LOADBALANCER

DEPLOYMENT GUIDE

Load Balancing Microsoft IIS

Version 1.9.0

Table of Contents

1. About this Guide

This guide details the steps required to configure a load balanced Microsoft IIS environment utilizing Loadbalancer.org appliances. It covers the configuration of the load balancers and also any Microsoft IIS 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 IIS. 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. Software Versions Supported

3.1. Loadbalancer.org Appliance

- V8.6 and later

> **Note** The screenshots used throughout this document aim to track the latest Loadbalancer.org software version. If using an older software version, note that the screenshots presented here may not match the WebUI exactly.

3.2. Microsoft IIS

- All versions

4. Microsoft Internet Information Services (IIS)

IIS is one of the components of Microsoft Windows and is Microsoft's implementation of a web server. The protocols supported include HTTP, HTTPS, FTP, FTPS, SMTP & NNTP. The latest versions of IIS are built on an open and modular architecture that allows users to customize and add new features through various IIS Extensions. It's estimated that around 25% of all websites utilize IIS.

5. Load Balancing IIS

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

5.1. The Basics

The primary function of the load balancer is to distribute inbound requests across multiple IIS servers. This allows

administrators to configure multiple servers and easily share the load between them. Adding additional capacity as demand grows then becomes straight forward and can be achieved by simply adding additional IIS servers to the load balanced cluster.

5.2. Ports & Protocols

The following table shows the ports that are normally used with IIS for web based applications:

Port	Protocol	Use
80	TCP/HTTP	HTTP web traffic
443	TCP/HTTPS	HTTPS web traffic

5.3. IIS Server Health-checks

Regular IIS server monitoring ensures that failed servers are marked as down and client requests are only directed to functional servers. Health checks can range from a simple ICMP PING to a full negotiate check where content on a certain page is read and verified. Please refer to Real Server (IIS) Health Checks for more details.

5.4. SSL Termination & Certificates

SSL can be terminated on the IIS servers (**SSL pass-through**) or on the load balancer (**SSL offloading**). When terminated on the load balancer, it's also possible to enable re-encryption so that the connection from the load balancer to the IIS servers is also protected (**SSL bridging**). Please refer to SSL Termination for more details of each option.

> **Note** SSL termination on the load balancer can be very CPU intensive. In most cases, for a scalable solution, terminating SSL on the IIS servers is usually the best option.

5.5. Persistence (aka Server Affinity)

Ideally, persistence should be considered at the start of any IIS project. A database is typically used to maintain session information. This information is then available to all IIS servers so that whenever a user connects, any previous session details can be accessed. If this structure is not in place, persistence can be implemented on the load balancer. This ensures that requests from a particular user will be handled by the same IIS server during their session. For web based applications, persistence can be based on:

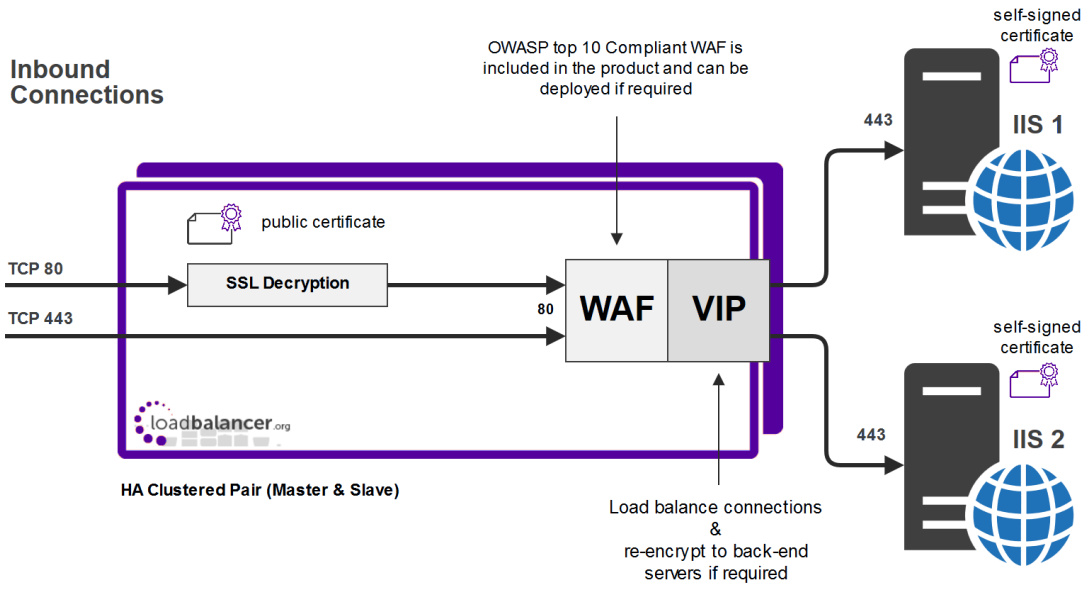
1. Source IP address
2. HTTP Cookie (inserted by the load balancer)
3. Application Cookie (inserted by the application)
4. SSL Session ID
5. HTTP Cookie / failing back to Source IP address if the cookie is missing
6. X-Forwarded-For / failing back to Source IP address if the header is missing

> **Note** For persistence options 2 to 6, a layer 7 SNAT mode VIP is required – please refer to Layer 7 SNAT Mode and Appliance & IIS Server Configuration – Using Layer 7 SNAT Mode for more

details. For HTTPS traffic, when SSL is terminated on the IIS Servers, only source IP address persistence can be used. To use the other persistence methods, SSL must be terminated on the load balancer so that the traffic is readable – please refer to SSL Termination for more details on SSL termination.

5.6. Load Balancer Deployment

The following diagram illustrates how the load balancer is deployed with multiple IIS servers.



WAF =**W**eb **A**pplication **F**irewall

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

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

WAF

As illustrated in the diagram above, a WAF is included with the appliance at no extra cost and can be deployed if required. Please refer to Web Application Firewall (WAF) for more details.

SSL Decryption / Re-Encryption

As illustrated in the diagram above and as mentioned in SSL Termination & Certificates, the load balancer can be configured to terminate SSL and also re-encrypt to the backend servers if required. Please refer to the SSL Termination for more details.

5.7. Load Balancer Deployment Modes

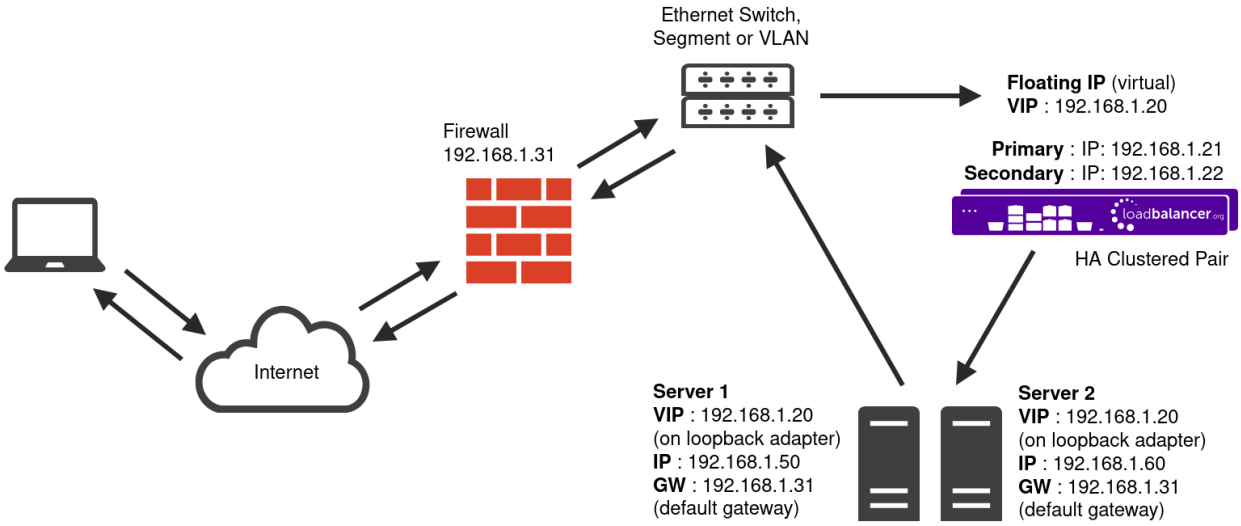
The load balancer can be deployed in 4 fundamental ways: ***Layer 4 DR mode***, ***Layer 4 NAT mode***, ***Layer 4 SNAT mode*** and ***Layer 7 SNAT mode***. For IIS, Layer 4 DR mode, Layer 4 NAT mode or Layer 7 SNAT are recommended. These modes are described below and are used for the configurations presented in this guide. For configuring using DR mode, please refer to Appliance & IIS Server Configuration – Using Layer 4 DR Mode, for configuring

using NAT mode, refer to Appliance & IIS Server Configuration – Using Layer 4 NAT Mode, and for layer 7 SNAT mode, refer to Appliance & IIS Server Configuration – Using Layer 7 SNAT Mode.

Layer 4 DR Mode

One-arm direct routing (DR) mode is a very high performance solution that requires little change to your existing infrastructure.

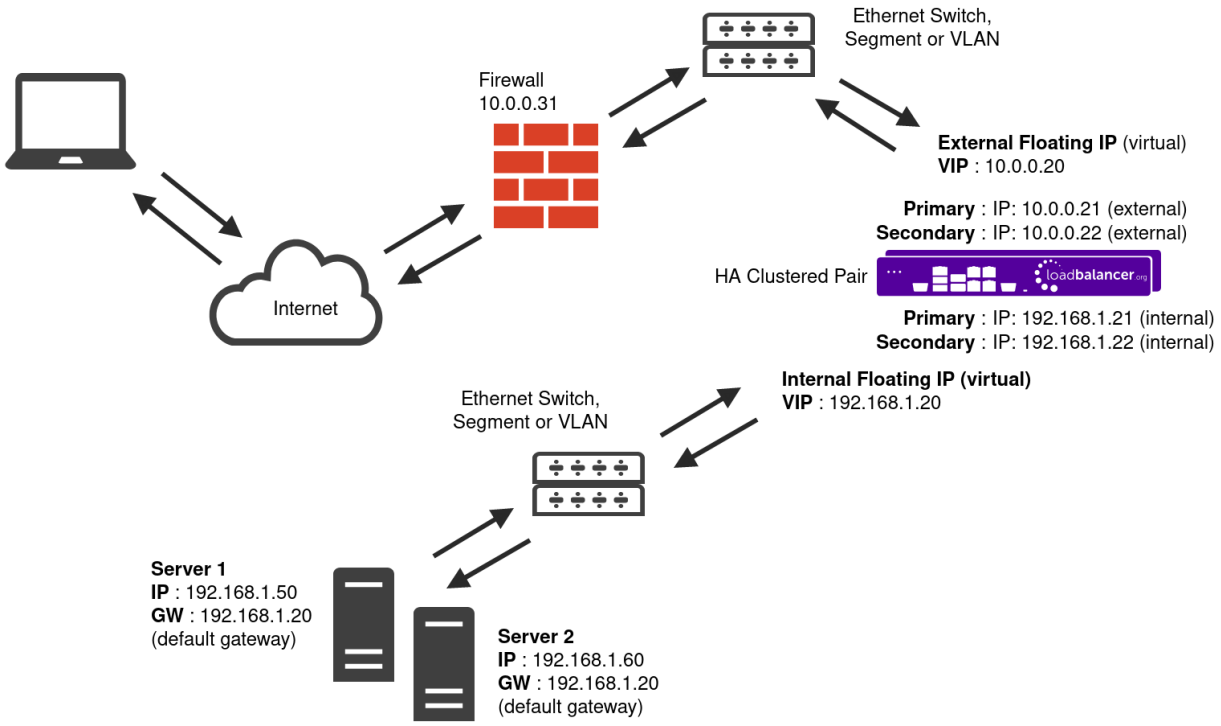
> **Note** Kemp, Brocade, Barracuda & A10 Networks call this *Direct Server Return* and F5 call it *nPath*.



- DR mode works by changing the destination MAC address of the incoming packet to match the selected Real Server on the fly which is very fast.
- When the packet reaches the Real Server it expects the Real Server to own the Virtual Services IP address (VIP). This means that you need to ensure that the Real Server (and the load balanced application) respond to both the Real Server's own IP address and the VIP.
- The Real Servers should not respond to ARP requests for the VIP. Only the load balancer should do this. Configuring the Real Servers in this way is referred to as ***Solving the ARP problem***. For more information please refer to DR Mode Considerations.
- On average, DR mode is 8 times quicker than NAT for HTTP, 50 times quicker for Terminal Services and much, much faster for streaming media or FTP.
- The load balancer must have an Interface in the same subnet as the Real Servers to ensure layer 2 connectivity required for DR mode to work.
- The VIP can be brought up on the same subnet as the Real Servers, or on a different subnet provided that the load balancer has an interface in that subnet.
- Port translation is not possible with DR mode, e.g. VIP:80 → RIP:8080 is not supported.
- DR mode is transparent, i.e. the Real Server will see the source IP address of the client.

Layer 4 NAT Mode

Layer 4 NAT mode is a high performance solution, although not as fast as layer 4 DR mode. This is because real server responses must flow back to the client via the load balancer rather than directly as with DR mode.



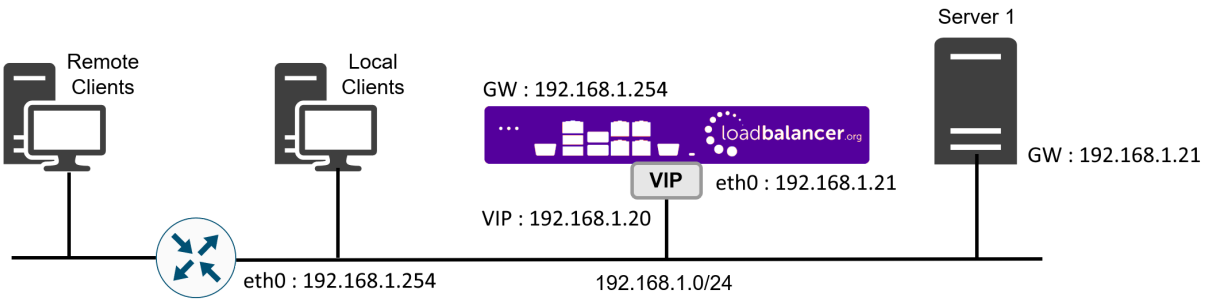
- The load balancer translates all requests from the Virtual Service to the Real Servers.
- NAT mode can be deployed in the following ways:
  - **Two-arm (using 2 Interfaces)** (as shown above) - Here, 2 subnets are used. The VIP is located in one subnet and the load balanced Real Servers are located in the other. The load balancer requires 2 interfaces, one in each subnet.

> **Note** This can be achieved by using two network adapters, or by creating VLANs on a single adapter.

    - Normally **eth0** is used for the internal network and **eth1** is used for the external network although this is optional. Any interface can be used for any purpose.
    - If the Real Servers require Internet access, ***Autonat*** should be enabled using the WebUI menu option: ***Cluster Configuration > Layer 4 - Advanced Configuration***, the external interface should be selected.
    - The default gateway on the Real Servers must be set to be an IP address on the load balancer.

