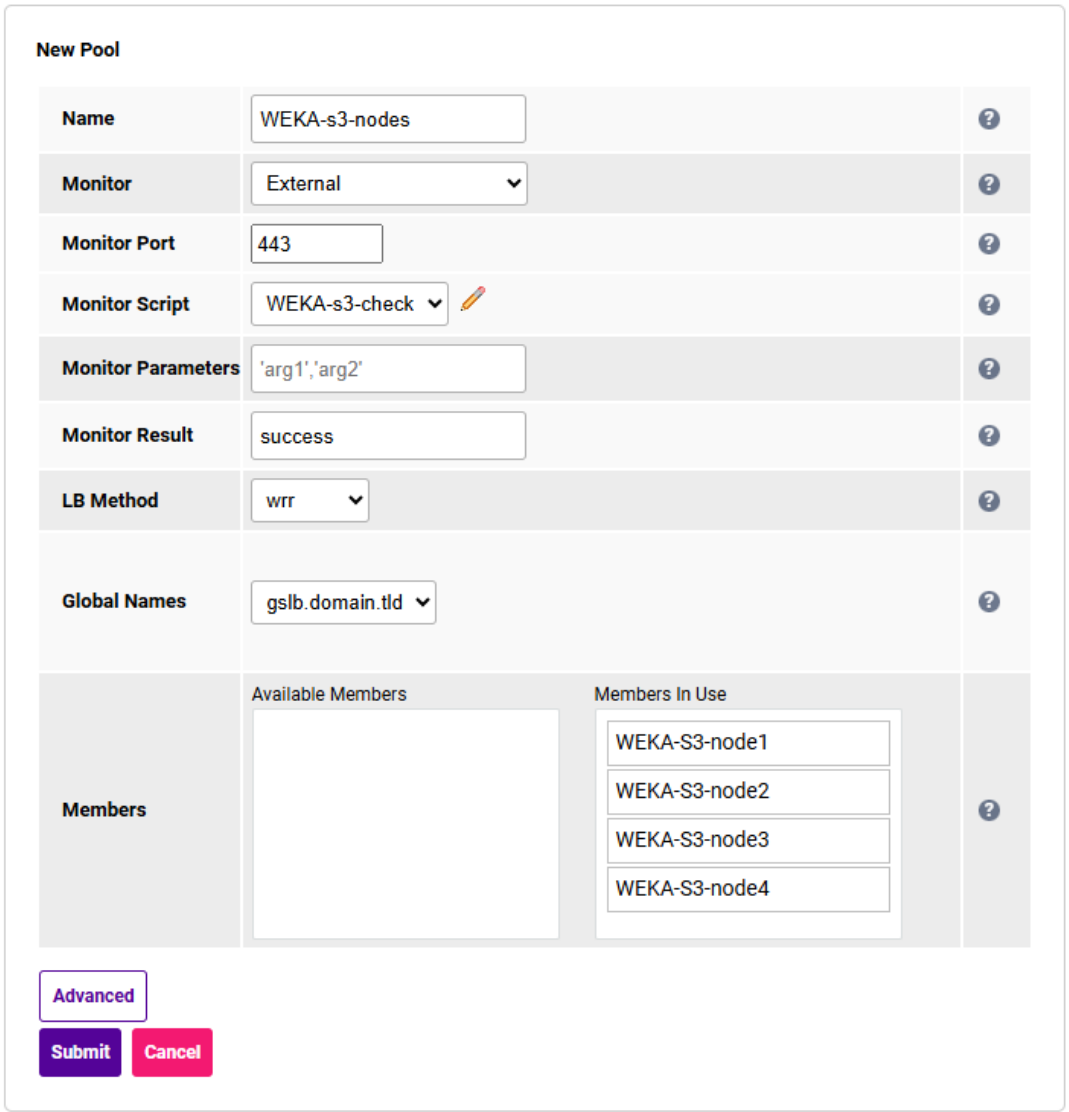
3. Specify an appropriate *Name* for the member, e.g. **WEKA-S3-node1**.
4. Specify an *IP* address for the member: in this context, this should be the IP address of the WEKA S3 node in question, e.g. **10.0.0.11**.
5. Specify the *Monitor IP*, normally this is the same as the *IP* address.
6. Click **Submit**.
7. Repeat these steps to add all additional WEKA S3 nodes.

