# Table of Contents

1. About this Guide .............................................. 3
2. Loadbalancer.org Appliances Supported .......................... 3
3. Loadbalancer.org Software Versions Supported ................. 3
4. Scality RING Software Versions Supported .................. 3
5. Scality RING .................................................. 3
6. Load Balancing Scality RING .................................. 3
   Load Balancing & HA Requirements .......................... 3
   Persistence (aka Server Affinity) ......................... 3
   Virtual Service (VIP) Requirements .......................... 4
   Port Requirements ............................................. 4
   SSL Termination ............................................... 4
   Health Checks .................................................. 4
   GSLB / Location Affinity .................................... 4
   Alternative Load Balancing Method for Read-Intensive Deployments (Direct Routing) .......... 4
7. Performance and Sizing for a Virtual Load Balancer Deployment with Scality RING ......................... 4
8. Deployment Concept .......................................... 5
9. Loadbalancer.org Appliance – the Basics ....................... 5
   Virtual Appliance ............................................. 5
   Initial Network Configuration ................................ 5
   Accessing the WebUI .......................................... 6
   Main Menu Options ........................................... 7
   HA Clustered Pair Configuration ............................. 8
10. Appliance Configuration for Scality RING ...................... 8
    Enabling Multithreaded Load Balancing .................... 8
11. Appliance Configuration for Scality RING – Using Layer 7 SNAT ............................................. 8
    Configuring VIP 1 – S3 ...................................... 8
    Configuring the Virtual Service (VIP) ...................... 8
    Defining the Real Servers (RIPs) .......................... 9
12. Additional Configuration Options & Settings .................. 10
    SSL Termination ............................................. 10
    SSL Termination on the load balancer - SSL Offloading .... 11
    Certificates .................................................. 11
    Configuring SSL Termination on the Load Balancer .......... 13
    Finalizing the Configuration ................................ 14
13. Testing & Verification ........................................ 14
    Using System Overview .................................... 14
14. Technical Support .............................................. 14
15. Further Documentation ........................................ 15
16. Conclusion ..................................................... 15
17. Appendix ....................................................... 16
    Configuring GSLB / Location Affinity ....................... 16
    Conceptual Overview ....................................... 16
    Appliance Configuration .................................... 17
    DNS Server Configuration .................................... 19
    Alternative Load Balancing Method for Read-Intensive Deployments (Direct Routing) ...... 19
    Caveats ....................................................... 20
    Appliance Configuration for Scality RING – Using Layer 4 DR Mode (Direct Routing) ...... 20
    Configuring HA - Adding a Slave Appliance ............... 21
18. Document Revision History .................................... 24
1. About this Guide

This guide details the steps required to configure a load balanced Scality RING environment utilizing Loadbalancer.org appliances. It covers the configuration of the load balancers and also any Scality RING 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 Scality RING. For full specifications of available models please refer to https://www.loadbalancer.org/products. Some features may not be supported in all cloud platforms due to platform specific limitations, please check with Loadbalancer.org support for further details.

3. Loadbalancer.org Software Versions Supported

- V8.4.1 and later

4. Scality RING Software Versions Supported

- Scality RING – 7.4.4 and later

5. Scality RING

Scality is a global company that develops software-defined object storage via commercial products such as RING. Scality RING software deploys on industry-standard x86 servers to store objects and files whilst providing compatibility with the Amazon S3 API.

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

A variety of load balancing methods are currently supported by Scality RING, dependent on customer infrastructure, including layer 4, layer 7, and geo GSLB / location affinity. The RING service that should be load balanced is the S3 component.

6. Load Balancing Scality RING

**Note**

It’s highly recommended that you have a working Scality RING environment first before implementing the load balancer.

**Load Balancing & HA Requirements**

The function of the load balancer is to distribute inbound connections across a cluster of Scality RING nodes, to provide a highly available and scalable service. One virtual service is used to load balance the S3 aspect of RING.

**Persistence (aka Server Affinity)**

Client persistence is not required and should not be enabled.
Virtual Service (VIP) Requirements
To provide load balancing for Scality the following VIP is required:

- S3: handles requests from S3 client applications via HTTP and HTTPS

Port Requirements
The following table shows the ports that are load balanced:

<table>
<thead>
<tr>
<th>Port</th>
<th>Protocols</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>TCP/HTTP</td>
<td>Requests from S3 client applications</td>
</tr>
<tr>
<td>443</td>
<td>TCP/HTTPS</td>
<td>Requests from S3 client applications</td>
</tr>
</tbody>
</table>

SSL Termination
SSL termination on the load balancer is recommended for load balancing Scality RING.