> **Note** For an HA clustered pair, a floating IP should be added to the load balancer and used as the Real Server's default gateway. This ensures that the IP address can 'float' (move) between Primary and Secondary appliances.

    - Clients can be located in the same subnet as the VIP or any remote subnet provided they can route to the VIP.
  - **One-arm (using 1 Interface)** - Here, the VIP is brought up in the same subnet as the Real Servers.



- To support remote clients, the default gateway on the Real Servers must be an IP address on the load balancer and routing on the load balancer must be configured so that return traffic is routed back via the router.

> **Note** For an HA clustered pair, a floating IP should be added to the load balancer and used as the Real Server's default gateway. This ensures that the IP address can 'float' (move) between Primary and Secondary appliances.

- To support local clients, return traffic would normally be sent directly to the client bypassing the load balancer which would break NAT mode. To address this, the routing table on the Real Servers must be modified to force return traffic to go via the load balancer. For more information please refer to One-Arm (Single Subnet) NAT Mode.

- If you want Real Servers to be accessible on their own IP address for non-load balanced services, e.g. RDP, you will need to setup individual SNAT and DNAT firewall script rules for each Real Server or add additional VIPs for this.
- Port translation is possible with Layer 4 NAT mode, e.g. VIP:80 → RIP:8080 is supported.
- NAT mode is transparent, i.e. the Real Servers will see the source IP address of the client.

NAT Mode Packet re-Writing

In NAT mode, the inbound destination IP address is changed by the load balancer from the Virtual Service IP address (VIP) to the Real Server. For outbound replies the load balancer changes the source IP address of the Real Server to be the Virtual Services IP address.

The following table shows an example NAT mode setup:

Protocol	VIP	Port	RIP	Port
TCP	10.0.0.20	80	192.168.1.50	80

In this simple example all traffic destined for IP address 10.0.0.20 on port 80 is load-balanced to the real IP address 192.168.1.50 on port 80.

Packet rewriting works as follows:

1) The incoming packet for the web server has source and destination addresses as:

Source	x.x.x.x:34567	**Destination**	10.0.0.20:80

2) The packet is rewritten and forwarded to the backend server as:

Source	x.x.x.x:34567	**Destination**	192.168.1.50:80

3) Replies return to the load balancer as:

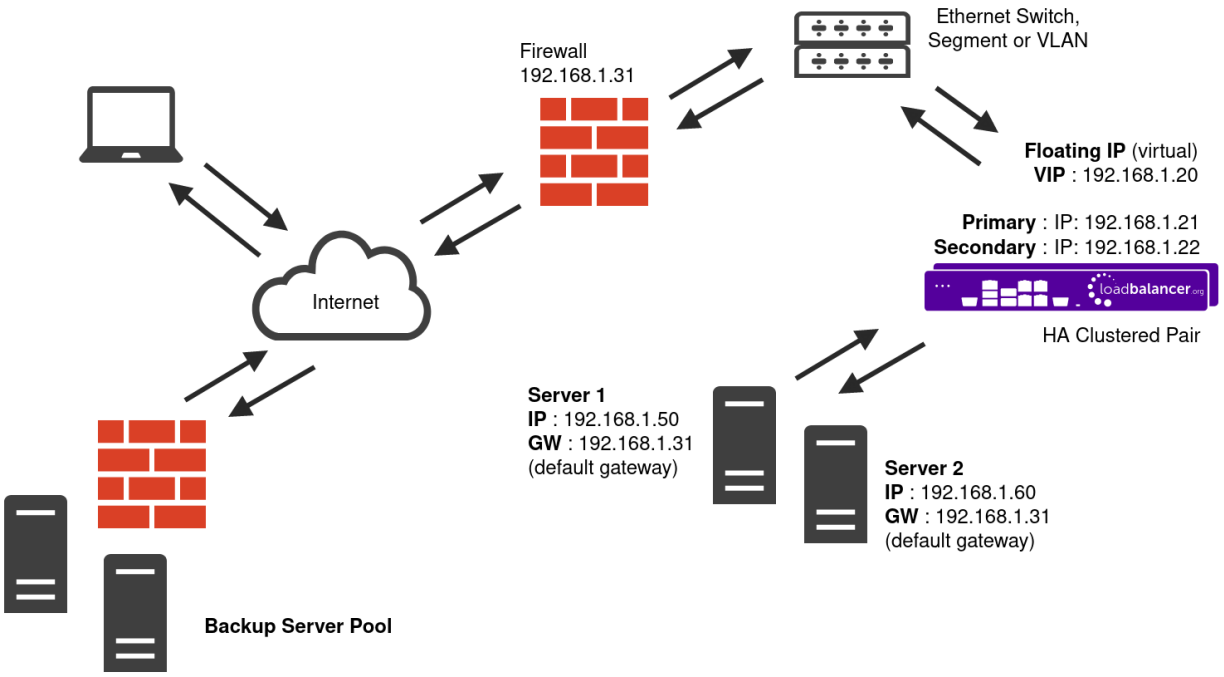
Source	192.168.1.50:80	**Destination**	x.x.x.x:34567

4) The packet is written back to the VIP address and returned to the client as:

Source	10.0.0.20:80	**Destination**	x.x.x.x:34567

Layer 7 SNAT Mode

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



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

packets and replacing the IP address of the load balancer with the IP address of the client. For more information on these methods please refer to Transparency at Layer 7.

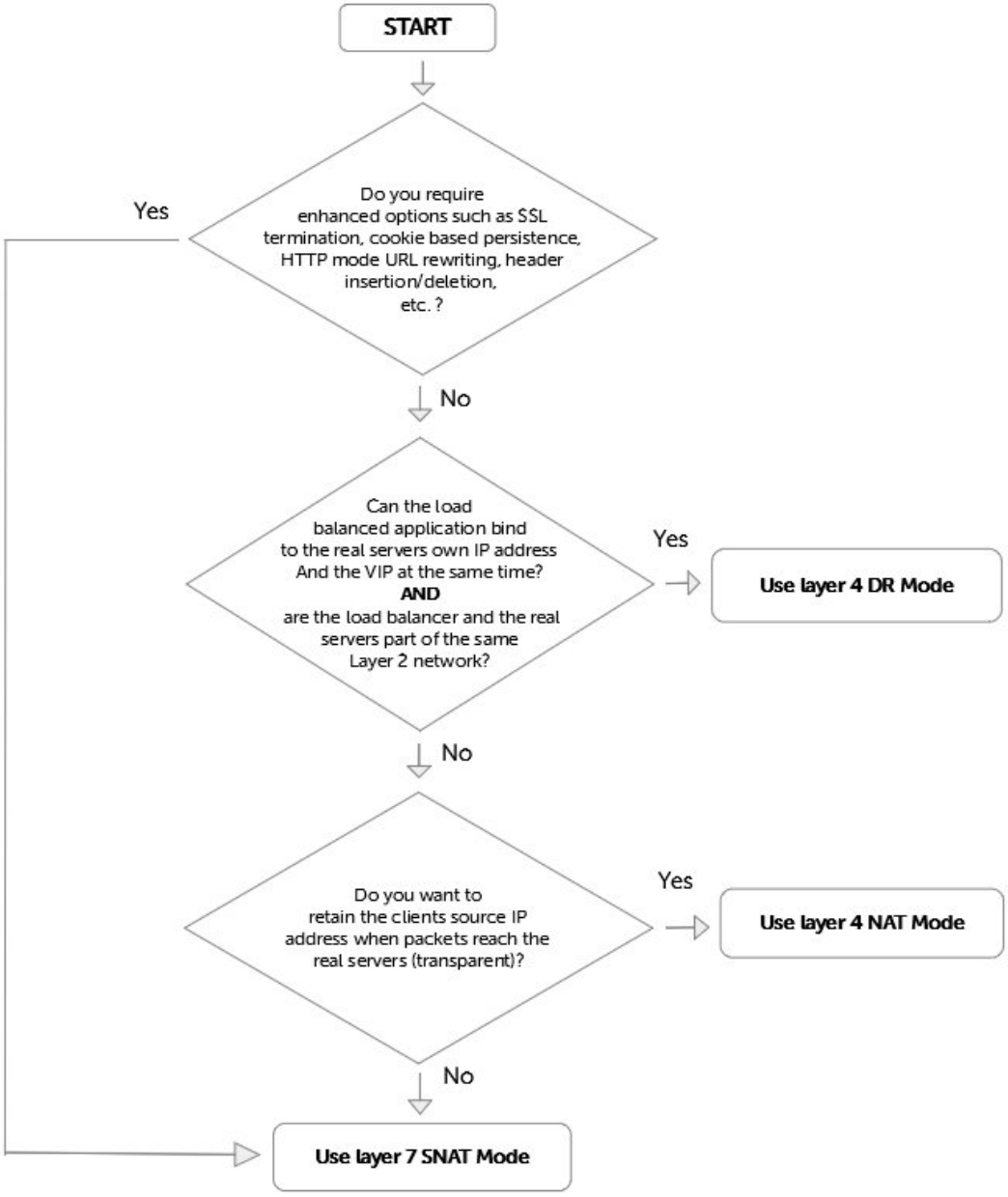
- Layer 7 SNAT mode can be deployed using either a one-arm or two-arm configuration. For two-arm deployments, **eth0** is normally used for the internal network and **eth1** is used for the external network although this is not mandatory.
- Requires no mode-specific configuration changes to the load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

Loadbalancer.org Recommended Mode

Where possible we recommend that Layer 4 Direct Routing (DR) mode is used. This mode offers the best possible performance since replies go directly from the IIS servers to the client, not via the load balancer. It's also relatively simple to implement. Ultimately, the final choice does depend on your specific requirements and infrastructure.

Helping you Choose

The flow chart below is intended as a simple guide to help determine which deployment mode is most appropriate. Please also refer to the previous section which describes each deployment mode.



6. Loadbalancer.org Appliance – the Basics

6.1. Virtual Appliance

A fully featured, fully supported 30 day trial is available if you are conducting a PoC (Proof of Concept) deployment. The VA is currently available for VMware, Virtual Box, Hyper-V, KVM, XEN and Nutanix AHV and has been optimized for each Hypervisor. By default, the VA is allocated 2 vCPUs, 4GB of RAM and has a 20GB virtual disk. The Virtual Appliance can be downloaded here.

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

Note	Please refer to Virtual Appliance Installation and the ReadMe.txt text file included in the VA download for additional information on deploying the VA using the various Hypervisors.

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

6.2. Initial Network Configuration

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

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

6.3. Accessing the Appliance WebUI

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

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

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

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

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

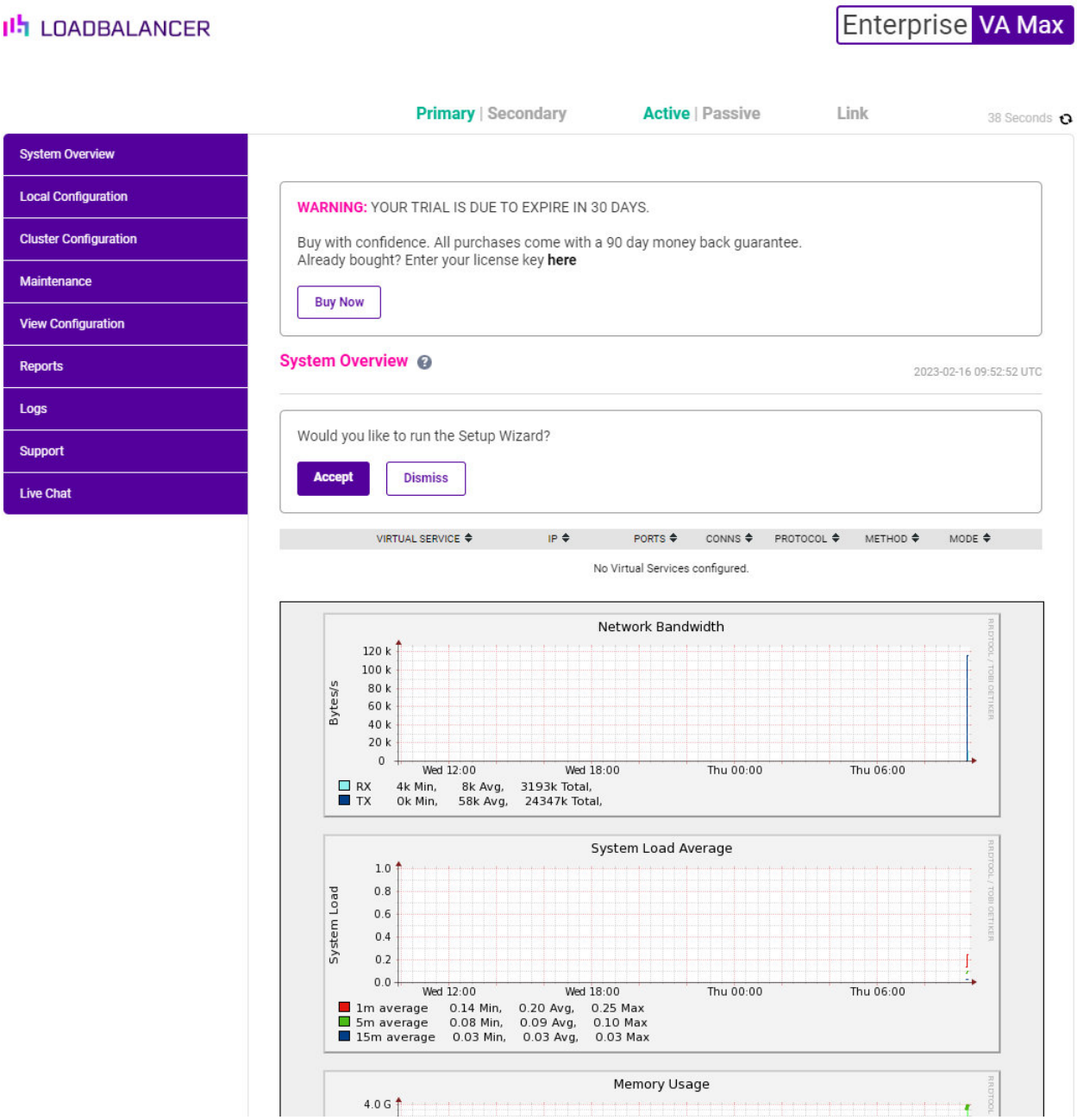
   > **Note** You'll receive a warning about the WebUI's certificate. This is due to the default self signed certificate that is used. If preferred, you can upload your own certificate - for more information, please refer to Appliance Security Features.

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

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

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

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



3. You'll be asked if you want to run the Setup Wizard. Click **Dismiss** if you're following a guide or want to configure the appliance manually. Click **Accept** to start the Setup Wizard.

