# Table of Contents

1. Introduction .................................................................................................................. 3
2. About Enterprise Azure ........................................................................................................ 3
   - Main Differences to Our Standard (Non-Cloud) Product .............................................. 3
   - Why use Enterprise Azure? .......................................................................................... 4
3. Azure Deployment Models ................................................................................................. 4
4. Accessing Microsoft Azure ............................................................................................... 4
5. Azure Management ............................................................................................................ 4
   - Accessing the Azure Portal ......................................................................................... 5
   - Azure PowerShell & Azure CLI ..................................................................................... 5
6. Deploying Enterprise Azure From the Marketplace .......................................................... 5
7. Accessing the Appliance .................................................................................................... 11
   - Accessing the Appliance using the WebUI ................................................................. 11
     - WebUI Menu Options ............................................................................................... 12
     - Appliance Security .................................................................................................. 13
   - Checking For Updates .................................................................................................. 13
   - Appliance Licensing .................................................................................................... 13
   - Enterprise Azure Non-standard WebUI Menu Options .............................................. 14
   - Accessing the Appliance using SSH ........................................................................... 14
     - Generating SSH Keys ........................................................................................... 15
     - Accessing the Appliance from Linux ....................................................................... 16
     - Accessing the Appliance from Windows using PuTTY ............................................ 16
8. Configuration Examples .................................................................................................... 18
   - High Availability ......................................................................................................... 18
   - Key Concepts .............................................................................................................. 18
   - Implementing HA in Azure .......................................................................................... 19
     - 1 – Load Balancing Web Servers, HA Configuration, 1 subnet, layer 7 ................. 20
     - 2 – Load Balancing Web Servers, HA Configuration, 1 subnet, layer 7 with SSL Termination ... 32
     - 3 – Load Balancing Web Servers, HA Configuration, 1 subnet, layer 7 (Manual Configuration) with SSL Termination ...... 45
     - 4 – Load Balancing Web Servers, Single Appliance, 2 subnets, layer 4 NAT Mode ...... 47
     - 5 – Load Balancing Web Servers, Single Appliance, 2 subnets, layer 7 with TProxy .... 52
9. Testing – General Comments ............................................................................................ 54
   - Testing Load Balanced Services .................................................................................. 54
     - Diagnosing VIP Connection Problems .................................................................... 54
     - Taking Real Servers Offline ..................................................................................... 55
     - Using Reports & Log Files ....................................................................................... 56
10. More Information ........................................................................................................... 56
11. loadbalancer.org Technical Support .............................................................................. 56
12. Company Contact Information ...................................................................................... 57
1. Introduction

Microsoft Azure is a comprehensive set of cloud services that developers and IT professionals use to build, deploy and manage applications through Microsoft's global network of data centers. It allows services to be deployed as and when required. Charges are made for what is used making it an extremely flexible and cost effective solution. The Loadbalancer.org Enterprise Azure cloud based load balancer allows customers to rapidly deploy and configure a feature rich load balancing solution within the Azure cloud.

2. About Enterprise Azure

Enterprise Azure is a fully featured Application Delivery Controller (ADC) / Load Balancer designed specifically for Azure. The core software is based on customized versions of Centos 6.x/RHEL 6.x, Linux 4.9.x, LVS, HA-Linux, HAProxy, Pound, STunnel & Ldirectord.

Enterprise Azure can be deployed as a single instance although we always recommend that 2 appliance's are deployed as an HA clustered pair to avoid introducing a single point of failure.

Enterprise Azure is based on the same code base as our main hardware/virtual product. This means that Enterprise Azure supports many of the same features as the hardware & virtual based products. There are certain differences due to the way the Microsoft Azure environment works. The main differences are listed below.

MAIN DIFFERENCES TO OUR STANDARD (NON-CLOUD) PRODUCT

- Layer 4 DR mode is currently **not** supported.
- In Azure, you should configure your HA pair **first** before setting up your load balanced services. This is different to the recommendation for our hardware/virtual products and is due to the way HA is handled. In our standard product, when a failover occurs, the same VIP address is brought up on the passive device. In Azure, in order to minimize the time taken for the failover a different approach is used. When creating a VIP on an Azure HA pair, 2 private IPs must be specified – one for the VIP when it’s active on the Master and one for the VIP when it’s active on the Slave. The IPs for the VIP on the Master & Slave are selected using drop-downs within the VIP configuration screen. An Azure load balancer is used in front of the Loadbalancer.org HA pair to direct inbound connections to the active appliance. Both Master & Slave appliances must be in the same Availability Set or deployed within the same Availability Zone / split across 2 different Availability Zones. Please refer to page 18 for more information on configuring an HA pair.

**Note:**

The private IPs for the VIP on the Master & Slave are selected using drop-downs within the VIP configuration screen. These drop-downs are only displayed **after** the pair is configured. They are populated with the IPs that are assigned to the network interface using the WebUI option: *Local Configuration > Network Interface Configuration*.

**Adding VIPs after creating an HA pair (RECOMMENDED)** – If you add VIPs after creating an HA pair, you’ll be prompted for both IPs. Add the IPs you intend to use for the VIPs to the local interface on both Master & Slave and they’ll be available in the drop-downs.

**Creating an HA pair after configuring VIPs on the Master** – If you add a Slave appliance and create an HA pair after adding VIPs to the Master appliance, the floating IPs that were automatically configured for each VIP must first be removed using the WebUI option: *Cluster Configuration > Floating IPs* and then added to the network interface instead. This will ensure that these IPs appear in the drop-downs mentioned above. You’ll also need to configure IPs in a similar way on the Slave device so that corresponding Slave IPs can be selected for each VIP using the drop-downs.
About Enterprise Azure

- Layer 4 NAT mode where the default gateway on the load balanced real servers is required to be the load balancer is not supported. Routing rules for the real server subnet must be changed instead. Please refer to the example on page 47 for more details.

- Layer 7 SNAT mode with TProxy enabled where the default gateway on the load balanced real servers is required to be the load balancer is not supported. Routing rules for the real server subnet must be changed instead. Please refer to the example on page 52 for more details.

WHY USE ENTERPRISE AZURE?
Microsoft Azure's load balancer provides basic load balancing functionality but is limited in several areas. Loadbalancer.org's Enterprise Azure load balancer provides the following additional features & advantages:

1. Supports comprehensive Layer 7 load balancing
2. Load balances both Azure based and non-Azure based servers
3. Supports Round Robin and Least Connection connection distribution algorithms
4. Supports customizable timeouts for custom applications beyond those offered by Azure
5. Supports comprehensive back-end server health-check options
6. Enables fallback servers to be configured and invoked when all load balanced servers/services fail
7. Provides extensive real time and historical statistics reports
8. Supports session distribution based on actual server load (utilizing Loadbalancer.org's feedback agent which is available for both Linux & Windows)
9. Supports SSL Termination
10. Supports Microsoft RDP Cookie based persistence
11. Supports full integration with Microsoft Remote Desktop Services Connection Broker

Note:
For a full feature comparison please refer to our Enterprise Azure product page.

3. Azure Deployment Models
The Azure platform currently supports both the original Classic model and the latest Resource Manager model. To simplify the deployment and management of resources, Microsoft recommends that the Resource Manager model is used for new resources, and, if possible, existing resources are re-deployed through Resource Manager. For a more detailed comparison of Classic and Resource Manager models, please refer to this URL.

4. Accessing Microsoft Azure
To start using Microsoft Azure, you'll need an Azure account. If you don't already have one you can create one at the following URL: https://azure.microsoft.com/en-gb/free/

5. Azure Management
Azure resources can be managed in 3 ways:

- Azure Portal
- Azure PowerShell
- Azure CLI
ACCESSING THE AZURE PORTAL
The Azure Portal is available here.

AZURE POWERSHELL & AZURE CLI
Information on how to obtain, install and configure PowerShell is available here.
Information on how to obtain, install and configure Azure CLI is available here.

6. Deploying Enterprise Azure From the Marketplace

1. Login to the Azure Portal
2. Select Virtual Machines
3. Click Add

Configure the Basics

Subscription *
Resource group *

Instance details
Virtual machine name *
Region *
Availability options
Availability set *
image *
Azure Spot instance
Size *

Administrator account
Authentication type
Username *
Password *
Confirm password *

1. Configure the Subscription & Resource group settings according to your requirements
2. Enter a suitable name for the instance, e.g. **LB1**
3. Select the required **Region**
4. Configure the **Availability options** according to your requirements

**Note:**
For an HA clustered pair, both VMs must be in the same Availability Set or deployed within the same Availability Zone / split across 2 different Availability Zones. Please refer to page 18 for more details on setting up an HA pair.

5. Select the required **Image** – to do this, click **See all images** then enter “Loadbalancer.org” in the Marketplace search box and hit <ENTER>
6. The following options are available:

   **Load Balancer ADC** – Fully featured appliance, hourly based billing
   - Load Balancer ADC – 10G (licensed for up to 20Gbps throughput)
   - Load Balancer ADC – 1G (licensed for up to 4Gbps throughput)

   **Load Balancer ADC – BYOL** – Fully featured appliance, one time payment (*)
   - Load Balancer ADC – BYOL (purchase & apply either the 1G or the 10G license)

(*) Azure infrastructure costs apply

**Note:**
The BYOL version will work completely unrestricted for 30 days without any license applied. During this period, only Azure usage charges will apply. After the 30 days, the trial will still function, but no configuration changes will be possible until the license is applied.

7. Select the required **Size** – this can be changed by expanding the drop-down and selecting from the recently used or recommended image sizes. Or alternatively by clicking **See all sizes** and choosing from the expanded list of options.