Health Checks
The S3 service uses the "Negotiate HTTP (GET)" health check.

GSLB / Location Affinity
For multi-site RING 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 RING service and/or load balancers are off-line then local clients are automatically directed to a functioning RING cluster at another site.

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

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

A more detailed explanation of this alternative load balancing method can be found in Alternative Load Balancing Method for Read-Intensive Deployments (Direct Routing).

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

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

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

VIPS = Virtual IP Addresses

NOTE: The load balancer can be deployed as a single unit, although Loadbalancer.org recommends a clustered pair for resilience & high availability. Please refer to Configuring HA - Adding a Slave Appliance for more details on configuring a clustered pair.

9. Loadbalancer.org Appliance – the Basics

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 1 CPU, 2GB 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 The Virtual Appliance - Hypervisor Deployment and the ReadMe.txt text file included in the VA download for more detailed information on deploying the VA using various Hypervisors.

Note
For the VA, 4 NICs are included but only eth0 is connected by default at power up. If the other NICs are required, these should be connected using the network configuration screen within the Hypervisor.

Initial Network Configuration

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

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

**Accessing the WebUI**

The WebUI is accessed using a web browser. Appliance authentication is based on Apache .htaccess files. User admin tasks such as adding users and changing passwords can be performed using the WebUI menu option: *Maintenance > Passwords*.

**Note**  
A number of compatibility issues have been found with various versions of Internet Explorer. The WebUI has been tested and verified using both Chrome & Firefox.

**Note**  
If required, users can also be authenticated against LDAP, LDAPS, Active Directory or Radius. For more information please refer to [External Authentication](#).

1. Using a browser, access the WebUI using the following URL:


2. Log in to the WebUI:

   **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:
Note: The WebUI for the VA is shown, the hardware and cloud appliances are very similar. The yellow licensing related message is platform & model dependent.

3. You'll be asked if you want to run the Setup Wizard. If you click **Accept** the Layer 7 Virtual Service configuration wizard will start. If you want to configure the appliance manually, simple click **Dismiss**.

**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
HA Clustered Pair Configuration
Loadbalancer.org recommend that load balancer appliances are deployed in pairs for high availability. In this guide a single unit is deployed first, adding a secondary slave unit is covered in Configuring HA - Adding a Slave Appliance.

10. Appliance Configuration for Scality RING

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

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

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

To enable multithreaded mode from the WebUI:

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

11. Appliance Configuration for Scality RING – Using Layer 7 SNAT

Configuring VIP 1 – S3
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. S3
3. Set the Virtual Service IP Address field to the required IP address, e.g. 172.16.254.120
4. Set the Ports field to 80
5. Set the Layer 7 Protocol to HTTP Mode

6. Click Update to create the virtual service

7. Click Modify next to the newly created VIP

8. Set Persistence Mode to None

9. Set Health Checks to Negotiate HTTP (GET)

10. Set Request to send to /_/healthcheck/deep/

11. Scroll to the Other section and click Advanced

12. Enable Force to HTTPS by clicking the Yes radio button**

13. Click Update

** Defining the Real Servers (RIPs)

1. Using the web user interface, navigate to Cluster Configuration > Layer 7 – Real Servers and click on Add a new Real Server next to the newly created VIP

2. Define the Label for the real server as required, e.g. ring-node1

3. Set the Real Server IP Address field to the required IP address, e.g. 172.16.254.101

4. Click Update

5. Repeat these steps to add additional RING nodes as real servers as required
12. Additional Configuration Options & Settings

SSL Termination

SSL termination can be handled in the following ways:

1. On the Real Servers - aka SSL Pass-through

2. On the load balancer – aka SSL Offloading *(recommend for Scality RING)*

3. On the load balancer with re-encryption to the backend servers – aka SSL Bridging

In the case of Scality RING, it is recommended that SSL be terminated on the load balancer *(SSL offloading)* with *Force to HTTPS* enabled.

NOTE:

- SSL termination on the load balancer can be very CPU intensive.
- By default, a self-signed certificate is used for the new SSL VIP. Certificates can be requested on the load balancer or uploaded as described in the section below. The default self-signed certificate can be regenerated if needed using the WebUI menu option: SSL Certificate and clicking the *Regenerate Local SSL Certificate button*.
- The backend for the SSL VIP can be either a Layer 7 SNAT mode VIP or a Layer 4 NAT or SNAT mode VIP. Layer 4 DR mode cannot be used since stunnel acts as a proxy, and the RING servers see requests with a source IP address of the VIP. However, since the RING servers believe that they own the VIP (due to the loopback adapter configured to handle to ARP problem) they are unable to reply to stunnel.
- Finally, ensure that the Scality RING S3 Console and S3 Browser URL are configured as HTTPS via the S3 Service as per the example image below.
SSL Termination on the load balancer - SSL Offloading

In this case, an SSL VIP utilizing STunnel is configured on the appliance and an SSL certificate is uploaded and associated to the Virtual Service. Data is encrypted from the client to the load balancer, but is unencrypted from the load balancer to the backend servers as shown above.