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

Main Menu Options

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

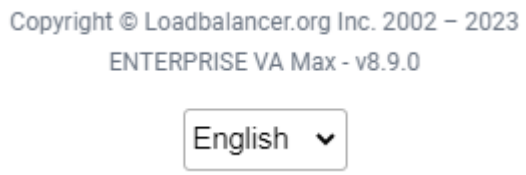
Live Chat - Start a live chat session with one of our Support Engineers

6.4. Appliance Software Update

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

Determining the Current Software Version

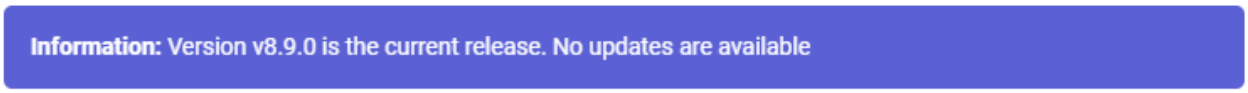
The software version is displayed at the bottom of the WebUI as shown in the example below:



Checking for Updates using Online Update

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

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



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

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

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

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

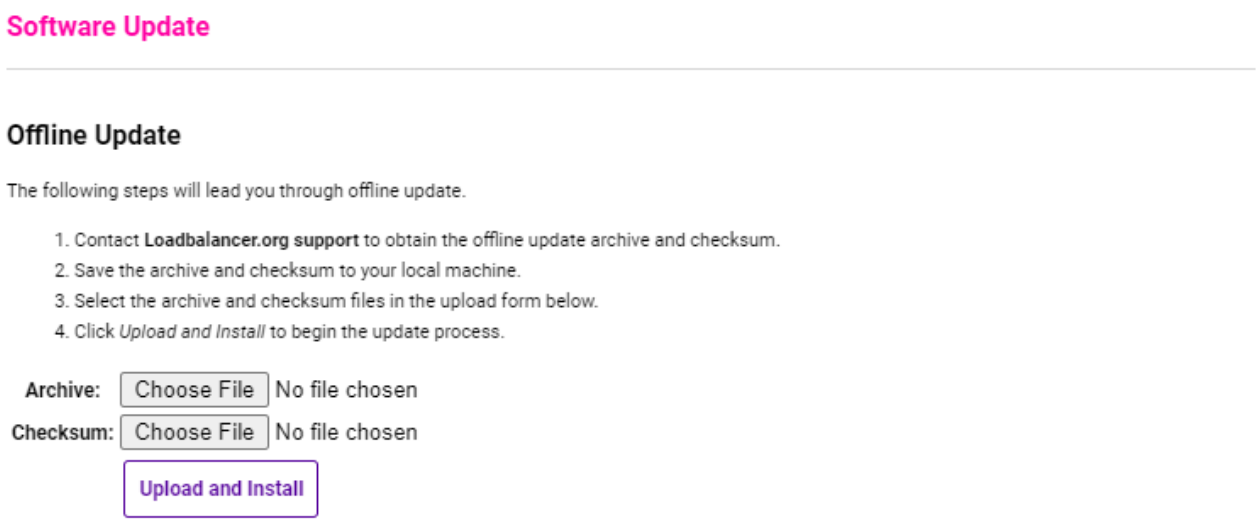
Using Offline Update

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

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

To perform an offline update:

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



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

6.5. Ports Used by the Appliance

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

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

6.6. HA Clustered Pair Configuration

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

7. Appliance & IIS Server Configuration – Using Layer 4 DR Mode

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

7.1. Overview

This is our recommended deployment mode for IIS. It's ideal when you want the fastest possible deployment and don't need layer 7 techniques such as advanced persistence methods, SSL termination, URL rewriting, header insertion/manipulation etc. If you do need to use these features, you should use Layer 7 SNAT mode instead – please refer to Appliance & IIS Server Configuration – Using Layer 7 SNAT Mode for more details.

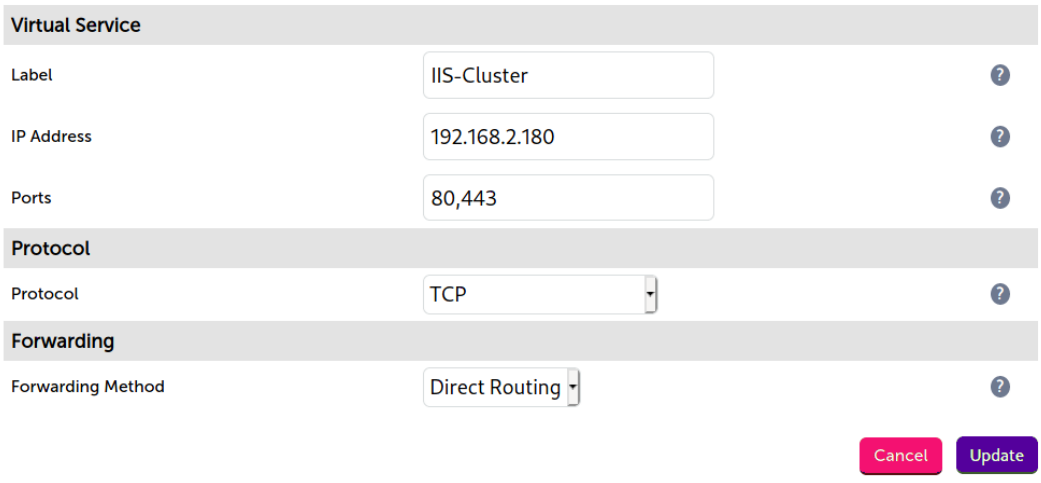
7.2. Load Balancer Configuration

Configure the Network Interface

1. One interface is required - for details on setting up the network, please refer to Initial Network Configuration.

Configure the Virtual Service (VIP)

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



3. Enter an appropriate name (Label) for the VIP, e.g. **IIS-Cluster**.
4. Set the *Virtual Service IP address* field to the required IP address, e.g. **192.168.2.180**.

5. Set the *Virtual Service Ports* field to **80,443**.

> **Note**
>
> Including port 443 here means that SSL is terminated on the IIS servers. HTTP and HTTPS traffic will be forwarded to the same IIS server – provided that persistence is enabled (see step 11 below).
>
> If you want to terminate SSL on the load balancer, you'll have to use one of the other modes (layer 4 NAT mode or Layer 7 SNAT mode) because DR mode cannot be used as explained in SSL Termination on the Load Balancer (SSL Offloading).

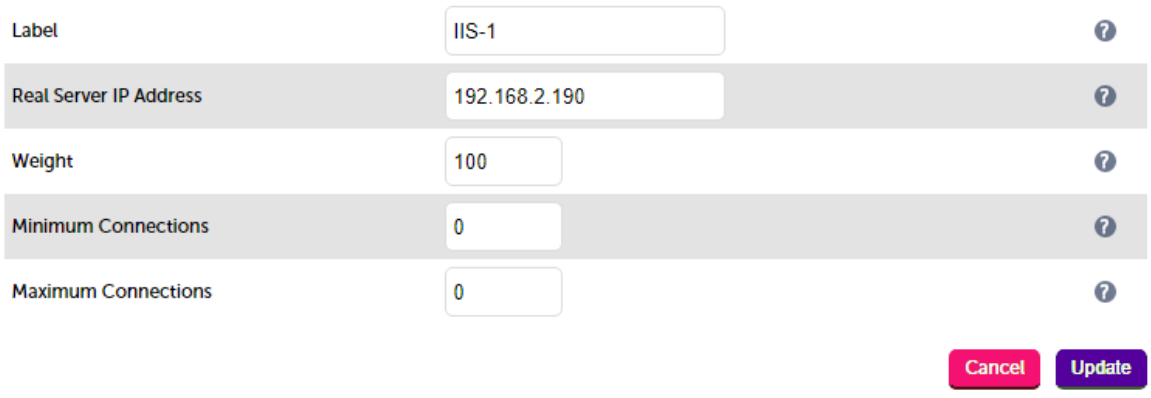
6. Leave *Protocol* set to **TCP**.
7. Ensure that *Forwarding Method* is set to **Direct Routing**.
8. Click **Update**.
9. Now click **Modify** next to the newly created Virtual Service.
10. Set *Balance Mode* (the load balancing algorithm) according to your requirements. Weighted least connection is the default and recommended method.
11. Persistence is enabled by default for new layer 4 VIPs and is based on source IP address. The persistence timeout can be set using the *Persistence Timeout* field, the default is 5 minutes which is normally fine for HTTP/HTTPS traffic.

> **Note** For more information about persistence, please refer to Persistence (aka Server Affinity).

12. Click **Update**.

Configure the Real Servers (RIPs)

1. Using the WebUI, navigate to: *Cluster Configuration > Layer 4 – Real Servers* and click **Add a new Real Server** next to the newly created Virtual Service.
2. Enter the following details:



3. Enter an appropriate name (Label) for the first IIS server, e.g. **IIS-1**.
4. Change the *Real Server IP Address* field to the required IP address, e.g. **192.168.2.190**.

5. Leave other settings at their default values.
6. Click **Update**.
7. Repeat the above steps for your other IIS server(s).

7.3. IIS Server Configuration

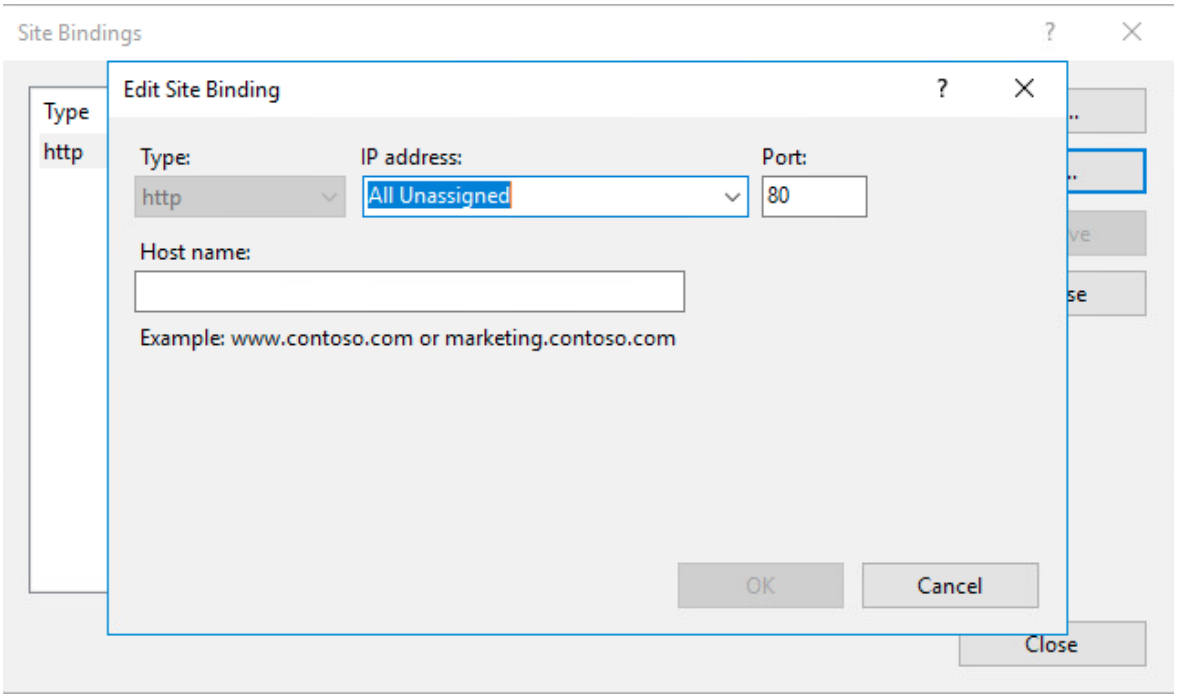
Solve the 'ARP Problem'

As mentioned previously, DR mode works by changing the destination MAC address of the incoming packet to match the selected IIS server on the fly which is very fast. When the packet reaches the IIS server it expects the IIS server to own the Virtual Services IP address (VIP). This means that you need to ensure that the IIS server (and the load balanced application) respond to both the IIS servers own IP address and the VIP. The IIS server should not respond to ARP requests for the VIP. Only the load balancer should do this.

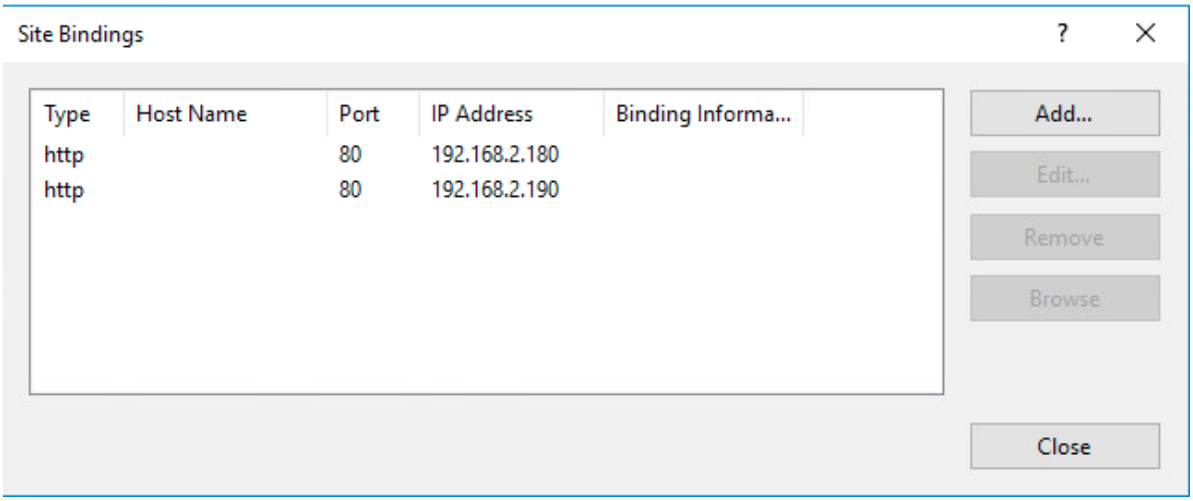
To achieve this, a loopback adapter is added to the IIS servers. The IP address is set to be the same as the Virtual Service and the loopback adapter is configured so that it does not respond to ARP requests. For details on how to solve the ARP Problem for Windows 2012 & later, please refer to Solving the ARP Problem.

Configure IIS Bindings

By default, IIS listens on all configured IP addresses as shown below:



If the default configuration is left, no further IIS configuration is required. If you do change the IP address in the bindings from "All Unassigned" to a specific IP address, then you need to make sure that you also add a binding for the Virtual Service IP address (VIP) as shown below:



In this example, **192.168.2.180** is the main NIC interface for the IIS server and **192.168.2.190** is the Virtual Service's IP address (assigned to the loopback Interface). This ensures that IIS responds to both the RIP and the VIP.

7.4. DR Mode – Key Points

- You *must* solve the 'ARP Problem' on all IIS servers in the cluster (refer to Solving the ARP Problem for more information)
- Virtual Services & Real Servers (i.e. the IIS servers) must be within the same switch fabric. They can be on different subnets but this cannot be across a router – this is due to the way DR mode works, i.e. by changing MAC addresses to match the destination server
- Port translation is not possible, e.g. **VIP:80 > IIS:82** is not allowed. The port used for the VIP & RIP must be the same
- IIS bindings must include the Virtual Service IP (VIP) address – this is the default for IIS when **All Unassigned** is selected

> **Note** For more information about DR mode, please refer to Layer 4 DR Mode.

8. Appliance & IIS Server Configuration – Using Layer 4 NAT Mode

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

8.1. Overview

If you have a custom application that is installed on IIS that is unable to bind to the IIS servers own address *and* the VIP address at the same time, or the load balancer and the IIS servers are not part of the same layer 2 network, then DR mode cannot be used. If you require a high performance solution that is transparent by default (i.e. the client IP address is maintained through the load balancer) and you do not require layer 7 functionality such as advanced persistence methods, URL rewriting, header insertion/manipulation etc. then layer 4 NAT mode

can be used. Layer 4 NAT mode is also a high performance solution, although not as fast as layer 4 DR mode. This is because IIS server responses must flow back to the client via the load balancer rather than directly as with DR mode.