**Note:**
The Image size required depends on the anticipated workload. For more details on the image size that best suits your requirements, please refer to the following Microsoft links: [Sizes for Virtual Machines](#) and [Linux Virtual Machine Pricing](#). The standard A1 with 1 vCPU & 1.75GB RAM is primarily meant for PoC deployments and low traffic websites and application. For a non PoC deployment we recommend at least 2 vCPUs and 3.5GB RAM. For further help and advice please contact support@loadbalancer.org.

8. Select the required **Authentication type** – a **Password** or an **SSH Public key** can be used

**Note:**
Please refer to page 15 for more details on creating and using SSH keys.

9. If using password authentication, enter a suitable **Username & Password**
10. Click **Next : Disks** >
**Configure Disks**

1. Select the required *OS disk type* – by default this is set to *Standard SSD*, this can be changed if required. Typically, the default setting is appropriate for most deployments.

   **Note:**
   Information on the various disk types available in Azure can be found [here](#). Comparative disk pricing is available [here](#).

2. Select the required *Encryption type*

3. Click **Next : Networking**

**Configure Networking**
1. Configure the **Virtual network, Subnet & Public IP** settings according to your requirements

   **Note:**
   Public IP addresses can be either Basic or Standard SKU – the default is Basic. When setting up HA (see page 18), the Public IP SKU and the load balancer SKU must match to enable backend pools to be configured. For more information on setting up public IP addresses please click here.

2. Configure the **Network Security Group** settings according to your requirements

   **Note:**
   Microsoft recommends that where possible Network Security Groups are associated with subnets rather than individual interfaces since this simplifies management.

   - If you choose to create a new NSG rather than selecting an existing one, the following
inbound rules are included by default:

<table>
<thead>
<tr>
<th>Inbound rules</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1010: HTTP_access</td>
<td>Any</td>
</tr>
<tr>
<td>Custom (TCP/9080)</td>
<td>✓ ...</td>
</tr>
<tr>
<td>1020: HTTPS_access</td>
<td>Any</td>
</tr>
<tr>
<td>Custom (TCP/9443)</td>
<td>✓ ...</td>
</tr>
<tr>
<td>1030: default-allow-ssh</td>
<td>Any</td>
</tr>
<tr>
<td>SSH (TCP/22)</td>
<td>✓ ...</td>
</tr>
<tr>
<td>+ Add an inbound rule</td>
<td></td>
</tr>
</tbody>
</table>

- These inbound rules are required for managing the load balancer. If you’ll be deploying layer 7 services, TCP port 7777 can also be added – this allows the HAProxy statistics page to be viewed.
- The rules can be edited by clicking the *Create new* link under the network security group drop-down.
- Specify additional inbound rules for the ports used for your load balanced applications, e.g. TCP 80 and TCP 443 if you’re load balancing web servers, TCP 3389 if you’re load balancing RDP etc.
- To specify additional inbound rules, click *Add an inbound rule*. The example below shows additional ports TCP/80 (for load balanced HTTP web server traffic) and TCP/443 (for load balanced HTTPS web server traffic) and TCP/7777 (for HAProxy stats).

<table>
<thead>
<tr>
<th>Inbound rules</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1010: HTTP_access</td>
<td>Any</td>
</tr>
<tr>
<td>Custom (TCP/9080)</td>
<td>✓ ...</td>
</tr>
<tr>
<td>1020: HTTPS_access</td>
<td>Any</td>
</tr>
<tr>
<td>Custom (TCP/9443)</td>
<td>✓ ...</td>
</tr>
<tr>
<td>1030: default-allow-ssh</td>
<td>Any</td>
</tr>
<tr>
<td>SSH (TCP/22)</td>
<td>✓ ...</td>
</tr>
<tr>
<td>1040: Port_443</td>
<td>Any</td>
</tr>
<tr>
<td>Custom (Any/443)</td>
<td>✓ ...</td>
</tr>
<tr>
<td>1050: Port_7777</td>
<td>Any</td>
</tr>
<tr>
<td>Custom (Any/7777)</td>
<td>✓ ...</td>
</tr>
<tr>
<td>+ Add an inbound rule</td>
<td></td>
</tr>
</tbody>
</table>

**Note:**
The rules can also be edited after the NSG is created.

- Once the required rules have been defined, click **OK**.
3. Under the “Load Balancing” section, if you have already configured an Azure load balancer in preparation for configuring an HA pair, ensure the Place this virtual machine behind an existing load balancing solution? check-box is enabled, then set Load balancing options to **Azure load balancer** and select the relevant load balancer. Alternatively, the Azure load balancer can be created later, in which case leave the check-box unchecked.

**Note:**
When deploying an HA pair of Loadbalancer.org instances, they must be deployed behind an Azure load balancer. For more information please refer to page 18.

4. Click **Next : Management >**

**Configure Management**

1. Configure the Management Settings according to your requirements
2. Click **Next : Advanced >**

**Configure Advanced Options**

1. Configure the Advanced Settings according to your requirements
2. Click **Next : Tags >**

**Configure Tags**

![Configured Tags Table]

1. Configure Tags according to your requirements
2. Click **Next : Review & Create >**

**Review & Create**

1. Review all details, terms and settings, enter your *Name*, *Preferred e-mail address* and *Preferred phone number* in the fields provided and if you’re happy to proceed, click **Create**.

== If validation completes successfully, the load balancer instance will now be deployed ==

**Enable IP Forwarding (If Required)**

If you’ll be configuring layer 4 NAT mode services, or layer 7 services with TProxy, ensure that IP forwarding is enabled. This allows the VM to accept traffic that is not addressed to itself, i.e. the return traffic from the load balanced servers to the client that passes via the load balancer. For an HA pair, this must be done on both VMs.

To enable IP forwarding:
1. In the Azure Management Portal, select the Virtual Machines option, click on the newly deployed Load Balancer VM, click on Networking and then select the network interface attached to the load balancer, then click IP configurations.

2. Ensure that IP forwarding is enabled as shown below:

![IP forwarding settings](image)

7. Accessing the Appliance

ACCESSING THE APPLIANCE USING THE WEBUI

In a browser, navigate to the Public IP address or FQDN on port 9443, i.e.

https://<Public IP Address>:9443

or

https://<FQDN>:9443

Note:
To configure an FQDN in Azure under the Resource Manager model please refer to this link.

You'll receive a warning about the certificate as it’s a self signed cert not related to an Internet based CA. Confirm you want to continue and a login prompt will be displayed. Use the following default credentials:

Username: loadbalancer
Password: loadbalancer

Note:
To change the password for the 'loadbalancer' account, use the WebUI option: Maintenance > Passwords.

Once logged in, the WebUI is displayed:
WEBUI MENU OPTIONS

The main menu options are as follows:

**System Overview** – Displays a graphical summary of all VIPs, RIPS and key appliance statistics

**Local Configuration** – Configure local host settings such as DNS, Date & 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
Accessing the Appliance

APPLIANCE SECURITY
To control how the appliance is accessed and which features are enabled, 3 security modes are provided:

- **Secure** – this is the default mode. In this mode:
  - the WebUI is accessible on HTTPS port **9443**. If you attempt to access the WebUI on HTTP port **9080** you will be redirected to HTTPS port **9443**
  - access to the "Execute Shell Command" menu option is disabled
  - the ability to edit the firewall script & the lockdown wizard is disabled
  - ‘root’ user console & SSH password access are disabled

- **Custom** – In this mode, the security options can be configured to suit your requirements

- **Secure – Permanent** – this mode is the same as Secure, but the change is **irreversible**

**IMPORTANT:**
Only set the security mode to **Secure – Permanent** if you are 100% sure this is what you want!

To configure the Security Mode:

1. Using the WebUI, navigate to: **Local Configuration > Security**
2. Select the required **Appliance Security Mode**
3. If **Custom** is selected, configure the other options to suit your requirements
4. Click **Update**

**Note:**
For full details of all options, please refer to the [Administration Manual](#) and search for “Appliance Security Options”.

Default Password
We strongly recommend that the default ‘loadbalancer’ WebUI account password is changed as soon as the appliance is deployed. This can be changed using the WebUI menu option: **Maintenance > Passwords**.

CHECKING FOR UPDATES
Once you have access to the WebUI, we recommend that you use the online update feature to ensure that you're running the very latest version of the appliance. To check for updates, use the WebUI option: **Maintenance > Software Update** and click the **Online Update** button. If updates are available, you'll be presented with a list of changes that are included in the update. To start the update, click the second **Online Update** button at the bottom of the screen. Updates are incremental, so repeat the process until you're informed that no more updates are available.

APPLIANCE LICENSING
If you've deployed the BYOL version of the appliance, by default it runs as a 30 day trial and is completely unrestricted during this time. After 30 days, the appliance continues to work but it's no longer possible to make changes to the configuration. When a license is purchased, you'll be provided with a license key file by our sales team. This must then be installed on your appliance. To install the license, use the WebUI option: **Local Configuration > License Key** to browse to and select the license file provided. Once selected, click **Install License Key** to apply the license. We recommend that you should check for updates before applying the license key.
ENTERPRISE AZURE NON-STANDARD WEBUI MENU OPTIONS

Enterprise Azure has some differences to the standard hardware/virtual product range due to the way the Microsoft Azure environment works. The menu options that work differently are detailed below. For all others please refer to our main Administration Manual.

1) Local Configuration > Network Interface Configuration

This menu option works in a very similar way to the standard product range, although please note the following:

- On initial deployment, a single IP private address is allocated (either static or dynamic depending on the chosen setting)
- Additional addresses can be added as shown (10.1.6.10/24) – this is required when you require multiple VIPs on different IP addresses
- To add an additional IP address, enter the new address below the existing address as shown in the example above, then click Configure Interfaces

IMPORTANT:
If an IP address is added, you’ll also need to add the same IP address to the Network Interface on the load balancer VM via the Azure portal. If this is not done, Azure will not be aware of the new address.