Certificates
If you already have an SSL certificate in either PFX or PEM file format, this can be uploaded to the Load balancer using the certificate upload option as explained in Uploading Certificates. Alternatively, you can create a Certificate Signing Request (CSR) and send this to your CA to create a new certificate.

Generating a CSR on the Load Balancer
CSR’s can be generated on the load balancer to apply for a certificate from your chosen CA.
To generate a CSR:

1. Using the WebUI, navigate to: Cluster Configuration > SSL Certificates

2. Click **Add a new SSL Certificate** & select **Create a New SSL Certificate (CSR)**

3. Enter a suitable label (name) for the certificate, e.g. **Cert1**

4. Populate the remaining fields according to your requirements

5. Once all fields are complete click **Create CSR**

6. To view the CSR click **Modify** next to the new certificate, then expand the Certificate Signing Request (CSR) section

7. Copy the CSR and send this to your chosen CA

8. Once received, copy/paste your signed certificate into the Your Certificate section

9. Intermediate and root certificates can be copied/pasted into the Intermediate Certificate and Root Certificate sections as required

10. Click **Update** to complete the process

**Uploading Certificates**

If you already have a certificate in either PEM or PFX format, this can be uploaded to the load balancer.

**To upload a Certificate:**
1. Using the WebUI, navigate to: Cluster Configuration > SSL Certificates

2. Click **Add a new SSL Certificate** & select Upload prepared PEM/PFX file

![Image of SSL Certificate upload interface]

3. Enter a suitable Label (name) for the certificate, e.g. **Cert1**

4. Browse to and select the certificate file to upload (PEM or PFX format)

5. Enter the password, if applicable

6. Click **Upload Certificate**, if successful, a message similar to the following will be displayed:

   ![Information message: cert1 SSL Certificate uploaded successfully]

   **Note** It’s important to backup all your certificates. This can be done via the WebUI from **Maintenance > Backup & Restore > Download SSL Certificates**.

### Configuring SSL Termination on the Load Balancer

**To configure SSL VIP:**

1. Using the WebUI, navigate to: Cluster Configuration > SSL Termination and click **Add a new Virtual Service**

   ![SSL Termination - Add a new Virtual Service interface]

   2. Set Associated Virtual Service to the appropriate VIP, e.g. **S3**. This will automatically fill in the label as the VIP name with SSL inserted in front of the VIP name e.g. **SSL-S3**.
The Associated Virtual Service drop-down is populated with all single port, standard (i.e. non-manual) Layer 7 VIPs available on the load balancer. Using a Layer 7 VIP for the backend is the recommended method although as mentioned earlier, Layer 4 NAT mode and layer 4 SNAT mode VIPs can also be used if required. To forward traffic from the SSL VIP to these type of VIPs, you’ll need to set Associated Virtual Service to Custom, then configure the IP address & port of the required VIP.

3. Leave Virtual Service Port set to 443
4. Leave SSL operation Mode set to High Security
5. Select the required certificate from the SSL Certificate drop-down.
6. Click Update
7. Reload STunnel to apply the new settings using the link provided in the blue box

Once configured, HTTP traffic will be load balanced by the Layer 7 SNAT mode VIP and HTTPS traffic will be terminated by the SSL VIP, then passed on to the Layer 7 SNAT mode VIP as unencrypted HTTP for load balancing.

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

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

13. Testing & Verification
Using System Overview
The System Overview can be viewed in the WebUI. It shows a graphical view of all VIPs & RIPs (i.e. the RING Nodes) and shows the state/health of each server as well as the state of the each cluster as a whole. The example below shows that all RING nodes are healthy and available to accept connections.

14. 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
15. Further Documentation


16. Conclusion

Loadbalancer.org appliances provide a very cost effective solution for highly available load balanced Scality RING environments.
17. Appendix

Configuring GSLB / Location Affinity

Conceptual Overview

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

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

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

Explanation:

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

Note

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

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

Handling Multiple Sub domains

Scality RING DNS configurations typically use the following DNS sub domains (or something similar):

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

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

Appliance Configuration

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

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

Polaris Config

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

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

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

The elements presented in **boldface** should be customised to reflect the RING deployment in question. Stanzas should be added or removed under the 'members' section as needed to reflect all sites that are to be used in the deployment.
Be sure to click the **Update** button when finished to write the configuration to disk.