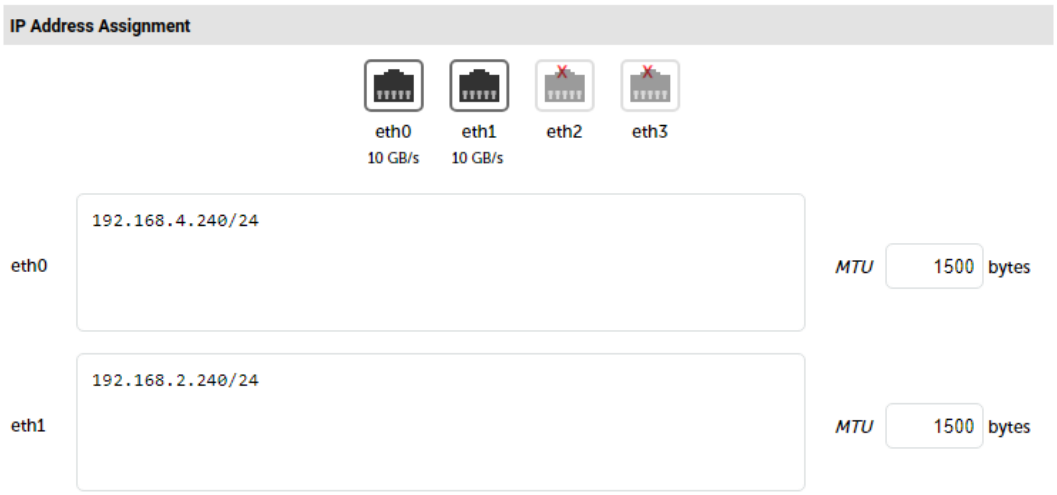
8.2. Load Balancer Configuration

Configure the Network Interfaces

Layer 4 NAT mode is typically used in a 2-arm configuration where the VIP is located in one subnet and the load balanced Real Servers are located in another. This can be achieved by using two network adapters, or by creating VLAN's on a single adapter. Single arm configuration is also supported under certain conditions - for more information please refer to Layer 4 NAT Mode.

To configure an additional network interface for a 2-arm configuration:

1. Using the WebUI, navigate to *Local Configuration > Network Interface Configuration*.
2. Scroll to the *IP Address Assignment* section.

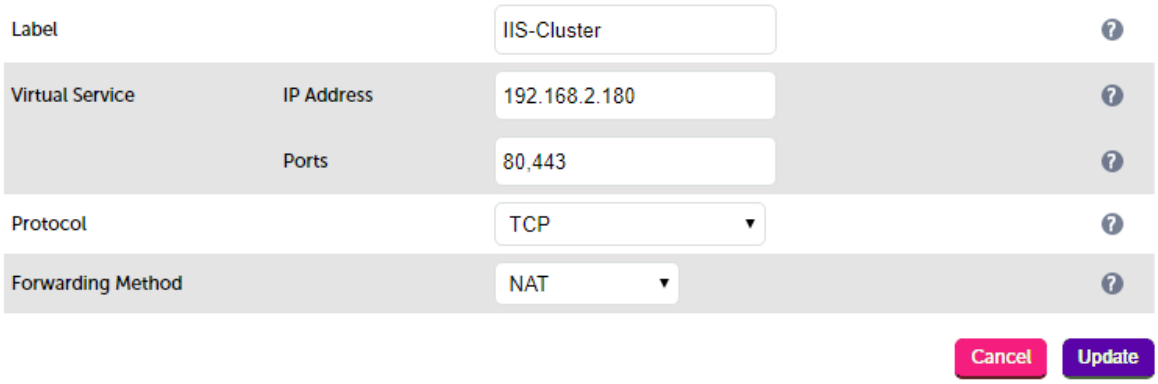


3. Specify an appropriate IP address for **eth1** in CIDR format as shown above.
4. Click **Configure Interfaces**.

> **Note** There are no restrictions on which interface is used for each requirement.

Configure the Virtual Service (VIP)

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



3. Enter an appropriate name (Label) for the VIP, e.g. **IIS-Cluster**.
4. Set the *Virtual Service IP address* field to the required IP address, e.g. **192.168.2.180**.
5. Set the *Virtual Service Ports* field to **80,443**.

> **Note**
>
> Including port 443 here means that SSL is terminated on the IIS servers. HTTP and HTTPS traffic will be forwarded to the same IIS server during a particular client session – provided that persistence is enabled (see step 11 below).
>
> If you want to terminate SSL on the load balancer, you'll need to setup an additional Pound or STunnel (default) SSL VIP to handle the offloading - please refer to SSL Termination for more information.

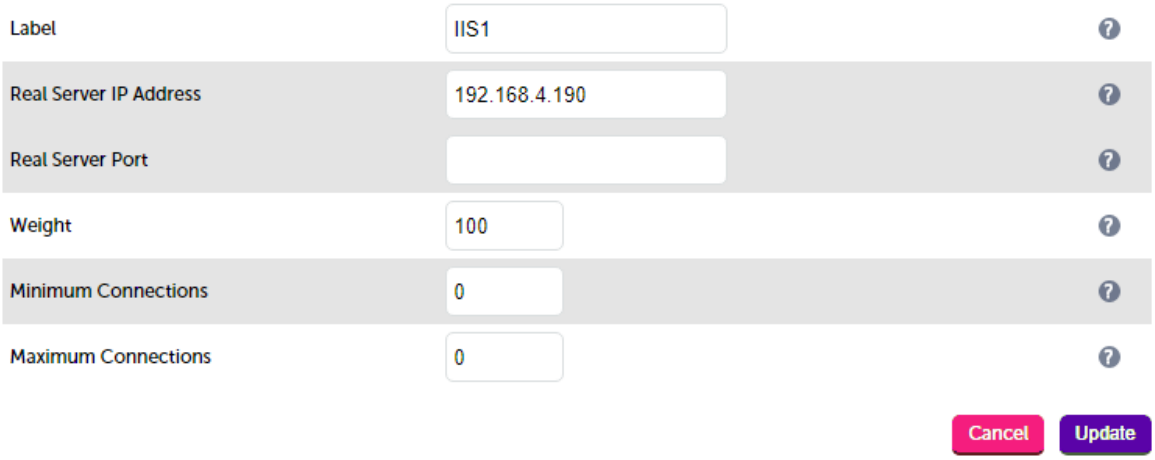
6. Leave *Protocol* set to **TCP**.
7. Set the *Forwarding Method* is to **NAT**.
8. Click **Update**.
9. Now click **Modify** next to the newly created Virtual Service.
10. Set *Balance Mode* (the load balancing algorithm) according to your requirements. Weighted least connection is the default and recommended method.
11. Persistence is enabled by default for new layer 4 VIPs and is based on source IP address. The persistence timeout can be set using the *Persistence Timeout* field, the default is 5 minutes which is normally fine for HTTP/HTTPS traffic.

> **Note**
>
> For more information about persistence, please refer to Persistence (aka Server Affinity).

12. Click **Update**.

Configure the Real Servers (RIPs)

1. Using the WebUI, navigate to: *Cluster Configuration > Layer 4 – Real Servers* and click **Add a new Real Server** next to the newly created Virtual Service.
2. Enter the following details:



3. Enter an appropriate name (Label) for the first IIS server, e.g. **IIS1**.
4. Change the ***Real Server IP Address*** field to the required IP address, e.g. **192.168.4.190**.
5. Leave the ***Real Server Port*** field blank.
6. Leave other settings at their default values.
7. Click **Update**.
8. Repeat the above steps for your other IIS server(s).

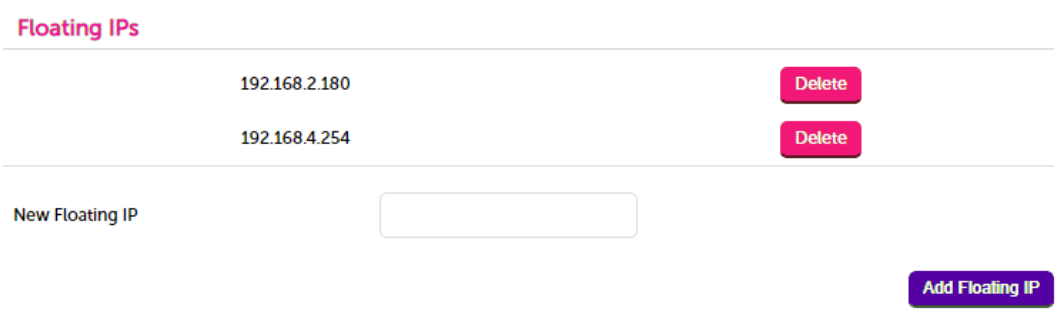
Create a Floating IP to use for the IIS server's Default Gateway

The default gateway on each IIS server must be configured to be an IP address on the load balancer. It's possible to use the IP address assigned to the internal facing interface (**eth0** in this example) for the default gateway, although it's recommended that an additional floating IP is created for this purpose. This is required if two load balancers (our recommended configuration) are used. In this scenario if the Primary unit fails, the floating IP will be brought up on the Secondary.

To create a floating IP address on the load balancer:

1. Using the WebUI, navigate to: ***Cluster Configuration > Floating IPs***.
2. Enter the required IP address to be used for the default gateway, e.g. **192.168.4.254**.
3. Click **Update**.

   Once added, there will be two floating IP's, one for the Virtual Service (**192.168.2.180**) and one for the default gateway (e.g. **192.168.4.254**) as shown below:



8.3. IIS Server Configuration

Default Gateway

To ensure that return traffic passes back to the client via the load balancer, set the default gateway of each IIS server to be the floating IP address added in the previous step, in this example **192.168.4.254**.

NAT Mode – Key Points

- The default gateway on the IIS servers should be an IP address on the load balancer (for an HA pair this must be a floating IP address)
- Port translation is possible, e.g. **VIP:80 > RIP:8080** is allowed

> **Note** For more information about NAT mode, please refer to Layer 4 NAT Mode.

9. Appliance & IIS Server Configuration – Using Layer 7 SNAT Mode

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

9.1. Overview

If you require enhanced options such as SSL termination, cookie based persistence, HTTP mode URL rewriting, header insertion/deletion, etc. then you must use a layer 7 SNAT mode VIP.

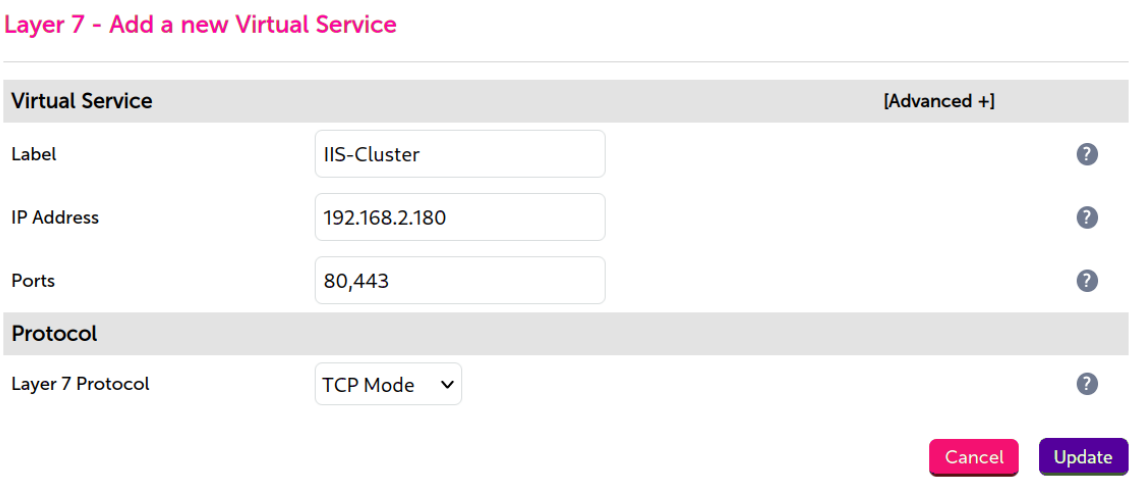
9.2. Load Balancer Configuration

Configure the Network Interface

1. One interface is required - for details on setting up the network, please refer to Initial Network Configuration.

Configure the Virtual Service (VIP)

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



3. Enter an appropriate name (Label) for the Virtual Service, e.g. **IIS-Cluster**.
4. Set the *Virtual Service IP address* field to the required IP address, e.g. **192.168.2.180**.
5. Set the *Virtual Service Ports* field to **80,443**.

> **Note**
>
> Including port 443 here means that SSL is terminated on the IIS servers. HTTP and HTTPS traffic will be forwarded to the same IIS server during a particular client session – provided that persistence is enabled (see step 10 below).
>
> If you want to terminate SSL on the load balancer, you'll need to setup an additional Pound or STunnel (default) SSL VIP to handle the offloading - please refer to SSL Termination for more information.

6. Set *Layer 7 Protocol* to **TCP Mode**.
7. Click **Update**.
8. Now click **Modify** next to the newly created Virtual Service.
9. Set *Balance Mode* (the load balancing algorithm) according to your requirements. Weighted least connection is the default and recommended method.
10. Persistence is enabled by default for new layer 7 VIPs. For TCP Mode (which is required when the VIP handles both HTTP and HTTPS) it's based on source IP address. The persistence timeout can be set using the *Persistence Timeout* field. The default is 30 minutes which is normally fine for HTTP/HTTPS traffic.

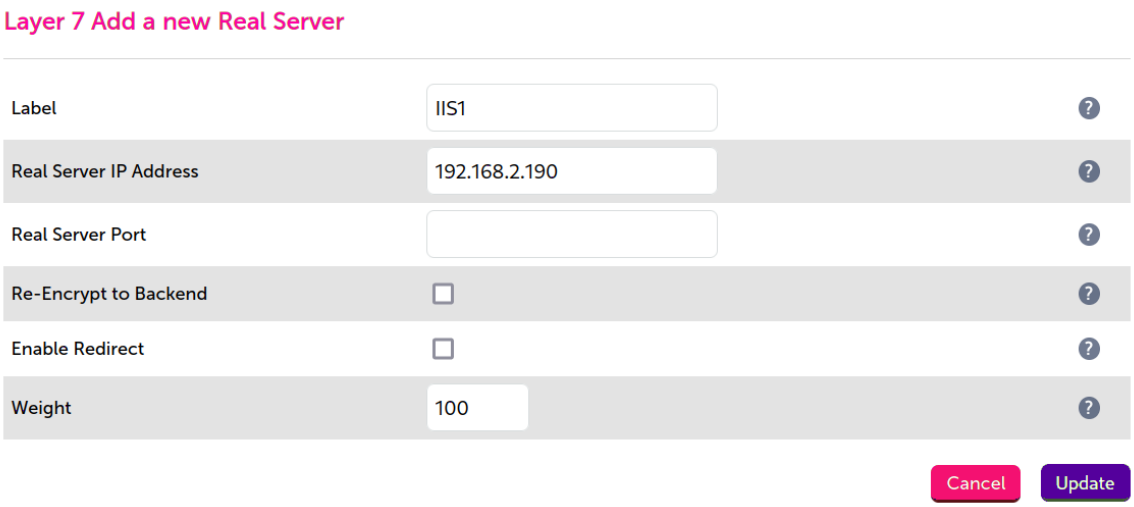
> **Note**
>
> If SSL is terminated on the IIS servers (as in this example) only Source IP address persistence can be used. Other methods such as HTTP Cookie persistence require the traffic to be unencrypted and therefore require SSL to be terminated on the load balancer - please refer to SSL Termination for more information.