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, a pool would be created linking the global name `gslb.domain.tld` with the members (WEKA S3 nodes), all of which should serve WEKA S3 traffic.

1. Select the **Pools** tab.
2. Click the **New Pool** button.



3. Specify an appropriate *Name* for the pool, e.g. **WEKA-S3-nodes**.
4. Set the *Monitor* to **External**.
5. Set *Monitor Port* to **443**.
6. Select the *Monitor Script* created previously, e.g. **WEKA-S3-check**.
7. Set *LB Method* to **wrr**.
8. From the *Global Names* list box, select the global name created previously, e.g. **gslb.domain.tld**.
9. In the *Members* section, drag the appropriate members (WEKA S3 nodes) from the *Available Members* box into the *Members In Use* box.
10. Click **Submit**.

Step 4 – 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**.

8.3.3. Testing the GSLB Configuration

The configuration can be tested to make sure it's working as expected.

From the command line on a Microsoft Windows machine, the *nslookup* program can be used to send test DNS queries to the load balancer(s). The Primary load balancer is located at IP address 10.0.0.1 in the example presented here.

For the test, use the *-norecurse* option to instruct the load balancer **not** to attempt to query another server for the answer. A successful test would see the load balancer respond with the IP address of one of the online WEKA S3 nodes, like so:

```
C:\Users\me>nslookup -norecurse gslb.domain.tld 10.0.0.1
Server: UnKnown
Address: 10.0.0.1

Name: gslb.domain.tld
Address: 10.0.0.11
```

8.3.4. 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 balancer's GSLB services will serve the appropriate IP addresses to the DNS servers. Using the example presented in this guide, 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. Steps to configure DNS

delegation on a Microsoft DNS server can be found in the appendix, in the section Microsoft DNS Server Delegation.

9. Testing & Verification

Since the load balancers have no Virtual Services, there is no graphical overview of 'healthy' services to check and verify on the System Overview page of the WebUI. Instead, the GSLB configuration should be checked to ensure that the client is able to successfully resolve the service FQDN and connect to a healthy WEKA S3 node.

9.1. Testing GSLB

Please refer to the previous section Testing the GSLB Configuration.

9.2. Accessing the Service

Verify that a test connection is passed to to one of the online WEKA S3 nodes.

10. 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.

11. Further Documentation

For additional information, please refer to the Administration Manual.