---

**Note**

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

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

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

---

**Topology Config**

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

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

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

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

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

```yaml
datacenter1:
  - 10.0.0.0/24

datacenter2:
  - 172.16.0.0/24

datacenter3:
  - 192.168.0.0/18
```
Applying the Configuration

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

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

DNS Server Configuration

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

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

The exact steps for creating a DNS delegation vary between different DNS servers and are outside the scope of this document. For further information, a blog post that walks through creating a DNS delegation on a Microsoft DNS server in the context of setting up GSLB on our appliance can be found at https://www.loadbalancer.org/blog/loadbalancer-org-releases-a-gslb/ (see the section titled “Delegating your subdomain to your GSLB’s using Microsoft’s DNS Server”.)

Alternative Load Balancing Method for Read-Intensive Deployments (Direct Routing)

Direct routing, also known as direct server return or DSR, is a method of load balancing. With direct routing, reply traffic flows directly from the back end servers to the clients. In this way, the load balancer is completely bypassed on the return journey for a given connection, thus removing the load balancer as a potential bottleneck for traffic on the return path.

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

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

- The load balancers must be on the same network segment / switching fabric as the RING nodes (due to the fact that this load balancing method works by rewriting MAC addresses, i.e. operates at layer 2 of the OSI model)

- Each RING node must own the VIP address so that they can all accept and reply to the load balanced traffic. This address should be assigned to a loopback network adaptor

- Each RING node must be configured to not reply to ARP requests for the VIP address or advertise that they own the address

For guidance on configuring the RING nodes for direct routing, in the context of the caveats described above, please consult with Scality Sales Engineering or Support.

Appliance Configuration for Scality RING – Using Layer 4 DR Mode (Direct Routing)

Configuring VIP 1 – S3

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. S3

3. Set the Virtual Service IP Address field to the required IP address, e.g. 192.168.87.67
4. Set the *Ports* field to **80**
5. Leave the *Protocol* set to **TCP**
6. Leave the *Forwarding Method* set to **Direct Routing**
7. Click **Update** to create the virtual service
8. Click **Modify** next to the newly created VIP
9. Ensure that the *Persistence Enable* checkbox is unchecked
10. Set the *Health Checks Check Type* to **Negotiate**
11. Set the *Check Port* to **80**
12. Set the *Protocol* to **HTTP**
13. Set the Request to send to */_/healthcheck/deep/*
14. 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. **ring-node1**
3. Set the *Real Server IP Address* field to the required IP address, e.g. **192.168.87.88**
4. Click **Update**
5. Repeat these steps to add additional RING nodes as real servers as required

**Configuring HA - Adding a Slave 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 should be configured first, then the Slave should be added. Once the Primary and Slave are paired, all load balanced services configured on the Primary are automatically replicated to the Slave over the network using SSH/SCP.

**Note**

For Enterprise Azure, the HA pair should be configured first. In Azure, when creating a VIP using an HA pair, 2 private IPs must be specified – one for the VIP when it’s active on the Primary and one for the VIP when it’s active on the Slave. Configuring the HA pair first, enables both IPs to be specified when the VIP is created.

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 Slave) will take over.

**Note**

A number of settings are not replicated as part of the Primary/Slave pairing process and therefore must be manually configured on the Slave appliance. These are listed by WebUI menu option in the table below:
To add a Slave node - i.e. create a highly available clustered pair:

1. Deploy a second appliance that will be the Slave and configure initial network settings.
2. Using the WebUI, navigate to: Cluster Configuration > High-Availability Configuration.
3. Specify the IP address and the loadbalancer user’s password for the Slave (peer) appliance as shown above
4. Click Add new node.
5. The pairing process now commences as shown below:
6. Once complete, the following will be displayed:

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

---

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

**Note**
For more details on configuring HA with 2 appliances, please refer to **Appliance Clustering for HA**.
## 18. Document Revision History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Change</th>
<th>Reason for Change</th>
<th>Changed By</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0.0</td>
<td>14 February 2020</td>
<td>Initial version</td>
<td></td>
<td>IBG</td>
</tr>
<tr>
<td>1.0.1</td>
<td>3 September 2020</td>
<td>New title page</td>
<td>Branding update</td>
<td>AH</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Updated Canadian contact details</td>
<td>Change to Canadian contact details</td>
<td></td>
</tr>
<tr>
<td>1.1.0</td>
<td>1 October 2021</td>
<td>Converted the document to AsciiDoc</td>
<td>Move to new documentation system</td>
<td>AH,RJC,ZAC</td>
</tr>
</tbody>
</table>
About Loadbalancer.org

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