> **Note**
>
> For more information about persistence, please refer to Persistence (aka Server Affinity).

11. Click **Update**.

Configure the Real Servers (RIPs)

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



3. Enter an appropriate name (Label) for the first IIS server, e.g. **IIS1**.
4. Change the *Real Server IP Address* field to the required IP address (e.g. **192.168.2.190)**.
5. Leave the *Real Server Port* field blank.
6. Click **Update**.
7. Repeat the above steps for your other IIS server(s).

Finalizing the Configuration

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

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

9.3. IIS Server Configuration

In layer 7 SNAT mode, no IIS server configuration changes are required.

9.4. SNAT Mode – Key Points

- Virtual Services & Real Servers (the IIS servers) can be on the same or different subnets
- Port translation is possible, e.g. **VIP:80 > RIP:8080** is allowed
- No configuration changes are required to the IIS servers
- Enables enhanced options such as SSL termination / re-encryption, cookie based persistence, HTTP mode URL rewriting, header insertion/deletion, etc.

- Not as fast as Layer 4 DR mode or NAT mode

> **Note** For more information about SNAT mode, please refer to Layer 7 SNAT Mode.

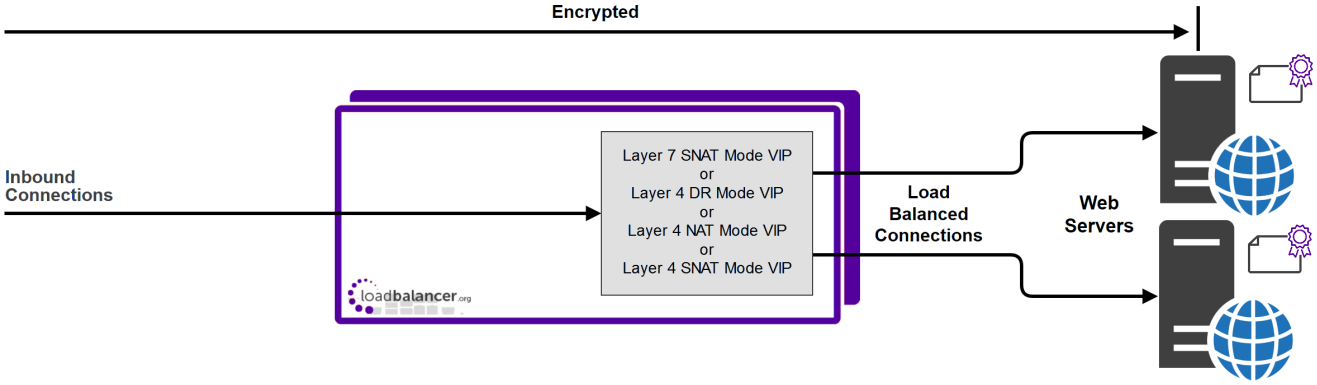
10. Additional Configuration Options & Settings

10.1. SSL Termination

SSL termination can be handled in the following ways:

1. On the IIS Servers (recommended) – aka **SSL Pass-through**.
2. On the load balancer – aka **SSL Offloading**.
3. On the load balancer with re-encryption to the IIS Servers – aka **SSL Bridging**.

SSL Termination on the IIS servers (SSL Pass-through)

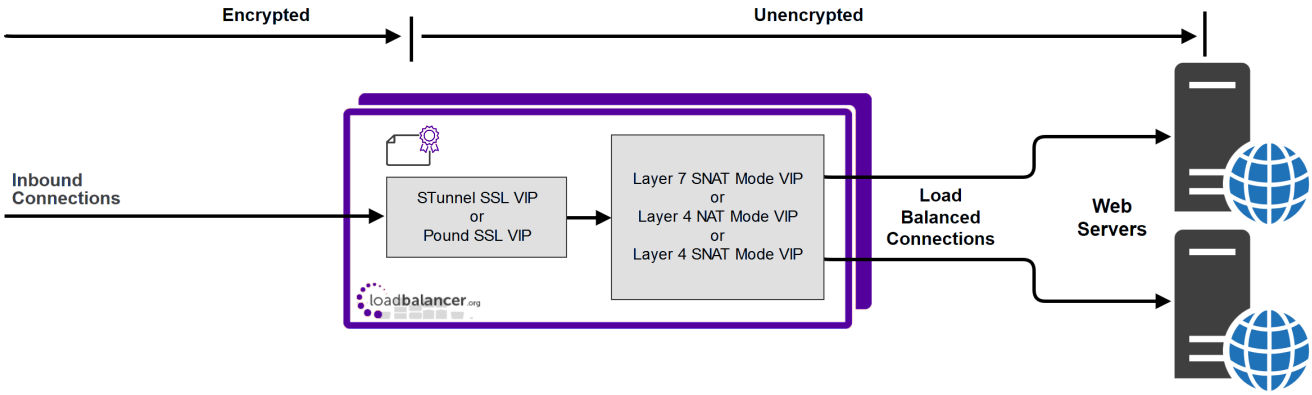


In this case, SSL certificates are installed on each IIS Server in the normal way. Data is encrypted from client to server. This provides full end-to-end data encryption as shown in the diagram above.

> **Note**
>
> The VIP on the load balancer is configured to listen on port 80 & 443.
>
> This is our recommended solution. SSL termination on the load balancer (SSL Offload) can be very CPU intensive and In most cases, for a scalable solution, terminating SSL on the IIS servers is the best option.
>
> It's not possible to use HTTP cookie persistence as well as other layer 7 techniques that control how traffic is sent to the IIS servers because all data is encrypted as it passes through the load balancer.

SSL Termination on the Load Balancer (SSL Offloading)

> **Note** SSL termination on the load balancer can be very CPU intensive. In most cases, for a scalable solution, terminating SSL on the IIS servers is the best option.



In this case, an SSL VIP utilizing either STunnel (default & recommended) or Pound 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. If you require SSL bridging where the data is re-encrypted to the backend IIS servers, please refer to SSL Termination on the Load Balancer with Re-encryption (SSL Bridging).

Note

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 Default Self Signed 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 Pound & STunnel act as a proxy, and the IIS servers see requests with a source IP address of the VIP. However, since the IIS servers believe that they own the VIP (due to the loopback adapter configured to handle to ARP problem) they are unable to reply to Pound/STunnel.

If a layer 7 VIP is used as the backend for the SSL VIP, it's possible to use cookie based persistence as well as other layer 7 techniques to control traffic flow to the IIS servers.

Certificates

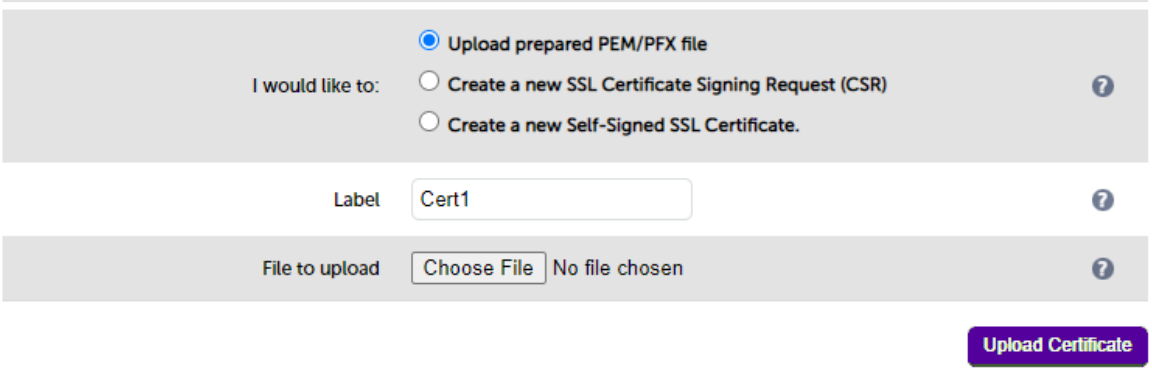
To enable the load balancer to perform SSL termination, an SSL certificate is required. If you already have an SSL certificate in either PFX or PEM file format, this can be uploaded to the Load balancer using the certificate upload option as explained below. Alternatively, you can create a Certificate Signing Request (CSR) on the load balancer and send this to your chosen CA to create a new certificate. For more information please refer to Generating a CSR on the Load Balancer.

Uploading Certificates

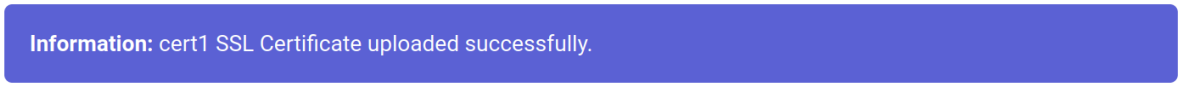
If you already have a certificate in either PEM or PFX format, this can be uploaded to the load balancer.

To upload a Certificate:

1. Using the WebUI, navigate to: ***Cluster Configuration > SSL Certificates***.
2. Click **Add a new SSL Certificate** and select *Upload prepared PEM/PFX file*.



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:



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

Exporting PFX Certificates from Windows Servers

When exporting certificates from Windows servers, make sure that ***Yes, export the private key*** is selected, this will enable the output format to be PFX. Also make sure that ***Include all certificates in the certification path if possible*** is selected.

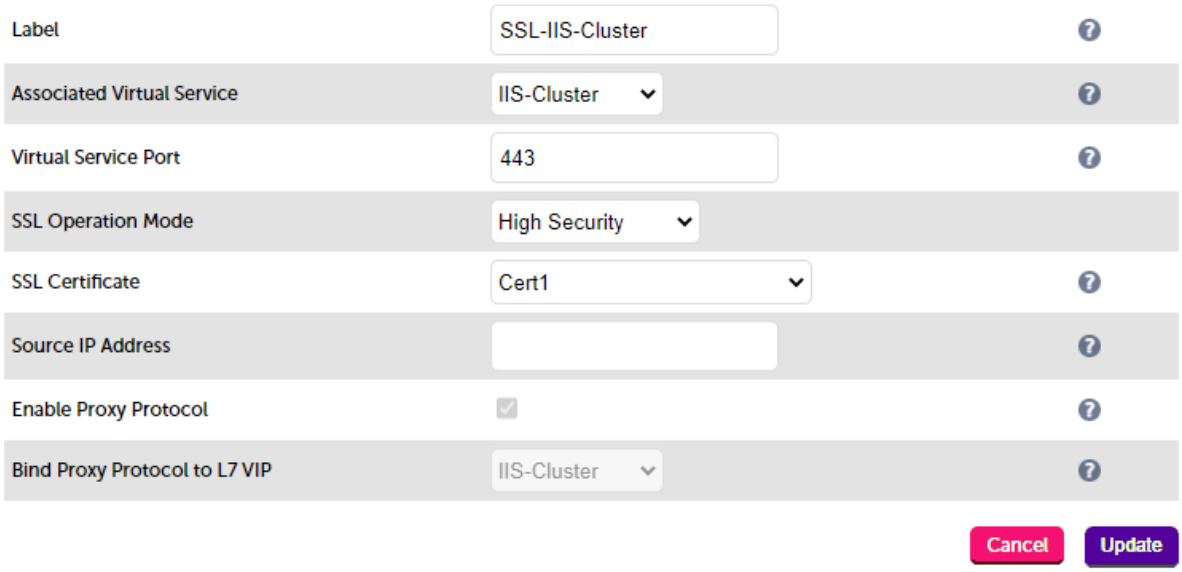
Creating a PEM file

For details, please refer to Creating a PEM File.

Configuring SSL Termination on the Load Balancer

To configure an SSL VIP:

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



2. Using the *Associated Virtual Service* drop-down, select the Virtual Service created above, e.g. **IIS_Cluster**.

> **Note** Once the VIP is selected, the *Label* field will be auto-populated with **SSL-IIS-Cluster**. This can be changed if preferred.

> **Note** 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.

> **Note** If you are following on from the example in Appliance & IIS Server Configuration – Using Layer 7 SNAT Mode, the IIS-Cluster VIP would need to be modified to make it a valid candidate for the *Associated Virtual Service* drop-down. Port 443 would need to be removed (i.e. set the port field to **80** not **80,443**). This is because HTTPS traffic would no longer be handled by the Layer 7 SNAT mode VIP, the SSL VIP would be used instead.

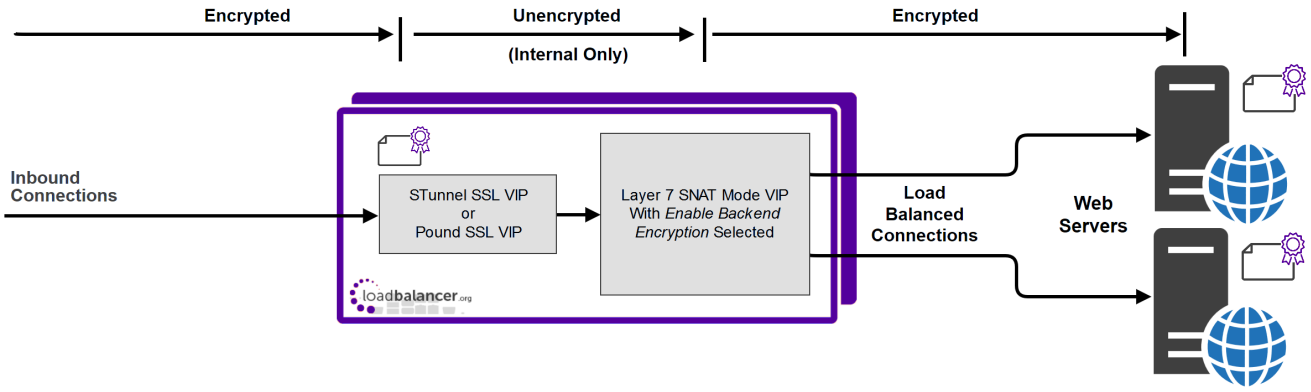
3. Leave *Virtual Service Port* set to **443**.
4. Leave *SSL operation Mode* set to **High Security**.
5. Select the required certificate from the *SSL Certificate* drop-down.
6. Click **Update**.

Now reload HAProxy and STunnel to apply the new settings using the links provided in the "Commit changes" box at the top of the screen.

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.

SSL Termination on the Load Balancer with Re-encryption (SSL Bridging)

> **Note** SSL termination on the load balancer can be very CPU intensive. In most cases, for a scalable solution, terminating SSL on the IIS servers is the best option.



In this case, an SSL VIP utilizing either STunnel (default & recommended) or Pound 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 and is also encrypted from the load balancer to the backend servers as shown above.