IMPORTANT:
If the IP address allocated to the VM on initial deployment (normally the first in the list) is changed, make sure that you add the same address to the VMs in Azure. If this is not done, you’ll loose connectivity to the VM.

ACCESSING THE APPLIANCE USING SSH

When the appliance is deployed, Authentication type must be set to either SSH Public key or Password. When set to SSH Public Key, a key pair must be manually generated outside of the Azure environment using tools such as ssh-keygen under Linux and PuttyGen under Windows. Once the key pair is generated, the public key must be copied into the SSH public key field at VM deployment, and the private key is then used on the SSH client machine to access the VM.
GENERATING SSH KEYS
The steps below show how to generate SSH key pairs using Linux and Windows.

Using Linux

Generate a keypair using ssh-keygen

All Distros:

```
# ssh-keygen -q -t rsa -b 2048 -f <output filename>
```

e.g.

```
# ssh-keygen -q -t rsa -b 2048 -f AzureKeys
```

When prompted, enter a pass-phrase, or leave empty for no passphrase:

Enter passphrase (empty for no passphrase):
Enter same passphrase again:

2 files are created:

- **AzureKeys** – this is the Private Key file and is used on the SSH client machine
- **AzureKeys.pub** – this is the Public Key file, the contents are copied into the SSH public key field when the VM is deployed

Using Windows

**STEP 1 – Install PuTTY**

1. Download PuTTY from [here](#)
2. Run the installer

**STEP 2 – Use PuTTYgen to generate a Public/Private key pair**

1. Browse to the PuTTY program folder and run PuTTYgen
2. Click the **Generate** button
3. As directed, move the mouse around to create random keys
4. Once generated, click the **Save public key** and **Save private key** buttons to save the keys

**ACCESSING THE APPLIANCE FROM LINUX**

Start SSH specifying the private key file and login as the user defined when deploying the VM, e.g.

Using the IP address:

```
# ssh -i /root/AzureKeys lbuser@1.2.3.4
```

Or using the fqdn:

```
# ssh -i /root/AzureKeys lbuser@fqdn
```

**Note:** To configure an FQDN in Azure under the Resource Manager model please refer to [this link](#).

**ACCESSING THE APPLIANCE FROM WINDOWS USING PUTTY**

1. Run PuTTY
2. Expand the SSH section and select **Auth** as shown below
3. Click **Browse** and select the private key created earlier
4. Click **Open** to start the SSH session
5. Login using the username specified when deploying the instance, no password will be required

**Note:**
To enable full root access, the following command can be used once logged in to the appliance via SSH:  
$ sudo su
8. Configuration Examples

This section presents 4 example configurations that illustrate how the load balancer is deployed. Web Servers are used in the examples, although the same concepts apply to other applications.

HIGH AVAILABILITY

We recommend that 2 appliance’s are deployed as an HA clustered pair to avoid introducing a single point of failure. In Azure, you should configure your HA pair **FIRST** before setting up your load balanced services.

KEY CONCEPTS

- In our standard hardware / virtual product, when a failover occurs, the *same* VIP address is brought up on the passive device. In Azure, in order to minimize the time taken for the failover a different approach is used.
- In Azure, when creating a VIP on an HA pair, 2 private IPs must be specified – one for the VIP when it’s active on the Master and one for the VIP when it’s active on the Slave.
- An Azure load balancer is used in front of the Loadbalancer.org HA pair to direct inbound connections to the **active** appliance.
- Both Master & Slave appliances must be in the same Availability Set or deployed within the same Availability Zone / split across 2 different Availability Zones.

**Note:**

If you’re using Availability Zones rather than Availability Sets, then a standard SKU Azure load balancer must be used. The standard SKU has a published SLA and the cost depends on the amount of data processed and the number of load balancing rules as explained here. The Basic SKU is free, but comes with no published Azure SLA.

If you have already configured a single appliance and now want to add a Slave appliance to create an HA pair, the following approach can be taken:

**Note:**

If your existing appliance has not been deployed in an Availability Set or Zone, you’ll have to re-deploy the instance because instances cannot be moved into a set or zone after creation. In this case, please refer to Example 1 on page 20 or Example 2 on page 32.

1. Deploy an additional Loadbalancer.org VM to be the Slave, make sure it’s in the same Availability Set or deployed within the same Availability Zone / split across 2 different Availability Zones.
2. On the Master appliance, using the WebUI option: **Cluster Configuration > Floating IPs**, remove the floating IPs that were automatically configured for each VIP, then using the WebUI option: **Local Configuration > Network Interface Configuration** add the same addresses to the network interface instead. This will ensure that these IPs appear in the drop-downs used to setup the VIP(s) when in HA mode as illustrated in step 7 on page 30.
3. On the Slave appliance, using the WebUI option: **Local Configuration > Network Interface Configuration** add corresponding IPs for each VIP. This will ensure that these IPs appear in the drop-downs used to setup the VIP(s) when in HA mode.
4. Follow Step 2 to 6 starting on page 20.
5. Using the WUI on the Master appliance, ensure that the Virtual Service IP address & Slave IP Address fields are set correctly for each VIP. As explained on page 3, these drop-downs are used to specify the IP address used for the VIP when active on the Master and when active on the Slave.
6. Follow Step 7 on page 29.
8. Follow Step 11 on page 32.

IMPLEMENTING HA IN AZURE

The following diagram shows how HA is configured in Azure. As shown, two Loadbalancer.org VMs are configured as a clustered pair in combination with an Azure load balancer.
• LB1 and LB2 are configured as an HA pair. In this mode, one device is active (typically the Master appliance) and the other is passive (typically the Slave appliance).

• The private IPs for the VIP on the Master & Slave are selected using drop-downs within the VIP configuration screen. These drop-downs are only displayed once the pair is configured. They are populated with the IPs that are assigned to the network interface using the WebUI option: Local Configuration > Network Interface Configuration.

• The probe service on TCP port 6694 is up on the active appliance (LB1) and down on the passive appliance (normally LB2). The active appliance responds with 200 OK.

• The Azure load balancer probes port 6694 on LB1 and LB2 and then forwards traffic to the active load balancer appliance (normally LB1).

• If the Master appliance fails for any reason, the passive appliance will detect this, become active and bring up the probe service on port 6694. In turn, the Azure load balancer detects this and will then forward traffic to the Slave device (LB2).

• If your configuration includes VIPs with multiple ports or if you have multiple VIPs you’ll need to setup multiple Load balancing rules to map from the Azure load balancer’s Frontend IP to the appropriate Backend Pool and appropriate port. Also, you may need to setup multiple Frontend IP Configurations & Backend Pools depending on whether your VIPs share the same IP or have unique IP addresses, and whether the load balanced servers are common between VIPs or unique. The same Health-probe should be used for all Load balancing rules.

1 – LOAD BALANCING WEB SERVERS, HA CONFIGURATION, 1 SUBNET, LAYER 7
This example demonstrates how to configure an HA pair of load balancers and then configure a layer 7 VIP to load balance 2 web servers. The Loadbalancer.org instances are deployed in the same Availability Set and a Basic SKU Azure load balancer is used to route connections to the active appliance.

Step 1 – Deploy 2 x Loadbalancer.org VMs – one to be the Master, the other to be the Slave

1. Please refer to the steps starting on page 5. Ensure that both load balancer VMs are in the same Availability Set

Step 2 – Verify Network Security Group Settings

1. Ensure that your Network Security Group(s) permit the following communication between the 2 VMs:
   • TCP port 22 (SSH)
   • UDP port 6694 (heartbeat)
   • ICMP Ping

   Note:
   These requirements are covered by default within the same Virtual Network. Please refer to this link for more information on default rules.

2. Ensure that your Network Security Group(s) permit the following inbound communication from the Azure load balancer to both VMs:
   • TCP port 6694 (Azure load balancer health probe)
Step 3 – Add the IP Address to be used for the VIP to the Master & Slave using the Azure Portal

1. In the Azure Portal select Virtual Machines
2. Select the Master VM
3. Select Networking, then click the Network Interface
4. Select IP Configurations
5. Click Add

- Enter a suitable name for the IP address, e.g. WebCluster1
- Set Private IP address Allocation to Static
- Enter an appropriate IP address (this must tally with the address to be used for your VIP on the Loadbalancer.org appliance), e.g. 10.1.3.80
- Click OK

6. Now repeat steps 1-5 on the Slave VM for the VIP using a corresponding (but different) IP address, e.g. 10.1.3.81

Step 4 – Add the IP Address to be used for the VIP to the Master & Slave using the Appliance WebUI

Master Appliance
- On the Master, navigate to the WebUI option: Local Configuration > Network Interface Configuration
- Add the IP you intend to use for the VIP, use CIDR notation, e.g. 10.1.3.80/24
Slave Appliance
- On the Slave, navigate to the WebUI option: Local Configuration > Network Interface Configuration
- Add the IP you intend to use for the VIP, use CIDR notation, e.g. 10.1.3.81/24

Step 5 – Add & Configure the Azure Load Balancer

1. First, add the Azure Load Balancer
   - In the Azure Portal select Load balancers
   - Click Add

   ![Azure Load Balancer Configuration](image)

   - Configure the Subscription & Resource group settings according to your requirements
   - Enter a suitable Name for the instance, e.g. AzureLB
   - Select the required Region
   - If deploying within a private network set Type to Internal, if it's public facing select Public
   - Set SKU to Basic

   Note:
   This configuration example uses a Basic SKU Azure load balancer. Configuration Example 2 on page 32 uses a Standard SKU Azure load balancer. Please note the following when
deciding which option to choose:

- If you have deployed your Master & Slave instances in an Availability Set then either the Basic or Standard SKU can be used
- If you have deployed your Master & Slave instances using Availability Zones, then the Standard SKU must be used
- The Basic SKU is free, rates apply to the Standard SKU
- The Standard SKU has a published SLA, the Basic SKU does not
- Microsoft recommends the Standard SKU load balancer for production workloads

- Configure the Public IP address settings (for external deployments) or the Virtual Network settings (for internal deployments) according to your requirements

**Note:**

Once the Azure Load balancer is created, the IP configuration can be modified using the Frontend IP Configuration in the Load balancer menu.