12. Appendix

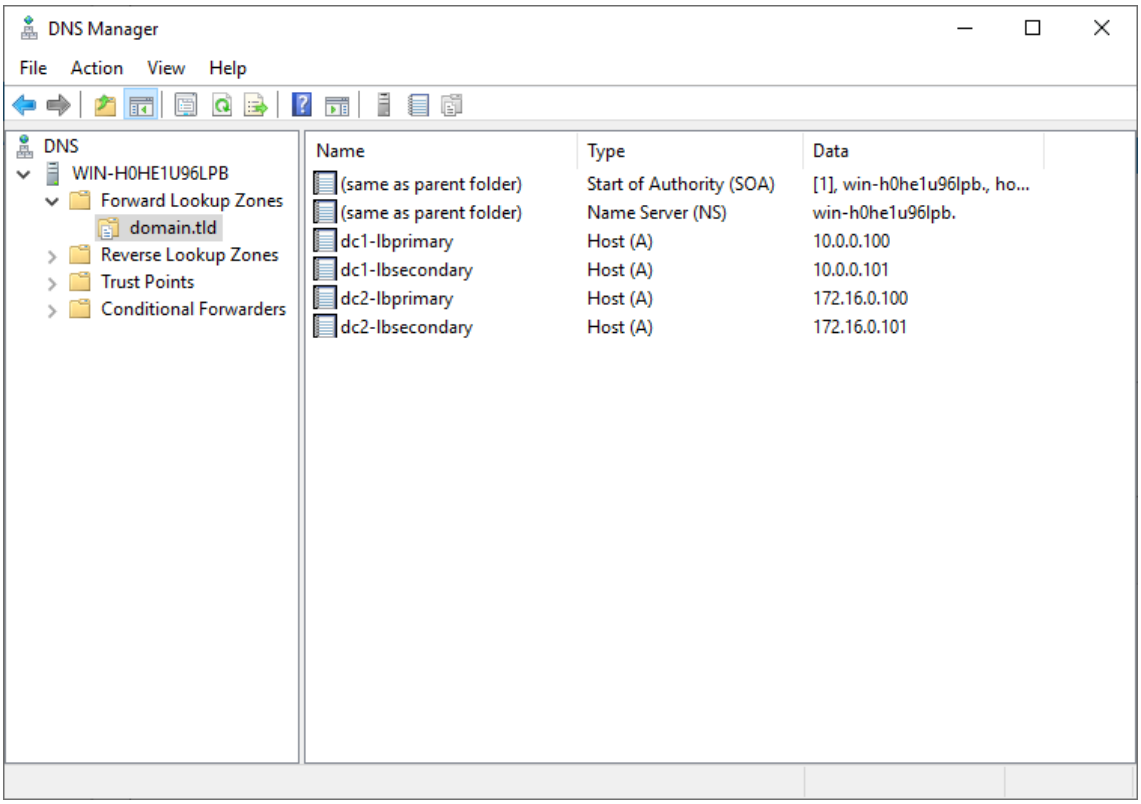
13. Microsoft DNS Server Delegation

The Steps to create a DNS delegation on a Microsoft DNS server in the context of setting up GSLB on our appliance are presented below.

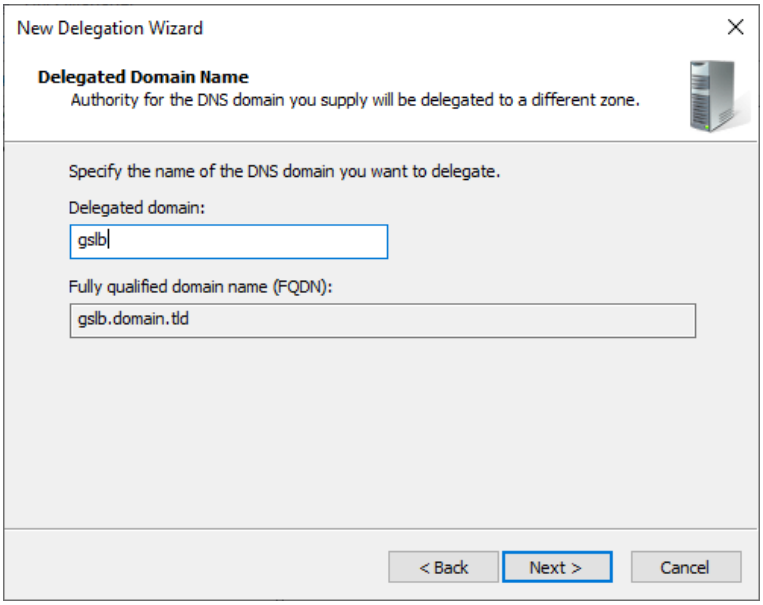
1. Using the *Action > New Host* menu option in **DNS Manager**, create A records for each load balancer at every site. For example:

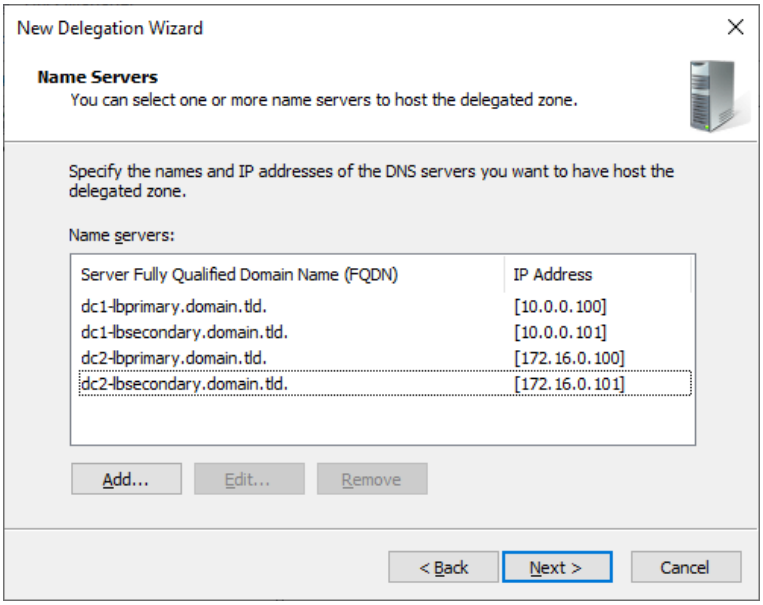
   - `dc1-lbprimary.domain.tld`
   - `dc1-lbsecondary.domain.tld`
   - `dc2-lbprimary.domain.tld`
   - `dc2-lbsecondary.domain.tld`