> **Note** This is similar to SSL Offload, the only difference is that the connection from the load balancer to the IIS servers is encrypted using the certificate located on the IIS server, this could be a self-signed certificate since no client connections are terminated here, only at the STunnel or Pound VIP.
>
> This mode can be enabled for the entire VIP and all associated IIS servers using the VIP option ***Enable Backend encryption*** or per IIS server using the ***Re-Encrypt to Backend*** option as detailed below.
>
> SSL termination on the load balancer can be very CPU intensive. In most cases, for a scalable solution, terminating SSL on the IIS servers is the best option.

To enable re-encryption at the Virtual Server level:

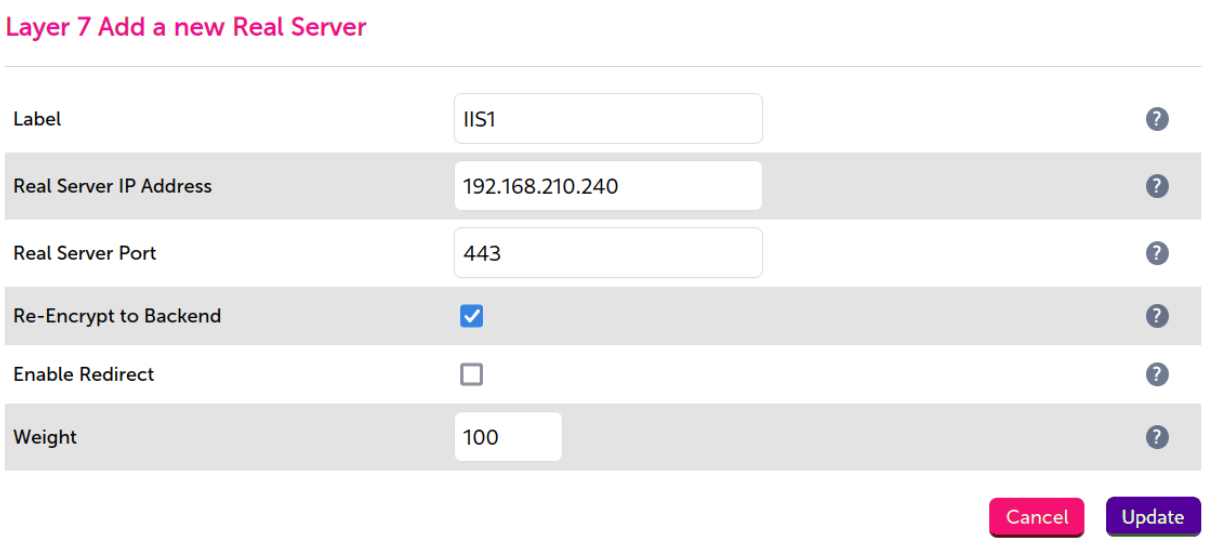
1. Use the WebUI menu option: *Cluster Configuration > Layer 7 – Virtual Servers* > **Modify**.

2. Enable the option ***Re-Encrypt to Backend***.
3. Click **Update**.
4. Now add the IIS servers ensuring that you specify the correct HTTPS port – typically 443.

> **Note** This setting only applies to IIS servers added ***after*** setting this option, it auto enables the Re-Encrypt to Backend option (see below) for all new IIS servers.

To enable re-encryption at the Real Server level:

1. For each Real Server use the WebUI menu option: *Cluster Configuration > Layer 7 – Real Servers* > **Modify**.



2. Set *Real Server Port* to **443**.
3. Enable the option *Re-Encrypt to Backend*.
4. Click **Update**.
5. Repeat for your other IIS server(s).

Now reload HAProxy and STunnel to apply the new settings using the links provided in the "Commit changes" box at the top of the screen.

10.2. Real Server (IIS) Health Checks

The load balancer performs regular health checks to ensure that each server in the cluster is healthy and able to accept client connections. The health check options depend on whether the VIP is defined at layer 4 or layer 7 as outlined below.

Layer 4

By default, a TCP connect health check is used for newly created layer 4 Virtual Services. The following tables list all options available:

Check Type	Description
Negotiate	Sends a request and looks for a specific response. This option enables the load balancer to perform a more robust check. For example, an HTTP check can be configured that requests a certain page and then looks for a specific word on that page.
Connect to port	Just do a simple connect to the specified port/service & verify that it's able to accept a connection.
Ping server	Sends an ICMP echo request packet to the Real Server.

Check Type	Description
External check	Use a custom script for the health check.
No checks, always Off	All Real Servers are off.
No checks, always On	All Real Servers are on (no checking).
5 Connects, 1 Negotiate	Do 5 connect checks and then 1 negotiate check.
10 Connects, 1 Negotiate	Do 10 connect checks and then 1 negotiate check.

Layer 7

By default, a TCP connect health check is used for newly created layer 7 Virtual Services. The following tables lists all options available:

Check Type	Description
Negotiate HTTP/HTTPS (GET)	Sends a request and looks for a specific response. This option enables the load balancer to perform a more robust check. For example, an HTTP or HTTPS check can be configured that requests a certain page and then looks for a specific word on that page.
Negotiate HTTP/HTTPS (HEAD)	Request the page headers of the page specified in Request to Send
Negotiate HTTP/HTTPS (OPTIONS)	Request the options of the page specified in Request to Send.
Connect to port	Just do a simple TCP connect to the specified port/service & verify that it's able to accept a connection.
External Script	Use a custom script for the health check.
MySQL	The check consists of sending two MySQL packets, one Client Authentication packet, and one QUIT packet, to correctly close the MySQL session. It then parses the MySQL Handshake Initialization packet and/or Error packet
No checks, always On	All Real Servers are assumed on (i.e. no checking)

> **Note** If a Negotiate check is selected and ***Response Expected*** is left blank, the appliance will check the location specified in ***Request to Send*** (if blank the root will be checked) and look for a **HTTP 200 OK** response from the Real Server.

> **Note** For full details on the options available, please refer to Real Server Health Monitoring & Control.

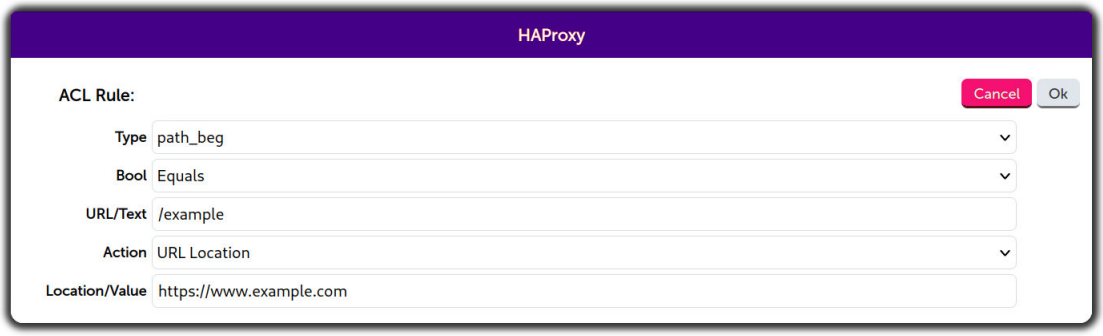
External Health-Check Scripts

Writing an external health check script enables the way the IIS servers are monitored to be customized. The example presented in this loadbalancer.org blog checks that it's possible to make a successful HTTP request and that the associated application pool is running.

> **Note** For more information, please also refer to External Health Check Scripts.

10.3. URL Rewriting / Content Switching (ACLs)

The WebUI supports the ability to create ACLs which can be used to control and direct HTTP traffic based on the rules defined. This option can be accessed in the *ACL Rules* section by clicking the **Add Rule** button when modifying a VIP.



- Multiple rules can be defined by using the **Add Rule** button multiple times.
- Once all rules have been defined, click **Update** to update the VIP and save the new configuration, then click **Reload HAProxy** at the top of the page to apply the new settings.

In the example above, requests are redirected to the URL location **https://www.example.com** if the path begins with **/example**.

E.g. if the requested URL is:

```
http://www.domain.com/example
```

the request is redirected to:

```
https://www.example.com
```

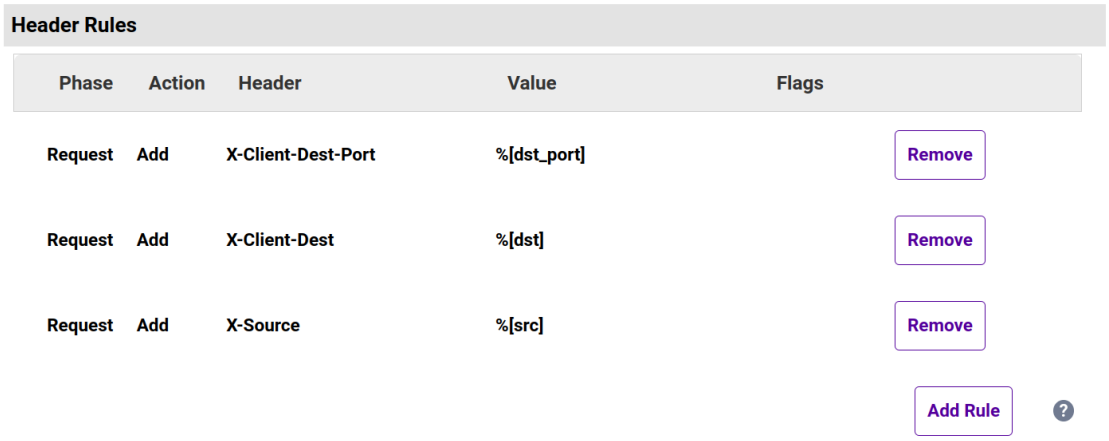
> **Note** For more information, please refer to ACLs (aka Content Switching) and URL Rewriting.

10.4. HTTP Header Manipulation

For HTTP mode virtual services, the WebUI supports the ability to add, set, delete, and replace HTTP header fields. This option can be accessed under the *Header Rules* section by clicking the **Add Rule** button when modifying a VIP.

Option	Description
Add	Append an HTTP header field. If a header field of the same name already exists then an additional header field will still be appended.

Option	Description
Set	Append an HTTP header field. If a header field of the same name already exists then it is first removed before a new one is appended. This is useful for handling security information which external users **must not** be able to set themselves.
Delete	Remove all HTTP header fields of a specified name.
Replace	Perform a regular expression powered "find and replace" operation on all HTTP header field values of a specified name.



- Multiple headers can be defined by repeatedly using the **Add Rule** button.
- Once all headers have been defined, click **Update** to update the VIP, then click **Reload HAProxy** at the top of the page to apply the new settings.

In the example above, the 3 header configuration rows result in the following headers being added to the requests sent from the appliance to the web servers:

- `X-Client-Dest-Port`, i.e. the port that the client connected to
- `X-Client-Dest`, i.e. the IP address that the client connected to
- `X-Source`, i.e. the client's source IP address

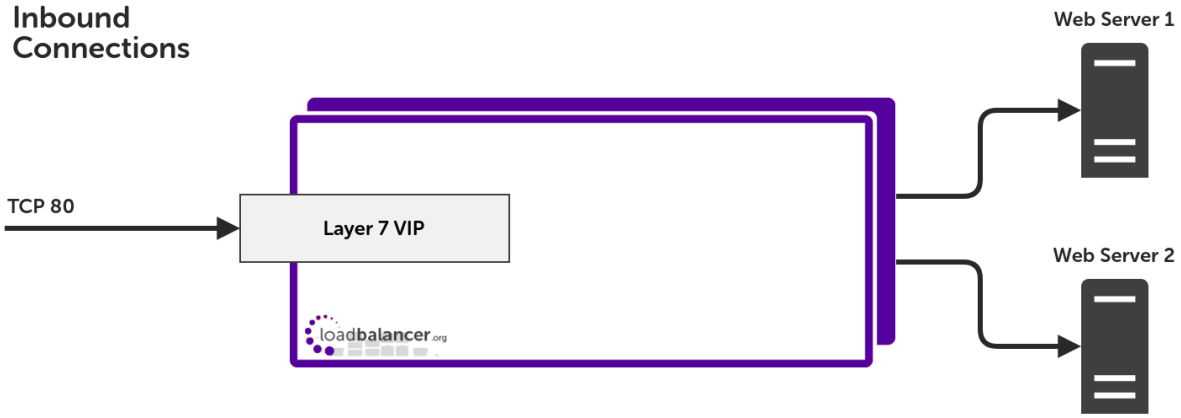
> **Note** For more information, please refer to Modifying HTTP Header Fields.

10.5. Web Application Firewall (WAF)

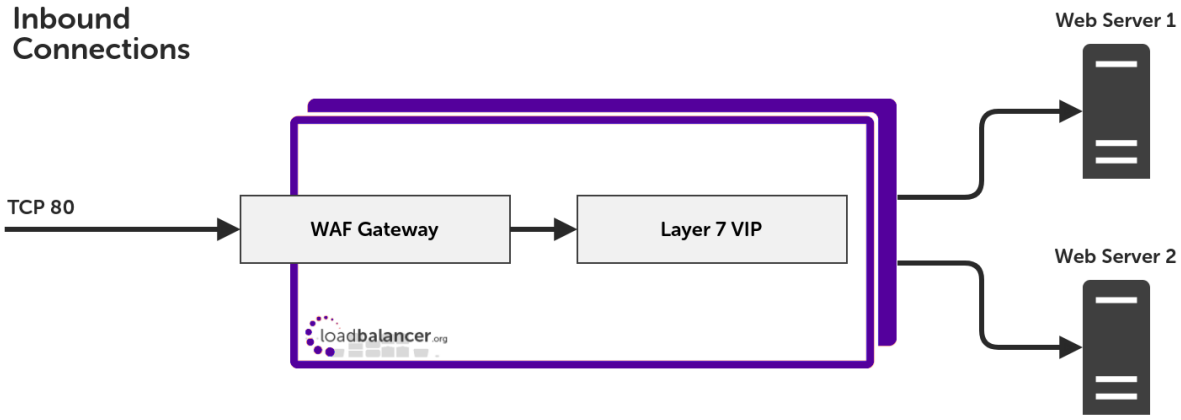
The load balancer includes a built in WAF that can be deployed if required. The WAF is based on the ModSecurity Open Source Project and includes a default vulnerability rule-set based on the "OWASP top 10". This defines the top 10 areas of vulnerability that can effect Web Applications.

The load balancer supports the ability to define multiple WAF gateways. Each gateway is associated with a layer 7 VIP when created. On creation, the data path is automatically modified so that the WAF becomes the initial connection point for inbound client connections as illustrated below:

Data flow before WAF is deployed:



Modified data flow once WAF is deployed:



Note

When defining a WAF Gateway on the load balancer, the associated layer 7 VIP must be selected from a drop-down list. This enables the WAF to be automatically configured to listen on the same TCP socket as the original layer 7 VIP. The WAF gateway is then automatically configured to forward packets to the original layer 7 VIP.

Each WAF gateway is associated with one layer 7 VIP.

Once the WAF gateway is defined, the ***Label***, ***IP Address***, ***Port*** and ***Protocol*** of the associated layer 7 VIP cannot be edited to ensure the association remains intact. If changes to these settings are required, remove the WAF, make the changes, then recreate the WAF.

Each WAF gateway is comprised of an additional layer 7 VIP which acts as the WAF frontend and an Apache/ModSecurity config. Both are auto-created when the WAF Gateway is configured.

For full details on creating and configuring a WAF, please refer to Web Application Firewall.