- Click Next : Tags >
- Configure the Tags according to your requirements
- Click Next : Review + Create >
- Once validated, review the settings and click Create

2. Next, create the Backend Pool(s)

- In the menu for the load balancer, click Backend pools
- Click Add
Configuration Examples

- Enter an appropriate Name, e.g. **WebCluster1**
- Select the required Virtual Network
- Under Associated to select Virtual machines
- Select the required IP version, e.g. IPv4
- Click **Add**

- Ensure that both load balancer VMs are selected as shown in the example above (LB1 & LB2), and that the IP addresses selected correspond to the VIP on each appliance (10.1.3.80 & 10.1.3.81)
- Click **Add**
### Virtual machines

You can only attach virtual machines in ulsoucrh that have a basic SKU public IP configuration or no public IP configuration. All virtual machines must be in the same availability set and all IP configurations must be on the same virtual network.

<table>
<thead>
<tr>
<th>Virtual machine</th>
<th>IP Configuration</th>
<th>Availability set</th>
</tr>
</thead>
<tbody>
<tr>
<td>LB1</td>
<td>WebCluster1 (10.1.3.80)</td>
<td>AS1</td>
</tr>
<tr>
<td>LB2</td>
<td>WebCluster1 (10.1.3.81)</td>
<td>AS1</td>
</tr>
</tbody>
</table>

- Click **Add** again to confirm your selection
- All settings will now be validated and the Backend Pool will be created

**Note:**

If you have multiple VIPs on different IPs you’ll need to setup a Backend Pool for each of these. This is illustrated in the diagram on page 19.

### 3. Next, create a Health Probe

- In the menu for the Load balancer, click **Health-probes**
- Click **Add**

<table>
<thead>
<tr>
<th>Name *</th>
<th>LB-Probe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protocol *</td>
<td>HTTP</td>
</tr>
<tr>
<td>Port *</td>
<td>6694</td>
</tr>
<tr>
<td>Path *</td>
<td>/</td>
</tr>
<tr>
<td>Interval *</td>
<td>5</td>
</tr>
<tr>
<td>Unhealthy threshold *</td>
<td>2</td>
</tr>
</tbody>
</table>

**Note:**

Setting **Protocol** to **HTTP** will configure the Azure load balancer to look for a **200 OK** response from each Loadbalancer.org VM.

- Set **Port** to **6694**
- Leave the remaining settings at their default values
4. Next, configure the Load Balancing Rule

- In the menu for the Load balancer, click *Load balancing rules*
- Click *Add*

```
Name *                      WebCluster1-80
IP Version *                IPv4  IPv6
Frontend IP address * 1     loadbalancerFrontEnd [51.132.22.253]  
Protocol                   TCP  UDP
Port *                     80   
Backend port * 1           80   
Backend pool * 1           WebCluster1  
Health probe * 1           LB-Probe [HTTP:6694]  
Session persistence 1      None  
Idle timeout (minutes) * 1  4   
Floating IP 1              Disabled  Enabled
```

- Enter an appropriate *Name*, e.g. *WebCluster1-80*
- Select the required *IP version*, e.g. *IPv4*
- Select the required *Frontend IP address*
- Set the *Protocol* to *TCP*
- Set the *Port* to the required value, e.g. *80*
- Set the *Backend port* to the required value, e.g. *80*
- Select the *Backend pool* created previously
- Select the *Health Probe* created previously
- Leave *Session persistence* set to *None* – session persistence is not required since the Azure Load balancer will simply send all traffic to the working Loadbalancer.org appliance, i.e. the appliance that is responding with a *200 OK* to the HTTP probe on TCP port 6694
- Click *Add*
Configuration Examples

Note:
If your configuration includes other ports (e.g. HTTPS port 443) or if you have multiple VIPs you’ll need to setup multiple **Load balancing rules** to map from the Azure load balancer’s Frontend IP to the appropriate **Backend Pool** and appropriate port. Also, you may need to setup multiple **Frontend IP Configurations & Backend Pools** depending on whether your VIPs share the same IP or have unique IP addresses, and whether the load balanced servers are common between VIPs or unique. The same **Health-probe** should be used for all **Load balancing rules**. This is illustrated in the diagram on page 19.

5. Next, configure the Inbound NAT Rules (if required) to enable VM access via the Azure Load balancer

Note:
Each load balancer retains it’s original public IP address by default and this is displayed in the Azure Portal > Home > Virtual Machines page in the Public IP Address column. If you continue to drill down into each VM, the same Public IP address is shown for both load balancers, this is the Front end IP address of the Azure load balancer. If preferred, the individual IP addresses can be removed, leaving just the front end IP address of the load balancer. In this case, NAT rules can be used to provide access to the individual load balancer VM’s that are located behind the Azure load balancer in an HA configuration.

The example below shows how to setup a rule to allow SSH access (TCP port 22) to the Master Loadbalancer.org VM via the Azure load balancer public IP address on TCP port 122.

- In the menu for the Load balancer, click **Inbound NAT rules**
- Click **Add**

<table>
<thead>
<tr>
<th>Name *</th>
<th>LB1-SSH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frontend IP address *</td>
<td>LoadBalancerFrontEnd (51.132.22.233)</td>
</tr>
<tr>
<td>IP Version</td>
<td>IPv4</td>
</tr>
<tr>
<td>Service *</td>
<td>Custom</td>
</tr>
<tr>
<td>Protocol</td>
<td>TCP</td>
</tr>
<tr>
<td>Idle timeout (minutes)</td>
<td>4</td>
</tr>
<tr>
<td>Port *</td>
<td>122</td>
</tr>
<tr>
<td>Target virtual machine</td>
<td>LB1 (Q5-RG1)</td>
</tr>
<tr>
<td>Network IP configuration</td>
<td>ipconfig1 (10.1.3.8)</td>
</tr>
<tr>
<td>Port mapping</td>
<td>Custom</td>
</tr>
<tr>
<td>Floating IP</td>
<td>Disabled</td>
</tr>
<tr>
<td>Target port *</td>
<td>22</td>
</tr>
</tbody>
</table>

- Enter an appropriate name, e.g. **LB1-SSH**
• Ensure the Frontend IP address is set to the correct address (IP’s from other Azure Load balancers will be included in the list)
• Set Service to Custom
• Set Protocol to TCP
• Set Port to 122
• Set Target Virtual Machine to the Master Loadbalancer.org appliance, e.g. LB1
• Set Network IP Configuration to the Interface on the Master Loadbalancer.org appliance
• Set Port Mapping to Custom
• Set Target port to 22
• Click Add

Note:
Don’t forget to modify the inbound rules on the appropriate Network Security Group to allow connections to the relevant target port, in this example TCP port 22.

Note:
Rules to access other Loadbalancer.org management ports can be added as required. The table below shows example rules and ports that can be configured to access SSH and the WebUI on both appliances. The configuration for the first rule listed is covered above.

Example rules for Master and Slave management

The table below shows example NAT rules that can be used to enable access to SSH and the WebUI on both the Master and Slave appliances.

<table>
<thead>
<tr>
<th>Rule Name</th>
<th>Port</th>
<th>Target Port</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>LB1-SSH</td>
<td>122</td>
<td>22</td>
<td>external access to SSH on LB1</td>
</tr>
<tr>
<td>LB2-SSH</td>
<td>222</td>
<td>22</td>
<td>external access to SSH on LB2</td>
</tr>
<tr>
<td>LB1-WebUI</td>
<td>19443</td>
<td>9443</td>
<td>external access to WebUI on LB1</td>
</tr>
<tr>
<td>LB2-WebUI</td>
<td>29443</td>
<td>9443</td>
<td>external access to WebUI on LB2</td>
</tr>
</tbody>
</table>

Step 6 – Configure the HA Clustered Pair

1. Open the WebUI on the Master appliance
2. Navigate to: Cluster Configuration > High Availability Configuration
3. In the **IP address of new peer** field, enter the Slave appliance's private IP address

4. In the **Password for loadbalancer user on peer** field enter the relevant password, the default password is 'loadbalancer'

5. Click **Add new node**

6. Once the pairing configuration has finished, any service restart messages and the confirmed pair message will be displayed as shown below:

7. Restart the services using the buttons presented, in this case Heartbeat

**Step 7 – Configure the Master appliance to allow service control during failover / fail-back**

1. On the Master appliance, navigate to: **Cluster configuration > Floating IPs**

2. In the **New Floating IP** field define an unused IP address in the same subnet as the appliances – this address is not used for any connections, it's required to allow service control on both Master & Slave units

   **Note:**
   The chosen IP address should not be in use anywhere else in the deployment.
3. Click **Add Floating IP**

**Step 8 – Configure the Virtual Service (VIP)**

**Note:**
If you plan on adding a custom Layer manual 7 VIP please see example 3 on page 45.

1. On the Master appliance, navigate to: Cluster Configuration > Layer 7 – Virtual Services and click **Add a New Virtual Service**
2. Enter the following details:

   ![Virtual Service Configuration](image)

   3. Enter an appropriate label for the VIP, e.g. **WebCluster1**
   4. Set the **IP Address** field to the IP address of the VIP when active on the Master appliance (the same address as added earlier in steps 3 & 4), e.g. **10.1.3.80**
   5. Set the **Slave IP Address** field to the IP address of the VIP when active on the Slave appliance (same address added earlier in steps 3 & 4), e.g. **10.1.3.81**

   **Note:**
   To assign additional IP addresses to the appliance, use the WebUI option: Local Configuration > Network Interface Configuration. If an IP address is added, you'll also need to add the same IP address to the Network Interface on the load balancer VM via the Azure portal.