   Once created, the entries in **DNS Manager** would appear as follows:



2. Ensure that the load balancer has been correctly configured for GSLB and is working, then use the **New Delegation** wizard to delegate the subdomain to the load balancers. The delegation must include the FQDNs for all the load balancer A records that were created in the previous step. The delegation wizard can be accessed using *Action > New Delegation*.





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

   From the Windows command line, the `nslookup` command 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 a different server(s), giving the other server's details as shown in the following example:

```
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 nodes (in "direct to node" scenarios) or one of the Virtual Services (in GSLB to Virtual Service scenarios) as shown in the following example:

```
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
```

13.1. 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.

13.1.1. 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

WebUI Main Menu Option	Sub Menu Option	Description
Local Configuration	Network Interface Configuration	Interface IP addresses, bonding configuration and VLANs
Local Configuration	Routing	Default gateways and static routes
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	Firewall Script	Firewall (iptables) configuration
Maintenance	Firewall Lockdown Wizard	Appliance management lockdown settings

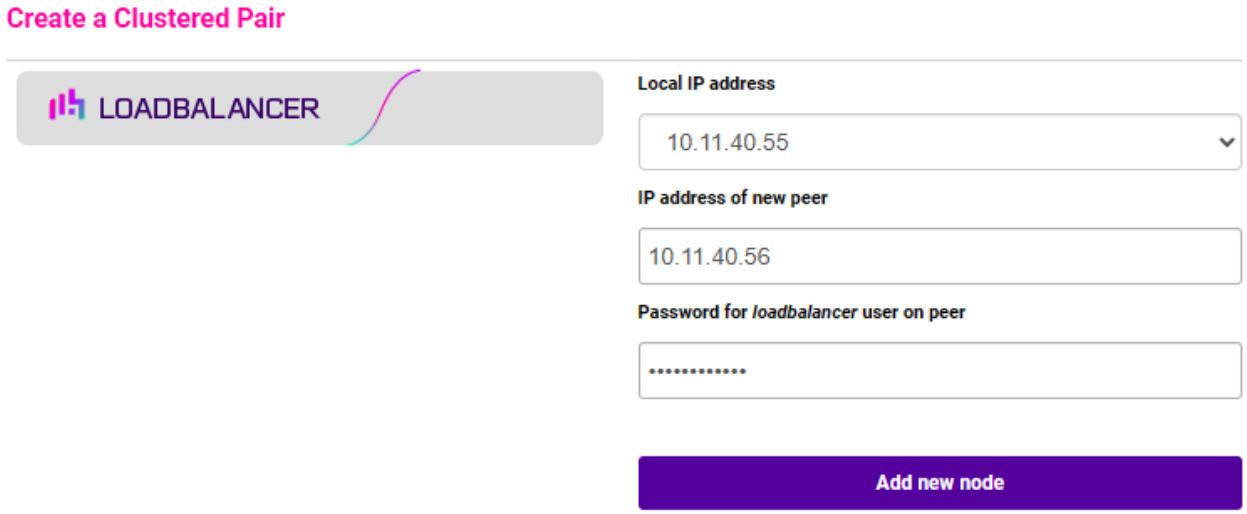
> **Important** Make sure that where any of the above have been configured on the Primary appliance, they're also configured on the Secondary.

13.1.2. Configuring the HA Clustered Pair

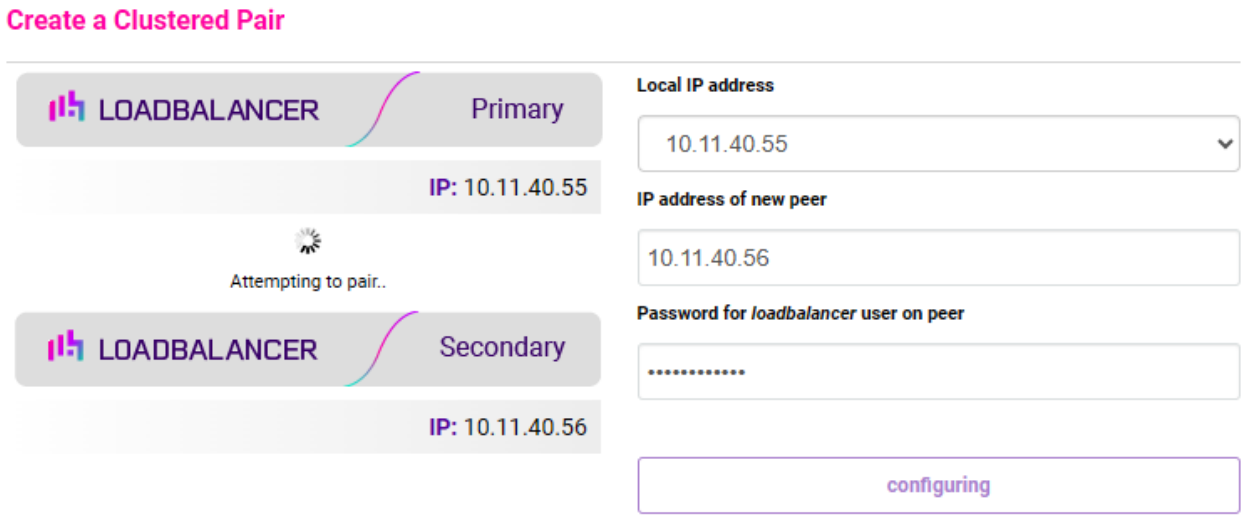
> **Important** During HA pairing, all WebUI users and passwords are synchronized from the Primary to the Secondary. After clustering completes (you will be logged out of the Secondary when this occurs), the Primary's credentials should be used to login to both nodes.

> **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.

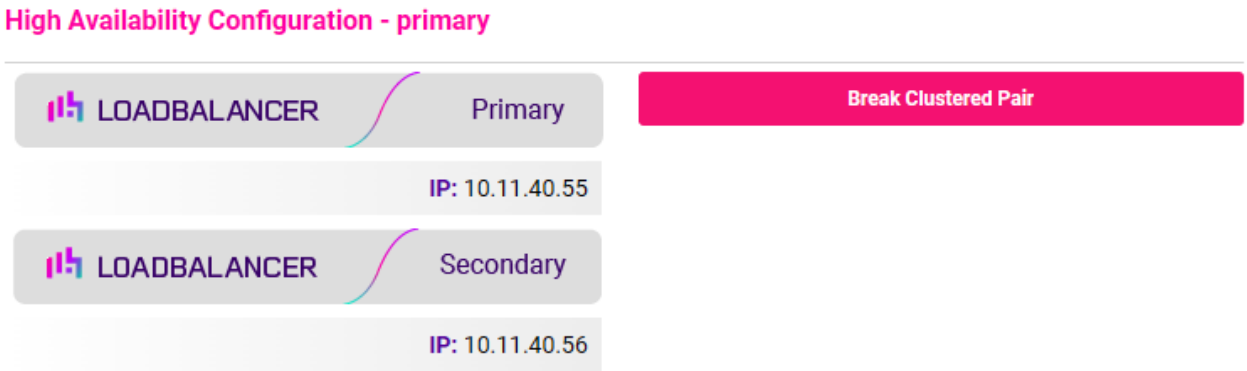
1. Deploy a second appliance that will be the Secondary and configure initial network settings.
2. Using the WebUI on the Primary appliance, navigate to: ***Cluster Configuration > High-Availability Configuration***.



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 Configuring High Availability.

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

14. Document Revision History

Version	Date	Change	Reason for Change	Changed By
1.0.0	2 June 2026	Initial version		RJC

LOADBALANCER

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.