10.6. Server Feedback Agent

The load balancer can modify the weight (amount of traffic) of each server by gathering data from either a custom agent or an HTTP server. For layer 4 VIPs the feedback method can be set to either agent or HTTP, for Layer 7 VIPs only the agent method is supported. By default the agent listens on TCP port 3333 but this can be changed if required.

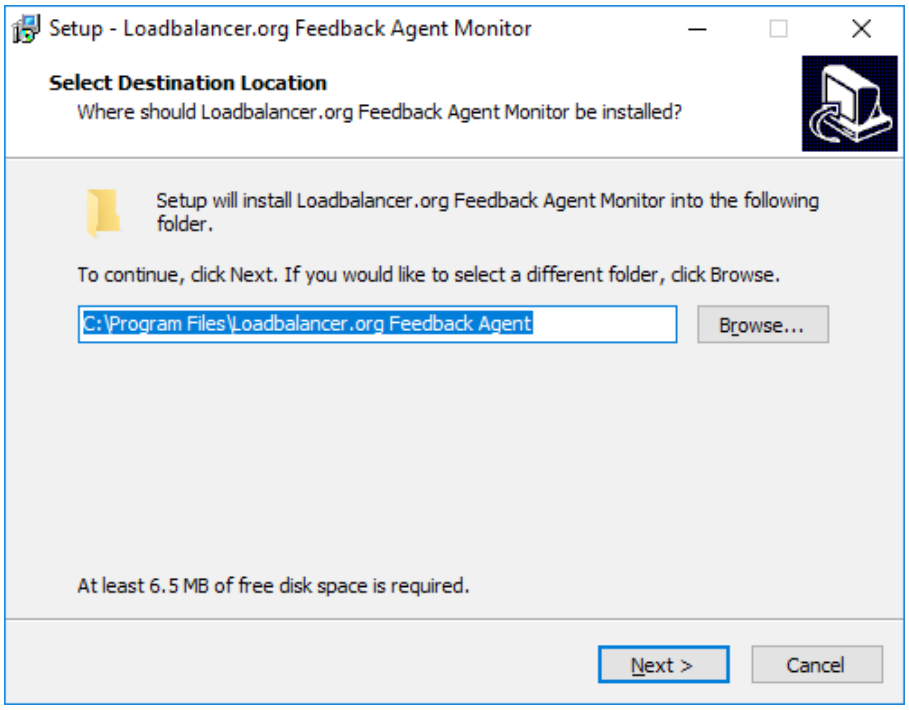
A telnet to port 3333 on a Real Server with the agent installed will return the current idle stats as an integer value in the range 0 - 100. The figure returned can be related to CPU utilization, RAM usage or a combination of both. This can be configured using the XML configuration file located in the agents installation folder (by default C:\ProgramData\LoadBalancer.org\LoadBalancer).

The load balancer typically expects a 0-99 integer response from the agent which by default relates to the current CPU idle state, e.g. a response of 92 would imply that the Real Servers CPU is 92% idle. The load balancer will then use the formula (92/100 * requested_weight) to find the new optimized weight.

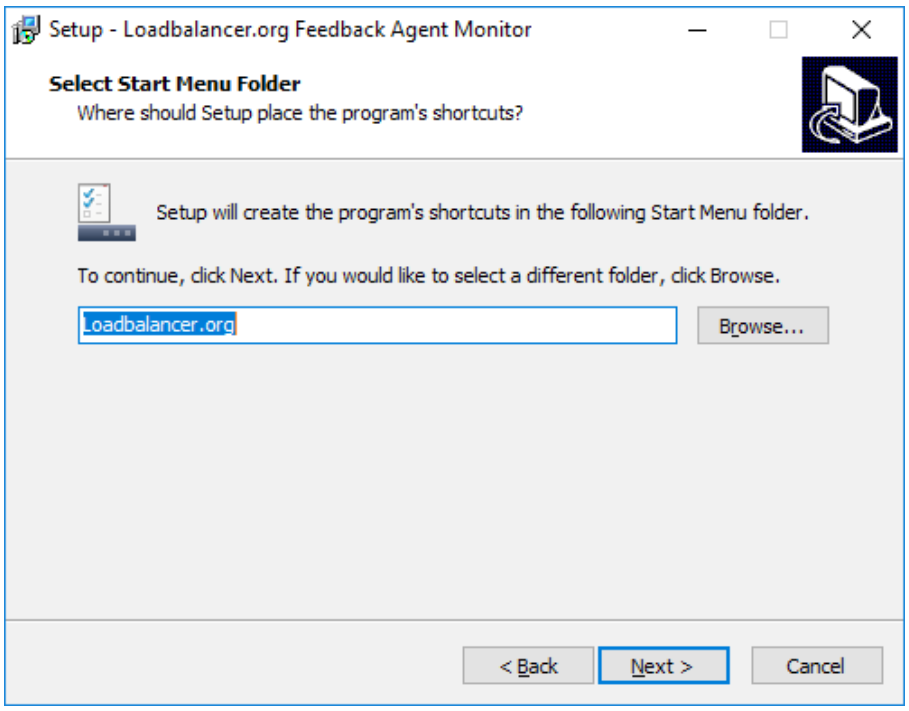
> **Note**
>
> The 'Requested Weight' is the weight set in the WebUI for each Real Server.
>
> For more information about the feedback agent, please refer to this blog.

Windows Agent

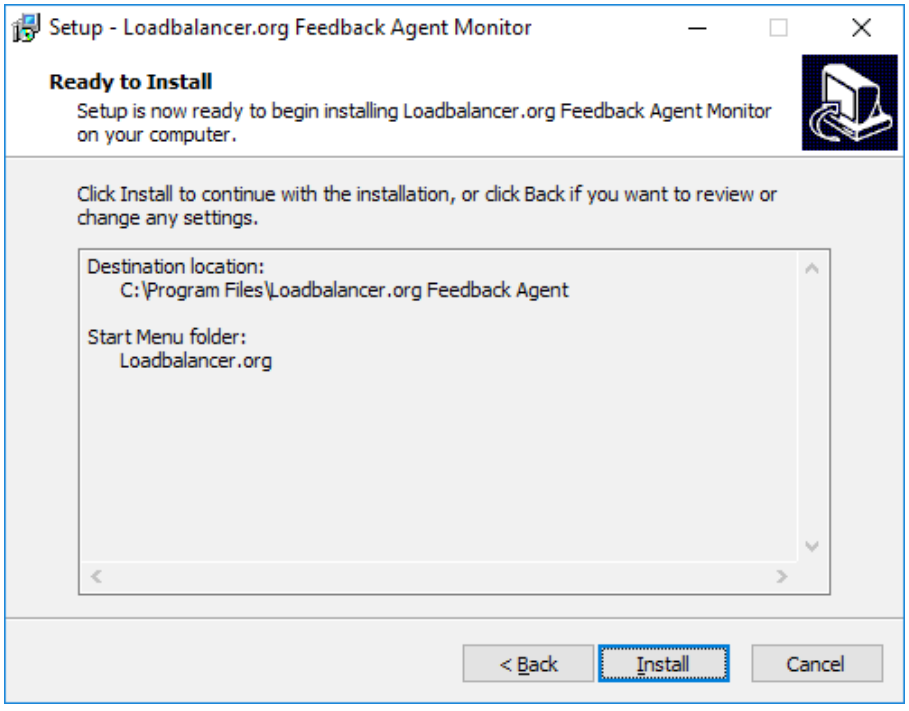
The latest Windows feedback agent can be downloaded from here. To install the agent, run **loadbalanceragent.msi** on each Real Server:



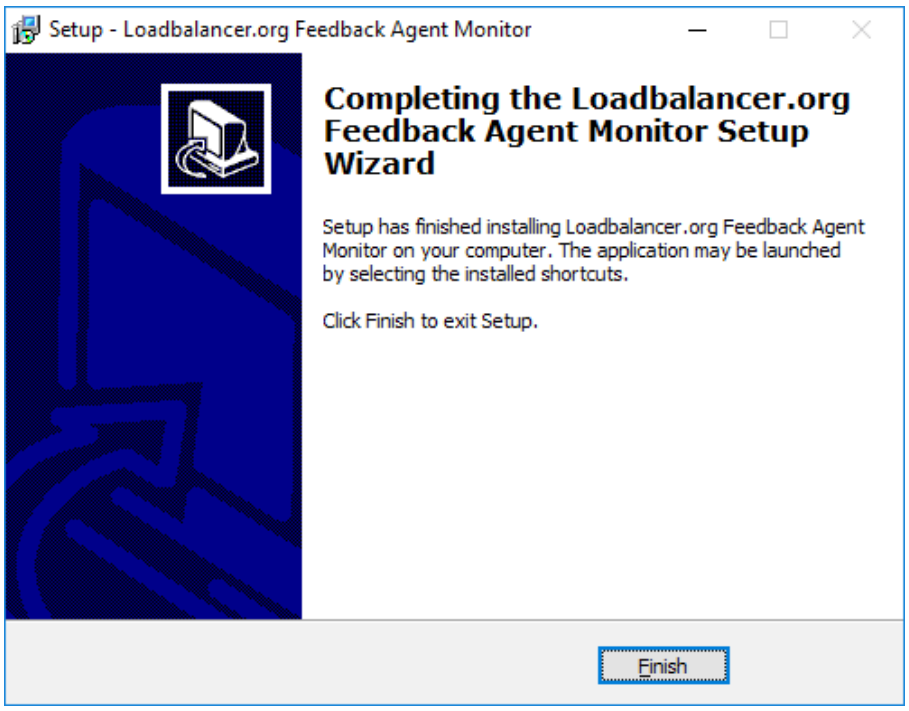
Leave the default location or change according to your requirements, click **Next**.



Leave the default location or change according to your requirements, click **Next**.



Click **Install** to start the installation process.

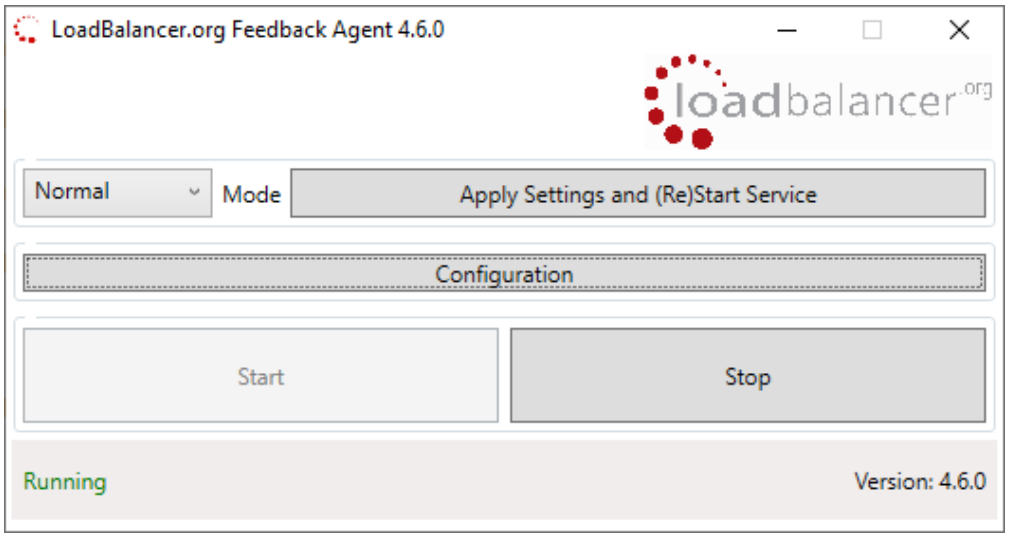


Click **Finish**.

Once the installation is complete, the Feedback Agent service is started.

Controlling the Agent

The Feedback Agent service (**LBCPUMon**) can be controlled & configured using the *Feedback Agent Service Monitor* program. By default this can be accessed from: *Start> Loadbalancer.org*.



Linux/Unix Agent

The Linux feedback agent files can be downloaded using the following links:

readme file: https://downloads.loadbalancer.org/agent/linux/v4.1/readme.txt
xinetd file: https://downloads.loadbalancer.org/agent/linux/v4.1/lb-feedback
feedback script: https://downloads.loadbalancer.org/agent/linux/v4.1/lb-feedback.sh

Installation & Testing

Install xinetd - if not already installed:

```
apt-get install xinetd
```

Insert this line into /etc/services:

```
lb-feedback 3333/tcp # Loadbalancer.org feedback daemon
```

Then run the following commands:

```
cp lb-feedback.sh /usr/bin/lb-feedback.sh
chmod +x /usr/bin/lb-feedback.sh
cp lb-feedback /etc/xinetd.d/lb-feedback
chmod 644 /etc/xinetd.d/lb-feedback
/etc/init.d/xinetd restart
```

To test the agent:

```
telnet 127.0.0.1 3333
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.
95%
Connection closed by foreign host.
```

> **Note** The agent files must be installed on each Real Server.

Custom HTTP Agent

You can use any HTTP server responding on port 3333 to give feedback information to the load balancer. The format of this information must be an integer number of 0-100 without any header information. Using this method, you can generate a custom response based on your application's requirements.

10.7. Load Balancer Transparency

Layer 4

Both Layer 4 DR mode and layer 4 NAT mode are transparent by default. This means that IIS will log the actual IP address of the client rather than the IP address of the load balancer.

Layer 7

Because layer 7 is based on a proxy (HAProxy) it is not transparent by default, therefore IIS logs will show the load balancer's IP address rather than the client's IP. However, the load balancer can be configured to provide the actual client IP address to the IIS servers in 2 ways:

1. By inserting a header that contains the client IP source address. For HTTP traffic the **X-Forwarded-For (XFF)** header is used, for TCP traffic the **Proxy Protocol Header** is used.

> **Note** For more details of XFF headers please refer to this link, for more details of Proxy Protocol Headers please refer to this link.

2. By modifying the Source Address field of the IP packets and replacing the IP address of the load balancer with the IP address of the client. The load balancer uses TProxy for this purpose.

These methods can be used independently or in combination to achieve a range of objectives. For more information and details of how to use these methods, please refer Transparency at Layer 7.

11. Testing & Verification

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

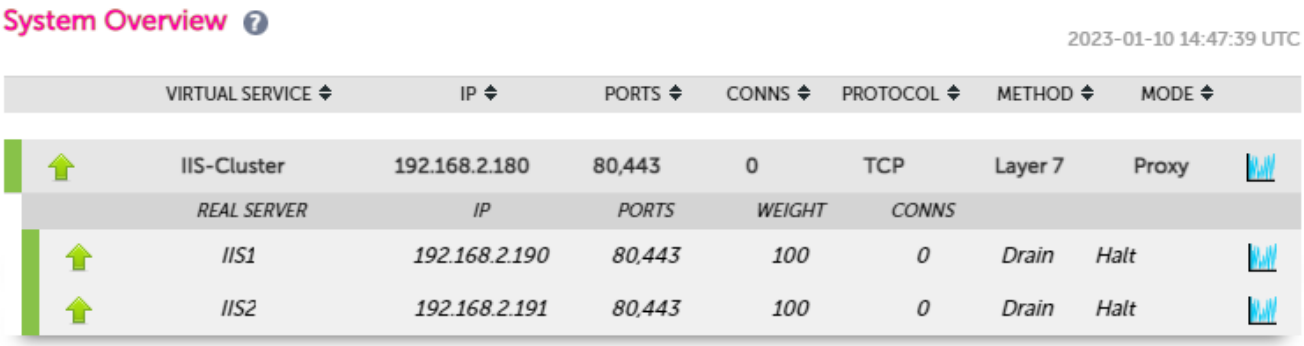
To test a web server based configuration, add a test page to each web server, e.g. test.html and put the server name on this page for easy identification during the tests.