   6. Set the **Virtual Service Ports** field to **80**
   7. Leave **Layer 7 Protocol** set to **HTTP**
   8. Click **Update**

**Step 9 – Configure the Real Servers (RIPs)**

1. On the Master appliance, navigate to: Cluster Configuration > Layer 7 – Real Servers and click **Add a new Real Server** next to the newly created VIP
2. Enter the following details:
3. Enter an appropriate label for the RIP, e.g. **Web1**
4. Change the *Real Server IP Address* field to the required IP address, e.g. **10.1.3.11**
5. Set the *Real Server Port* field to **80**
6. Click **Update**
7. Repeat the above steps to add your other Web Server(s)

**Step 10 – Verify synchronization state**

1. Verify that the status on the Master & Slave is as follows:

**Master Unit:**

```
Master | Slave   | Active | Passive | Link
-------|---------|--------|---------|-------
```

**Slave Unit:**

```
Master | Slave   | Active | Passive | Link
-------|---------|--------|---------|-------
```

The Slave can be made active by clicking **[Advanced]** in the green box, and then clicking the **Take over** button

**Other states:**

<table>
<thead>
<tr>
<th>Master</th>
<th>Slave</th>
<th>Active</th>
<th>Passive</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Master</td>
<td>Slave</td>
<td>Active</td>
<td>Passive</td>
<td>Link</td>
</tr>
<tr>
<td>Master</td>
<td>Slave</td>
<td>Active</td>
<td>Passive</td>
<td>Link</td>
</tr>
</tbody>
</table>

- **Master | Slave | Active | Passive | Link**
  - this is a Master unit, it's active, no Slave unit has been defined
- **Master | Slave | Active | Passive | Link**
  - this is a Master unit, it’s active, a Slave has been defined but the link to the Slave is down. **Action check & verify the heartbeat configuration**
- **Master | Slave | Active | Passive | Link**
  - this is a Slave unit, it’s active (a failover from the Master...
Configuration Examples

Step 11 – Testing & Verification

• Browse to the public IP address of the Azure load balancer on port 80 and port 443, i.e.

http://<Public IP Address of Azure Load Balancer>
and
https://<Public IP Address of Azure Load Balancer>

• To verify failover to the passive device (typically the Slave) click the Take Over button on the passive device and verify that load balanced services are still available via the now active Slave appliance – the failover time will depend on the settings for the health probe, but using default values it should complete in under 10 seconds.

2 – LOAD BALANCING WEB SERVERS, HA CONFIGURATION, 1 SUBNET, LAYER 7 WITH SSL TERMINATION

This is similar to the first example with the addition of SSL termination on the load balancer. We generally recommend that SSL should be terminated on the backend servers rather than the load balancer for scalability reasons, although in some cases terminating on the load balancer may be preferred. In this example, the Loadbalancer.org instances are deployed across 2 Availability Zones and a Standard SKU Azure load balancer is used to route connections to the active appliance.

Step 1 – Deploy 2 x Loadbalancer.org VMs – one to be the Master, the other to be the Slave

1. Please refer to the steps starting on page 5. Set the Availability Option to Availability Zone and set the Availability zone number according to your requirements. For example, select Availability Zone 1 for the Master VM and Availability Zone 2 for the Slave VM.

Note:
Some Images may not be supported within certain Availability Zones. If this is the case, please click Select size and choose an alternative option. For example, the Standard_A1_v2 image can be used instead of the Standard_A1 image.

Step 2 – Verify Network Security Group Settings

1. Ensure that your Network Security Group(s) permit the following communication between the 2 VMs:

• TCP port 22 (SSH)
• UDP port 6694 (heartbeat)
• ICMP Ping

Note:
These requirements are covered by default within the same Virtual Network. Please refer to this link for more information on default rules.
2. Ensure that your Network Security Group(s) permit the following inbound communication from the Azure load balancer to both VMs:

- TCP port 6694 (Azure load balancer health probe)

Note:
This requirement is covered by default within the same Virtual Network. Please refer to this link for more information on default rules.

**Step 3 – Add the IP Address to be used for the VIP to the Master & Slave using the Azure Portal**

1. In the Azure Portal select *Virtual Machines*
2. Select the Master VM
3. Select *Networking*, then click the Network Interface
4. Select *IP Configurations*
5. Click *Add*

<table>
<thead>
<tr>
<th>Name</th>
<th>WebCluster1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Primary</td>
</tr>
<tr>
<td></td>
<td>Secondary</td>
</tr>
</tbody>
</table>

- Primary IP configuration already exists

**Private IP address settings**

- Allocation: Dynamic, Static

**IP address**

- 10.1.3.80

- Click *OK*

6. Now repeat steps 1-5 on the Slave VM for the VIP using a corresponding (but different) IP address, e.g. 10.1.3.81
Configuration Examples

Step 4 – Add the IP Address to be used for the VIP to the Master & Slave using the **Appliance WebUI**

**Master Appliance**
- On the Master, navigate to the WebUI option: *Local Configuration > Network Interface Configuration*
- Add the IP you intend to use for the VIP, use CIDR notation, e.g. **10.1.3.80/24**

**Slave Appliance**
- On the Slave, navigate to the WebUI option: *Local Configuration > Network Interface Configuration*
- Add the IP you intend to use for the VIP, use CIDR notation, e.g. **10.1.3.81/24**

Step 5 – Add & Configure the Azure Load Balancer

1. First, add the Azure Load Balancer
   - In the Azure Portal select **Load balancers**
   - Click **Add**

![Load Balancer Configuration Form](image-url)

- **Subscription**: LoadBalancer.org Pay-As-You-Go
- **Resource group**: QS-rg1
- **Name**: AzureLB
- **Region**: (Europe) UK South
- **Type**
  - Public
- **SKU**
  - Standard
- **Tier**
  - Regional
- **Public IP address**
  - **Create new**: Use existing
- **Public IP address name**: WebClusterPublicIP
- **IP address assignment**
  - Dynamic
- **Availability zone**
  - Zone-redundant
- **Add a public IPv6 address**
  - No
- **Routing preference**
  - Microsoft network
• Configure the Subscription & Resource group settings according to your requirements

• Enter a suitable Name for the instance, e.g. AzureLB

• Select the required Region

• If deploying within a private network set Type to Internal, if it's public facing select Public

• Set SKU to Standard

Note:
This configuration example uses a Standard SKU Azure load balancer. Configuration Example 1 on page 20 uses a Basic SKU Azure load balancer.

Please note the following when deciding which option to choose:

• If you have deployed your Master & Slave instances in an Availability Set then either the Basic or Standard SKU can be used

• If you have deployed your Master & Slave instances using Availability Zones, then the Standard SKU must be used

• The Basic SKU is free, rates apply to the Standard SKU

• The Standard SKU has a published SLA, the Basic SKU does not

• Microsoft recommends the Standard SKU load balancer for production workloads

• Set the Tier according to your requirements

• Configure the Public IP address settings (for external deployments) or the Virtual Network settings (for internal deployments) according to your requirements

Note:
Once the Azure Load balancer is created, the IP configuration can be modified using the Frontend IP Configuration in the Load balancer menu.

• If you have deployed your Master and Slave VM's in different Availability Zones, set Availability Zone to Zone Redundant, if you have deployed your VM’s in the same Availability Zone, set Availability Zone to the same zone

• Click Next : Tags >

• Configure the Tags according to your requirements

• Click Next : Review + Create >

• Once validated, review the settings and click Create

2. Next, create the Backend Pool(s)

• In the menu for the load balancer, click Backend pools

• Click Add
Configuration Examples

- Enter an appropriate **Name**, e.g. **WebCluster1**
- Select the required **Virtual Network**
- Leave **Backend Pool Configuration** set to **NIC**
- Select the required **IP version**, e.g. **IPv4**
- Click **Add** in the **Virtual Machines** section

- Ensure that both load balancer VMs are selected as shown in the example above (LB1 & LB2), and that the IP addresses selected correspond to the VIP on each appliance (**10.1.3.80** & **10.1.3.81**)
- Click **Add**
Virtual machines
You can only attach virtual machines in uksouth that have a standard SKU public IP configuration or no public IP configuration. All IP configurations must be on the same virtual network.

<table>
<thead>
<tr>
<th>Virtual machine</th>
<th>IP Configuration</th>
<th>Availability set</th>
</tr>
</thead>
<tbody>
<tr>
<td>ib2</td>
<td>WebCluster1 (10.1.3.81)</td>
<td>-</td>
</tr>
<tr>
<td>ib1</td>
<td>WebCluster1 (10.1.3.80)</td>
<td>-</td>
</tr>
</tbody>
</table>

- Click **Add** at the bottom of the screen to confirm your selections
- All settings will now be validated and the Backend Pool will be created

**Note:**
If you have multiple VIPs on different IPs you’ll need to setup a Backend Pool for each of these. This is illustrated in the diagram on page **19**.

3. Next, create a Health Probe

- In the menu for the Load balancer, click **Health-probes**
- Click **Add**

<table>
<thead>
<tr>
<th>Name</th>
<th>LB-Probe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protocol</td>
<td>HTTP</td>
</tr>
<tr>
<td>Port</td>
<td>6694</td>
</tr>
<tr>
<td>Path</td>
<td>/</td>
</tr>
<tr>
<td>Interval</td>
<td>5</td>
</tr>
<tr>
<td>Unhealthy threshold</td>
<td>2</td>
</tr>
</tbody>
</table>

**Note:**
Setting **Protocol** to **HTTP** will configure the Azure load balancer to look for a **200 OK** response from each Loadbalancer.org VM.

- Set **Port** to **6694**
- Leave the remaining settings at their default values
4. Next, configure the Load Balancing Rule

- In the menu for the Load balancer, click *Load balancing rules*
- Click *Add*

<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>WebCluster1-80</td>
</tr>
<tr>
<td>IP Version</td>
<td>IPv4</td>
</tr>
<tr>
<td>Frontend IP address</td>
<td>LoadBalancerFrontEnd (20.49.171.40)</td>
</tr>
<tr>
<td>Protocol</td>
<td>TCP</td>
</tr>
<tr>
<td>Port</td>
<td>80</td>
</tr>
<tr>
<td>Backend port</td>
<td>80</td>
</tr>
<tr>
<td>Backend pool</td>
<td>WebCluster1</td>
</tr>
<tr>
<td>Health probe</td>
<td>LB-Probe (HTTP:8080)</td>
</tr>
</tbody>
</table>

- Enter an appropriate name, e.g. **WebCluster1-80**
- Select the required IP version, e.g. **IPv4**
- Select the required **Frontend IP address**
- Set the **Protocol** to **TCP**
- Set the **Port** to the required value, e.g. **80**
- Set the **Backend port** to the required value, e.g. **80**
- Select the **Backend pool** created previously
- Select the **Health Probe** created previously
- Leave **Session persistence** set to **None** — session persistence is not required since the Azure Load balancer will simply send all traffic to the working Loadbalancer.org appliance, i.e. the
appliance that is responding with a **200 OK** to the HTTP probe on TCP port 6694

- Leave the remaining settings at their default values
- Click **OK**

**Note:**

If your configuration includes other ports (e.g. HTTPS port 443) or if you have multiple VIPs you’ll need to setup multiple *Load balancing rules* to map from the Azure load balancer’s Frontend IP to the appropriate *Backend Pool* and appropriate port. Also, you may need to setup multiple *Frontend IP Configurations & Backend Pools* depending on whether your VIPs share the same IP or have unique IP addresses, and whether the load balanced servers are common between VIPs or unique. The same *Health-probe* should be used for all *Load balancing rules*. This is illustrated in the diagram on page 19.

5. Next, configure the Inbound NAT Rules (if required) to enable VM access via the Azure Load balancer

**Note:**

Each load balancer retains it’s original public IP address by default and this is displayed in the *Azure Portal > Home > Virtual Machines* page in the *Public IP Address* column. If you continue to drill down into each VM, the *same* Public IP address is shown for both load balancers, this is the Front end IP address of the Azure load balancer.

If preferred, the individual IP addresses can be removed, leaving just the front end IP address of the load balancer. In this case, NAT rules can be used to provide access to the individual load balancer VM’s that are located behind the Azure load balancer in an HA configuration.

The example below shows how to setup a rule to allow SSH access (TCP port 22) to the Master Loadbalancer.org VM via the Azure load balancer public IP address on TCP port 122.

- In the menu for the Load balancer, click *Inbound NAT rules*
- Click **Add**
Enter an appropriate name, e.g. **LB1-SSH**

Ensure the *Frontend IP address* is set to the correct address (IP's from other Azure Load balancers will be included in the list)

- Set **Service** to Custom
- Set **Protocol** to TCP
- Set **Port** to **122**
- **Set Target Virtual Machine** to the Master Loadbalancer.org appliance, e.g. **LB1**
- Set **Network IP Configuration** to the Interface on the Master Loadbalancer.org appliance
- Set **Port Mapping** to Custom
- Set **Target port** to **22**
- Click Add

**Note:**
Don't forget to modify the inbound rules on the appropriate Network Security Group to allow connections to the relevant target port, in this example TCP port 22.

**Note:**
Rules to access other Loadbalancer.org management ports can be added as required. The table below shows example rules and ports that can be configured to access SSH and the WebUI on both appliances. The configuration for the first rule listed is covered above.

**Example rules for Master and Slave management**

The table below shows example NAT rules that can be used to enable access to SSH and the
WebUI on both the Master and Slave appliances.

<table>
<thead>
<tr>
<th>Rule Name</th>
<th>Port</th>
<th>Target Port</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>LB1-SSH</td>
<td>122</td>
<td>22</td>
<td>external access to SSH on LB1</td>
</tr>
<tr>
<td>LB2-SSH</td>
<td>222</td>
<td>22</td>
<td>external access to SSH on LB2</td>
</tr>
<tr>
<td>LB1-WebUI</td>
<td>19443</td>
<td>9443</td>
<td>external access to WebUI on LB1</td>
</tr>
<tr>
<td>LB2-WebUI</td>
<td>29443</td>
<td>9443</td>
<td>external access to WebUI on LB2</td>
</tr>
</tbody>
</table>

Step 6 – Configure the HA Clustered Pair

1. Open the WebUI on the Master appliance
2. Navigate to: Cluster Configuration > High Availability Configuration

![Create a Clustered Pair](image)

3. In the IP address of new peer field, enter the Slave appliance’s private IP address
4. In the Password for loadbalancer user on peer field enter the relevant password, the default password is ‘loadbalancer’
5. Click Add new node
6. Once the pairing configuration has finished, any service restart messages and the confirmed pair message will be displayed as shown below:

![Commit changes](image)

7. Restart the services using the buttons presented, in this case Heartbeat
Step 7 – Configure the Master appliance to allow service control during failover / fail-back

1. On the Master appliance, navigate to: Cluster configuration > Floating IPs

   ![New Floating IP](image)

2. In the New Floating IP field enter an unused IP address in the same subnet as the appliances – this address is not used for any connections, it’s required to allow service control on both Master & Slave units
3. Click Add Floating IP

Step 8 – Configure the Virtual Service (VIP)

Note:

If you plan on adding a custom Layer manual 7 VIP please see example 3 on page 45.

1. On the Master appliance, navigate to: Cluster Configuration > Layer 7 – Virtual Services and click Add a New Virtual Service
2. Enter the following details:

   ![Virtual Service](image)

3. Enter an appropriate label for the VIP, e.g. WebCluster1
4. Set the IP Address field to the IP address of the VIP when active on the Master appliance (the same address as added earlier in steps 3 & 4), e.g 10.1.3.80
5. Set the Slave IP Address field to the IP address of the VIP when active on the Slave appliance (same address added earlier in steps 3 & 4), e.g 10.1.3.81

Note:

To assign additional IP addresses to the appliance, use the WebUI option: Local Configuration > Network Interface Configuration. If an IP address is added, you’ll also need to add the same IP address to the Network Interface on the load balancer VM via the Azure portal.
6. Set the *Virtual Service Ports* field to **80**
7. Leave *Layer 7 Protocol* set to **HTTP**
8. Click **Update**

**Step 9 – Configure the Real Servers (RIPs)**

1. On the Master appliance, navigate to: *Cluster Configuration > Layer 7 – Real Servers* and click **Add a new Real Server** next to the newly created VIP
2. Enter the following details:

   ![Real Server Configuration Example]

   3. Enter an appropriate label for the RIP, e.g. **Web1**
   4. Change the *Real Server IP Address* field to the required IP address, e.g. **10.1.3.11**
   5. Set the *Real Server Port* field to **80**
   6. Click **Update**
   7. Repeat the above steps to add your other Web Server(s)

**Step 10 – Upload your SSL Certificate**

1. Using the WebUI, navigate to: *Cluster Configuration > SSL Certificate* and click **Add a new SSL Certificate**
2. Select **Upload prepared PEM/PFX file**
3. Enter an appropriate label (name) for the certificate, e.g. **Cert1**
4. Browse to and select the relevant certificate file
5. For PFX files, enter the **PFX File Password**
6. Click **Add Certificate**

   **Note:**
   You can also create a CSR on the load balancer. If this is required, select the **Create A New SSL Certificate (CSR)** option instead of **Upload prepared PEM/PFX file** in step 2 above. For additional information please refer to the [Administration Manual](#) and search for “Generating a CSR on the Load Balancer”.

**Step 11 – Configure SSL Termination**

1. Using the WebUI, navigate to: *Cluster Configuration > SSL Termination* and click **Add a New Virtual Service**
2. Enter a suitable label (name), e.g. **SSL-WebCluster1**.
3. Using the **IP Address** drop-down, set the address to the same address configured previously in Step 8 when active on the Master, e.g. **10.1.3.80**
4. Using the **Slave IP Address** drop-down, set the address to the same address configured previously in Step 8 when active on the Slave, e.g. **10.1.3.81**
5. Set the **Associated Virtual Service** drop-down to the VIP created previously in Step 8
6. Ensure the Backend Port is set to the same port that the VIP created previously listens on, e.g. **80**
7. Leave the **SSL Operation Mode** set to **High Security**
8. Select the **SSL Certificate** uploaded in step (d) above
9. Click **Update**

**Step 12 – Apply the New Settings**

- Once the configuration is complete:
  1. use the **Reload HAProxy** button at the top of the screen to commit the changes
  2. use the **Restart STunnel** button at the top of the screen to commit the changes

**Step 13 – Verify synchronization state**

1. Verify that the status on the Master & Slave is as follows:

   **Master Unit:**

   ![Master Status Table]

   **Slave Unit:**

   ![Slave Status Table]
The Slave can be made active by clicking [Advanced] in the green box, and then clicking the Take over button.

### Other states:

<table>
<thead>
<tr>
<th>Master</th>
<th>Slave</th>
<th>Active</th>
<th>Passive</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Master</td>
<td>Slave</td>
<td>Active</td>
<td>Passive</td>
<td>this is a Master unit, it's active, no Slave unit has been defined</td>
</tr>
<tr>
<td>Master</td>
<td>Slave</td>
<td>Active</td>
<td>Passive</td>
<td>this is a Master unit, it's active, a Slave has been defined but the link to the Slave is down. Action: check &amp; verify the heartbeat configuration</td>
</tr>
<tr>
<td>Master</td>
<td>Slave</td>
<td>Active</td>
<td>Passive</td>
<td>this is a Slave unit, it's active (a failover from the Master has occurred) and the heartbeat link to the Master has been established</td>
</tr>
</tbody>
</table>

### Step 14 – Testing & Verification

- Browse to the public IP address of the Azure load balancer on port 80 and port 443, i.e.
  
  http://<Public IP Address of Azure Load Balancer>
  
  and
  
  https://<Public IP Address of Azure Load Balancer>

- To verify failover to the passive device (typically the Slave) click the Take Over button on the passive device and verify that load balanced services are still available via the now active Slave appliance – the failover time will depend on the settings for the health probe, but using default values it should complete in under 10 seconds.

### 3 – LOAD BALANCING WEB SERVERS, HA CONFIGURATION, 1 SUBNET, LAYER 7 (MANUAL CONFIGURATION) WITH SSL TERMINATION

The configuration used in the example is very similar to Example 2. The difference is that the layer 7 VIP and associated RIPv2s are configured using a Layer 7 'Manual Configuration'.

Custom, manually configured Layer 7 services are useful when your configuration requires advanced HAProxy settings that are not directly supported by the WebUI when creating and modifying VIPs & RIPv2s.

This example demonstrates the key point for manual layer 7 based services which is that an additional HAProxy bind directive must be added to the VIP configuration. This additional bind directive allows the service to be brought up on a corresponding IP address on the Slave appliance should a failover occur. Since this example also has an SSL termination, an acl directive also needs to be added to allow failover to the Slave appliance as shown below.

1. Follow steps 1 to 7 in Example 2 starting on page 32
2. On the Master appliance, navigate to: Cluster Configuration > Layer 7 – Virtual Services and click
Add a New Virtual Service

3. Enter the following details:

4. Ensure that the Manual Configuration checkbox is ticked
5. Enter an appropriate label for the VIP, e.g. **WebCluster1**
6. Set the **IP Address** field to the IP address of the VIP when active on the **Master** appliance (the same address as added earlier in steps 3 & 4), e.g **10.1.3.80**
7. Set the **Slave IP Address** field to the IP address of the VIP when active on the **Slave** appliance (same address added earlier in steps 3 & 4), e.g **10.1.3.81**
8. Set the **Virtual Service Ports** field to **80**
9. Click **Update**
10. On the Master appliance, navigate to: **Cluster Configuration > Layer 7 – Real Servers** and click **Add a new Real Server** next to the newly created VIP
11. Enter the following details:

12. Enter an appropriate label for the RIP, e.g. **Web1**
13. Change the **Real Server IP Address** field to the required IP address, e.g. **10.1.3.11**
14. Set the **Real Server Port** field to **80**
15. Click **Update**
16. Repeat the above steps to add your other Web Server(s)
17. Now navigate to: **Cluster Configuration > Layer 7 – Manual configuration**
18. Paste the following configuration into the edit window under the last line of text:

```
listen WebCluster1
```
bind 10.1.3.80:80 transparent
bind 10.1.3.81:80 transparent  
mode http
balance leastconn
acl SSL-WebCluster1 src 10.1.3.80
acl SSL-WebCluster1 src 10.1.3.81  
tcp-request connection expect-proxy layer4 if SSL-WebCluster1
cookie SERVERID maxidle 30m maxlife 12h insert nocache indirect
server backup 127.0.0.1:9081 backup non-stick
option http-keep-alive
timeout http-request 5s
option forwardfor
timeout tunnel 1h
option redispatch
option abortonclose
maxconn 40000
server Web1 10.1.3.11:80 weight 100 cookie web1 check inter 4000 rise 2
fall 2 slowstart 8000 minconn 0 maxconn 0 on-marked-down shutdown-sessions
server Web2 10.1.3.12:80 weight 100 cookie web1 check inter 4000 rise 2
fall 2 slowstart 8000 minconn 0 maxconn 0 on-marked-down shutdown-sessions

Note 1:  This is an example, for more information please refer to the Administration Manual and search for “Layer 7 – Custom Configurations”
Note 2:  Additional bind statement for the VIP when active on the slave
Note 3:  Additional SSL termination related ACL statement when active on the slave
Note 4:  Make sure you use the same names, IP addresses and Ports as you did when graphically defining the VIP and RIPv

19. Click Update
20. Now follow steps 10 to 14 in Example 2 starting on page 43

4 – LOAD BALANCING WEB SERVERS, SINGLE APPLIANCE, 2 SUBNETS, LAYER 4 NAT MODE

This example uses 2 subnets – one public subnet for the load balancer and one private subnet for the web servers. The load balancer has a single network interface located in the first subnet. Routing rules for the second private subnet must be changed so that return traffic passes back via the load balancer. This is achieved by creating a custom routing table with the required rules, then associating this with the private subnet – this can now be done directly in the portal (step c below), previously PowerShell had to be used.

Note:
This configuration is currently not supported in HA mode. In this mode, the custom routing rules would need to be dynamically modified to route via the Slave appliance rather than the Master if a failover occurs. This is currently not supported.

Step 1 – Setting up Azure

Note
1) Deploy VM’s & Configure Network Security Groups:

1. Deploy the load balancer instance into the first (public) subnet as described in the section starting on page 5
2. Deploy your required web server VMs into the second (private) subnet
3. Configure Network Security Groups to permit the required traffic flows. Configure the following rules for the 2 subnets (assuming a public facing deployment):

   **Load Balancer (Public) Subnet**
   - Inbound rule – from 0.0.0.0/0 to port 80
   - Outbound rule – from 0.0.0.0/0 to private subnet, port 80

   **Web Server (Private) Subnet**
   - Inbound rule – from 0.0.0.0/0 to port 80

4. Ensure that you add the private IP address to be used for the VIP to the VMs NIC using the Azure Portal, otherwise Azure will **not** be aware of this address

2) Configure a Custom Routing Table

1. Using the search option at the top of the page, search for “Route tables”
2. Click **Add**

   ![Configuration Example](https://example.com)

   - **Subscription**: Loadbalancer.org Pay-As-You-Go
   - **Resource group**: QS-RG1
   - **Region**: UK South
   - **Name**: RT1
   - **Propagate gateway routes**: Yes

3. Configure the **Subscription**, **Resource group & Region** settings according to your requirements
4. Enter a suitable name for the Route table, e.g. **RT1**
5. Click **Next : Tags**
6. Define any required tags
7. Click **Review + Create**
8. Once created, select the newly created Route table
9. Click **Routes under Settings**, click **Add**
10. Enter a suitable name for the route, e.g. R1
11. Set the Address prefix to 0.0.0.0/0 (i.e. the default route)
12. Set the next hop type to Virtual appliance
13. Set the next hop address to the IP address of the load balancer in the public subnet, e.g. 10.1.6.5
14. Click OK

Note:
As mentioned in the note in the above screen shot, IP forwarding must be enabled for the load balancer VM. This is covered below in section 3).

15. Click Subnets under Settings
16. Click Associate
17. Select the relevant VNet and the Private Subnet

18. Click OK

3) Enable IP Forwarding for the Load balancer VM:

1. In the Azure Portal main menu, select Virtual Machines
2. Select the Load balancer VM and click Networking under Settings
3. Click the Network Interface for the VM
4. Click IP Configurations
5. Ensure that IP forwarding is enabled as shown below
6. Click **Save**

**Step 2 – Configure the Virtual Service on the Load Balancer**

1. Using the WebUI, navigate to: *Cluster Configuration > Layer 4 – Virtual Services* and click **Add a New Virtual Service**
2. Enter the following details:

   ![Virtual Service]

   - Enter an appropriate label for the VIP, e.g. **Web-Cluster1**
   - Set the *Virtual Service IP Address* field to an appropriate value, e.g. 10.1.6.40
   - Set the *Virtual Service Ports* field to 80
   - Leave *Protocol* set to TCP
   - Ensure *Forwarding Method* is set to NAT
   - Click **Update**

**Step 3 – Setting up the Real Servers**

1. Using the WebUI, navigate to: *Cluster Configuration > Layer 4 – Real Servers* and click **Add a new Real Server** next to the newly created VIP
2. Enter the following details:
3. Enter an appropriate label for the RIP, e.g. **Web1**
4. Change the **Real Server IP Address** field to the required IP address, e.g. **10.1.8.100**
5. Set the **Real Server Port** field to **80**
6. Click **Update**
7. Repeat the above steps to add your other Web Server(s)

**Note:**
If you want your Real Servers to be able to access the outside world, i.e. the Internet in a public facing deployment, outbound requests passing via the load balancer must be NAT’d so that the source IP becomes the load balancer’s own external address. This can be configured using the WebUI menu option: **Cluster Configuration > Layer 4 Advanced Configuration** and setting the **Auto-NAT** drop-down to **eth0**.

### Step 4 – Assigning a Public IP Address

- For public facing deployments, you’ll need to associate a Public IP address with the Private IP address used for the VIP. To associate a Public IP address with a Private address:

  1. Select the load balancer VM in the Azure Portal
  2. Click **Networking**
  3. Select the Network Interface
  4. Select **IP Configurations**
  5. Click the IP configuration for the VIP
  6. Change Public IP address to **Enabled**
  7. Select an existing available Public IP address or create a new one
  8. Click **Save**

### Step 5 – Testing & Verification

- To test the configuration is working, browse to the public IP address or FQDN on HTTP port 80, i.e.

  - `http://<Public IP Address>`
  - `http://<FQDN>`
5 – LOAD BALANCING WEB SERVERS, SINGLE APPLIANCE, 2 SUBNETS, LAYER 7 WITH TPROXY

This example is the same as the previous example with regard to network layout. It uses 2 subnets – one public subnet for the load balancer and one private subnet for the web servers. The load balancer has a single network interface located in the first subnet. Routing rules for the second private subnet must be changed so that return traffic passes back via the load balancer. This is achieved by creating a custom routing table with the required rules, then associating this with the private subnet.

In addition, Layer 7 transparency is enabled on the load balancer to ensure that the source IP address of packets reaching the web servers is the source IP of the clients and not the IP address of the load balancer.

Note:
This configuration is currently not supported in HA mode. In this mode, the custom routing rules would need to be dynamically modified to route via the Slave appliance rather than the Master if a failover occurs. This is currently not supported.

Step 1 – Setting up Azure
Follow steps 1 to 3 in the “Setting up Azure” section of example 3 on page 47.

Step 2 – Configure the Virtual Service on the Load Balancer

1. Using the WebUI, navigate to: Cluster Configuration > Layer 7 – Virtual Services and click Add a New Virtual Service
2. Enter the following details:
   - Enter an appropriate label for the VIP, e.g. Web-Cluster1
   - Set the Virtual Service IP Address field to an appropriate value, e.g. 10.1.6.40
   - Set the Virtual Service Ports field to 80
   - Leave Protocol set to HTTP Mode
   - Click Update
3. Now click Modify next to the newly created VIP
4. Scroll down to the ‘Other’ section, click [Advanced], scroll to the bottom and enable (check) Transparent Proxy
5. Click Update
Step 3 – Setting up the Real Servers

1. Using the WebUI, navigate to: *Cluster Configuration > Layer 7 – Real Servers* and click *Add a new Real Server* next to the newly created VIP
2. Enter the following details:

<table>
<thead>
<tr>
<th>Label</th>
<th>Web1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real Server IP Address</td>
<td>10.1.6.100</td>
</tr>
<tr>
<td>Real Server Port</td>
<td>80</td>
</tr>
<tr>
<td>Re-Encrypt to Backend</td>
<td></td>
</tr>
<tr>
<td>Weight</td>
<td>100</td>
</tr>
</tbody>
</table>

3. Enter an appropriate label for the RIP, e.g. **Web1**
4. Change the *Real Server IP Address* field to the required IP address, e.g. **10.1.8.100**
5. Set the *Real Server Port* field to **80**
6. Click **Update**
7. Repeat the above steps to add your other Web Server(s)

**Note:**
If you want your Real Servers to be able to access the outside world, i.e. the Internet in a public facing deployment, outbound requests passing via the load balancer must be NAT’d so that the source IP becomes the load balancer’s own external address. This can be configured using the WebUI menu option: *Cluster Configuration > Layer 4 Advanced Configuration* and setting the *Auto-NAT* drop-down to **eth0**.

Step 4 – Assigning a Public IP Address

- For public facing deployments, you’ll need to associate a Public IP address with the Private IP address used for the VIP. To associate a Public IP address with a Private address:

   1. Select the load balancer VM in the Azure Portal
   2. Click **Networking**
   3. Select the Network Interface
   4. Select **IP Configurations**
   5. Click the IP configuration for the VIP
   6. Change Public IP address to **Enabled**
   7. Select an existing available Public IP address or create a new one
   8. Click **Save**

Step 5 – Testing & Verification

- To test the configuration is working, browse to the public IP address or FQDN on HTTP port 80, i.e.


9. Testing – General Comments

Note:
For more information and guidance, please refer to the Administration Manual and search for “Testing Load Balanced Services”.

TESTING LOAD BALANCED SERVICES

For example, to test a web server based configuration, add a page to each web servers root directory, e.g. test.html and put the server name on this page for easy identification during the tests.

Use two or more clients to do the testing. Open up a web browser on each test clients and enter the URL for the VIP, e.g. http://104.40.133.119

Provided that persistence is disabled, each client should see a different server name because of the load balancing algorithm in use, i.e. they are being load balanced across the cluster.

Why test using two clients? If you use a single client it will most likely keep on hitting the same server for multiple requests. This is to do with the way that the load balancing algorithms are optimized.

DIAGNOSING VIP CONNECTION PROBLEMS

1. Make sure that the device is active – this can be checked in the WebUI. For a single appliance, the status bar should report Master & Active as shown below:

   Master | Slave | Active | Passive | Link

2. Check that the Real Servers are up – Using System Overview make sure that none of your VIPs are colored red. If they are, the entire cluster is down (i.e. all Real Servers). Green indicates a healthy cluster, yellow indicates that your cluster may need attention (one or more of the Real Servers may be down), and blue indicates all Real Server have been deliberately taken offline (by using either Halt or Drain).

   ![System Overview](image)

   3. Check the connection state
For layer 4 (NAT mode) VIPs, check Reports > Layer 4 Current Connections to view the current traffic in detail. Any packets with state SYN_RECV often implies a return traffic routing issue, so make sure that the routing rules for the real server subnet have been configured correctly.

For Layer 7 VIPs, check Reports > Layer 7 Status. The default credentials required are:

- **username**: loadbalancer
- **password**: <password>

The password is set to a random arbitrary value when first booted. To view the password, using the WebUI navigate to: View Configuration > Layer 7, scroll to the “listen stats” section at the bottom of the screen and the password can be found after the colon on the “stats auth” line, e.g.

```
stats auth loadbalancer:EBAIA2h2i2NJ
```

This will open a second tab in the browser and display a statistics/status report as shown in the example below (this is accessed on port TCP/7777 so make sure that the inbound rules allow connections on this port):

---

**Statistics Report for pid 3261**

---

TAKING REAL SERVERS OFFLINE

1) Using the System Overview check that when you Halt one of the Real Servers the connections are redirected to the other server in the cluster.

2) Stop the web service/process on one of the servers, wait a few seconds (for the load balancer to detect the change) and then refresh the browsers on both clients. They should now both switch to the same server (since one has been removed from the load balancing list). Also check that the server is shown red (down) in the system overview.

3) Start the web service/process on the server, wait a few seconds and then refresh the browsers again. After a few refreshes they should again show different web servers. Also check that the server is shown green (up) in the system overview.

The System Overview shows the status as these tests are performed:

---
In this example:

**RIP1** is green, this indicates that it’s operating normally
**RIP2** is blue, this indicates that it has been either Halted or Drained. in this example Halt has been used as indicated by *Online (Halt)* being displayed. If it had been drained it would show as *Online (Drain)*
**RIP3** is red, this indicates that it has failed a health check

**USING REPORTS & LOG FILES**
The appliance includes several logs and reports that are very useful when diagnosing issues. Both are available as main menu options in the WebUI. Details of both can be found in the administration manual.

### 10. More Information
Please refer to our website for all the latest [manuals](#) and [deployment guides](#).

### 11. Loadbalancer.org Technical Support
If you have any questions regarding the appliance or how to load balance your application, please don’t hesitate to contact our support team using the following email address: support@loadbalancer.org.
## 12. Company Contact Information

<table>
<thead>
<tr>
<th>Country</th>
<th>Address</th>
<th>Website URL</th>
<th>Contact Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Website</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>North America (US)</strong></td>
<td>Loadbalancer.org, Inc. 4550 Linden Hill Road, Suite 201 Wilmington, DE 19808 USA</td>
<td><a href="http://www.loadbalancer.org">www.loadbalancer.org</a></td>
<td>+1 833.274.2566 <a href="mailto:sales@loadbalancer.org">sales@loadbalancer.org</a> <a href="mailto:support@loadbalancer.org">support@loadbalancer.org</a></td>
</tr>
<tr>
<td><strong>North America (Canada)</strong></td>
<td>Loadbalancer.org Appliances Ltd. 300-422 Richards Street Vancouver, BC V6B 2Z4 Canada</td>
<td></td>
<td>+1 866.998.0508 <a href="mailto:sales@loadbalancer.org">sales@loadbalancer.org</a> <a href="mailto:support@loadbalancer.org">support@loadbalancer.org</a></td>
</tr>
<tr>
<td><strong>Europe (UK)</strong></td>
<td>Loadbalancer.org Ltd. Compass House North Harbour Business Park Portsmouth, PO6 4PS UK</td>
<td></td>
<td>+44 (0)330 380 1064 <a href="mailto:sales@loadbalancer.org">sales@loadbalancer.org</a> <a href="mailto:support@loadbalancer.org">support@loadbalancer.org</a></td>
</tr>
<tr>
<td><strong>Europe (Germany)</strong></td>
<td>Loadbalancer.org GmbH Tengstraße 27 80798 München Germany</td>
<td></td>
<td>+49 (0)89 2000 2179 <a href="mailto:vertrieb@loadbalancer.org">vertrieb@loadbalancer.org</a> <a href="mailto:support@loadbalancer.org">support@loadbalancer.org</a></td>
</tr>
</tbody>
</table>