Use two or more clients to do the testing. Open up a web browser on each test client and enter the VIP address or the corresponding URL, e.g. **http://192.168.2.180**.

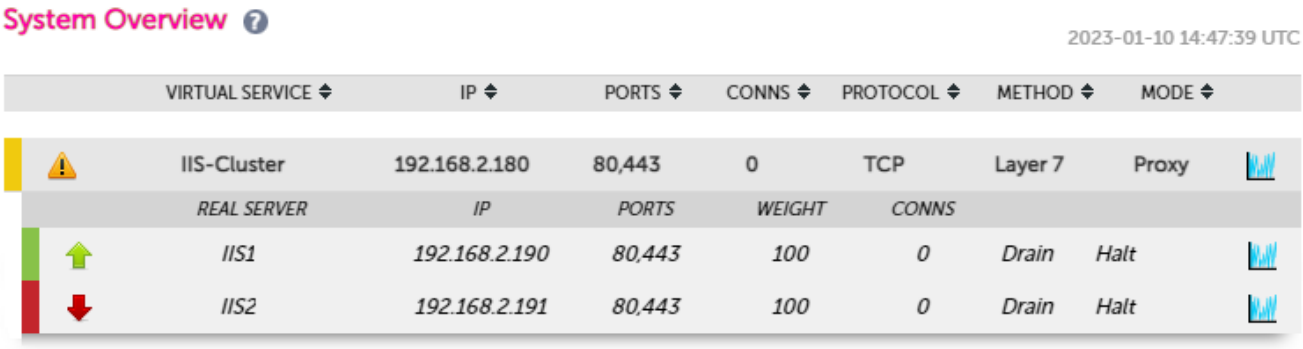
Each client should see a different server name because of the load balancing algorithm in use, i.e. they are being load balanced across the cluster.

11.1. Using the System Overview

The System Overview can be accessed via the WebUI. It shows a graphical view of all VIPs & RIPs (i.e. the IIS servers) and shows the state/health of each server as well as the state of the cluster as a whole. The example below shows that all IIS servers are healthy (green) and available to accept connections:



If one of the servers within the cluster fails its health check, that server will be colored red and the cluster will be colored yellow as shown below:



Make sure that all servers are up (green) and verify that clients can connect to the VIP and access all load balanced services.

> **Note** Make sure that DNS points at the VIP rather than individual servers.

12. Technical Support

If you have any questions regarding the appliance or would like assistance designing your deployment, please don't hesitate to contact our support team: support@loadbalancer.org.

13. Further Documentation

For additional information, please refer to the Administration Manual.

14. Appendix

14.1. Solving the ARP Problem

When using Layer 4 DR mode, the ARP problem must be solved. This involves configuring each IIS server to be able to receive traffic destined for the VIP, and ensuring that each IIS server does not respond to ARP requests for the VIP address – only the load balancer should do this.

The steps below are for Windows 2012 & later, for earlier versions of Windows please refer to the Administration Manual.

Windows Server 2012 & Later

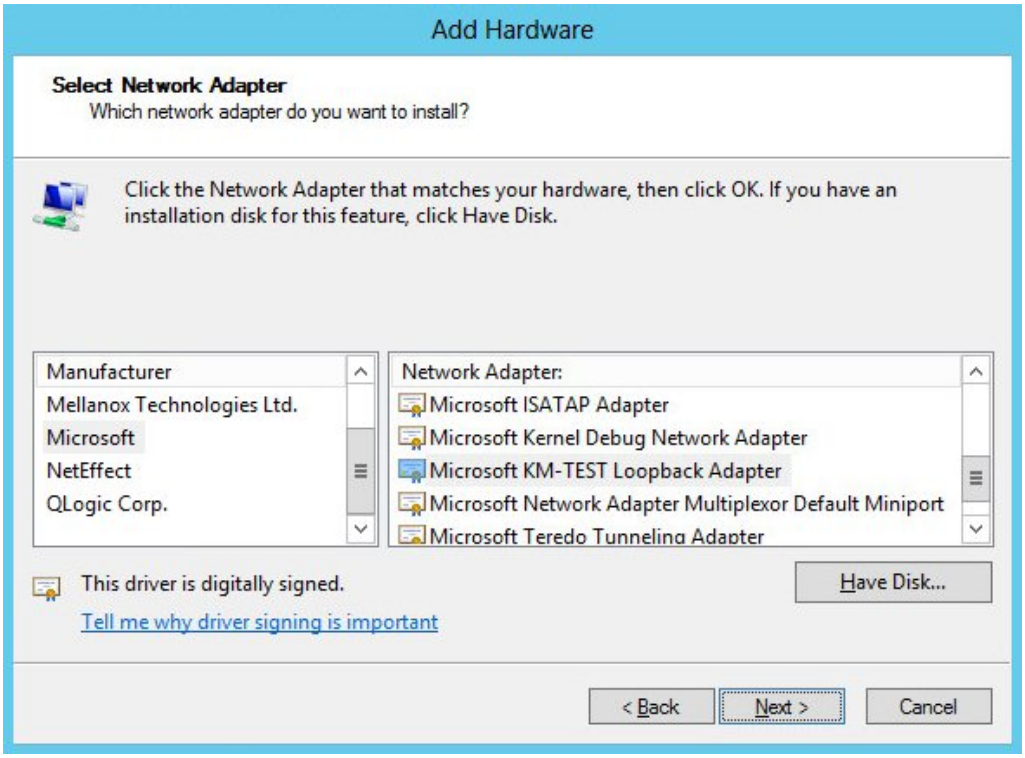
Windows Server 2012 and later support Direct Routing (DR) mode through the use of the Microsoft Loopback Adapter that must be installed and configured on each load balanced (Real) Server. The IP address configured on the Loopback Adapter must be the same as the Virtual Service (VIP) address. This enables the server to receive packets that have their destination set as the VIP address. If a Real Server is included in multiple DR mode VIPs, an IP address for each VIP must be added to the Loopback Adapter.

In addition, steps must be taken to set the strong/weak host behavior on each Real Server. This is used to either prevent or allow interfaces to receive packets destined for a different interface on the same server.

> **Important** The following 3 steps must be completed on ***all*** Real Servers associated with the VIP.

Step 1 of 3: Install the Microsoft Loopback Adapter

1. Click **Start**, then run **hdwwiz** to start the Hardware Installation Wizard.
2. Once the Wizard has started, click **Next**.
3. Select **Install the hardware that I manually select from a list (Advanced)**, click **Next**.
4. Select **Network adapters**, click **Next**.



5. Select **Microsoft** & **Microsoft KM-Test Loopback Adapter**, click **Next**.
6. Click **Next** to start the installation, when complete click **Finish**.

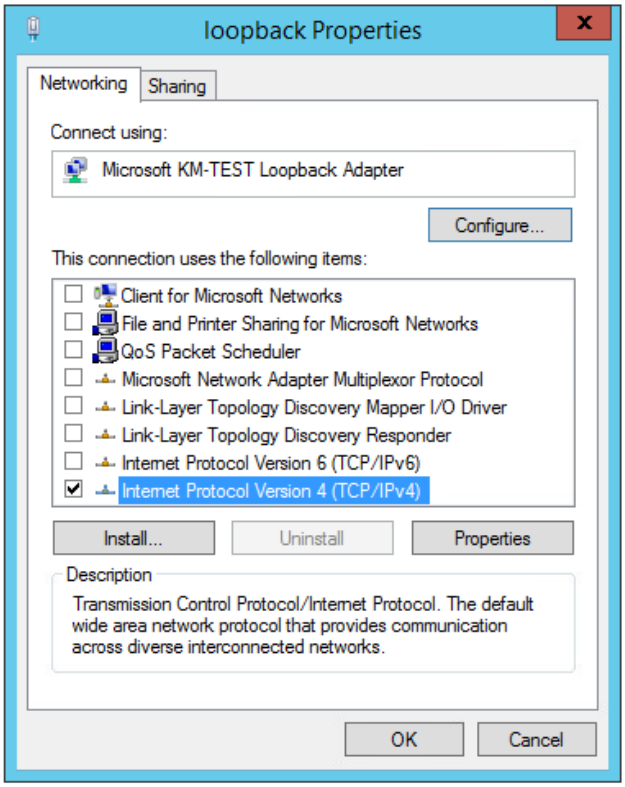
Step 2 of 3: Configure the Loopback Adapter

1. Open Control Panel and click **Network and Sharing Center**.
2. Click **Change adapter settings**.
3. Right-click the new Loopback Adapter and select **Properties**.

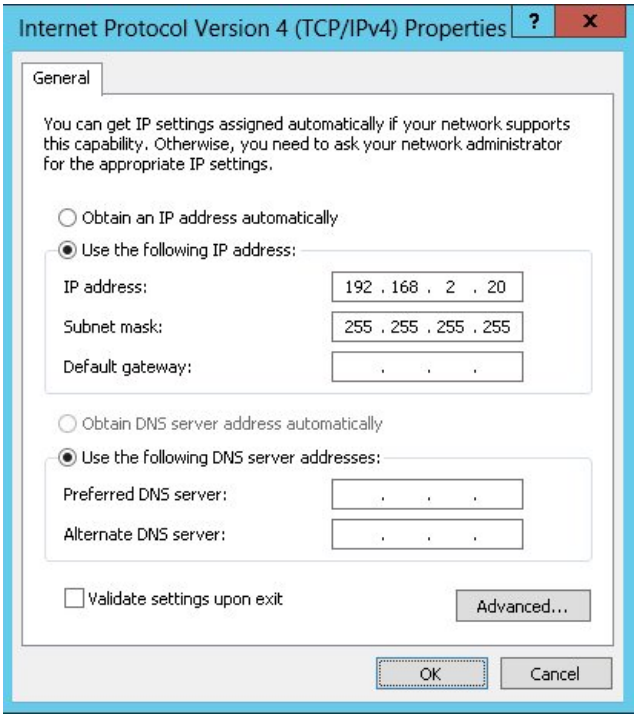
> **Note** You can configure IPv4 or IPv6 addresses or both depending on your requirements.

IPv4 Addresses

1. Uncheck all items except **Internet Protocol Version 4 (TCP/IPv4)** as shown below:



2. Ensure that **Internet Protocol Version (TCP/IPv4)** is selected, click **Properties** and configure the IP address to be the same as the Virtual Service address (VIP) with a subnet mask of **255.255.255.255**, e.g. **192.168.2.20/255.255.255.255** as shown below:



> **Note** **192.168.2.20** is an example, make sure you specify the correct VIP address.

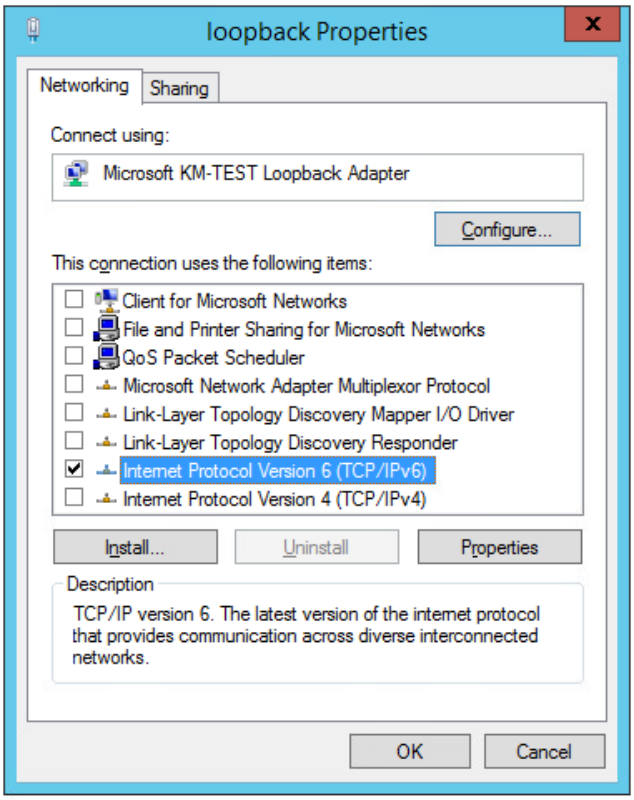
> **Note** If a Real Server is included in multiple DR mode VIPs, an IP address for each VIP must be

added to the Loopback Adapter.

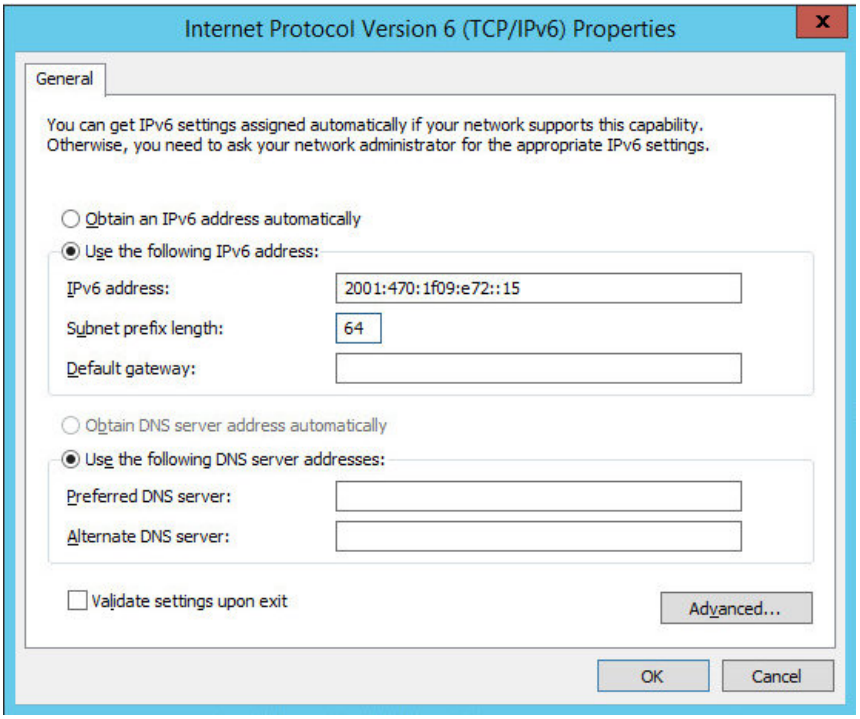
3. Click **OK** then click **Close** to save and apply the new settings.

IPv6 Addresses

1. Uncheck all items except **Internet Protocol Version 6 (TCP/IPv6)** as shown below:



2. Ensure that **Internet Protocol Version (TCP/IPv6)** is selected, click **Properties** and configure the IP address to be the same as the Virtual Service (VIP) and set the ***Subnet Prefix Length*** to be the same as your network setting, e.g. **2001:470:1f09:e72::15/64** as shown below:



> **Note** **2001:470:1f09:e72::15/64** is an example, make sure you specify the correct VIP address.

> **Note** If a Real Server is included in multiple DR mode VIPs, an IP address for each VIP must be added to the Loopback Adapter.

3. Click **OK** then click **Close** to save and apply the new settings.

Step 3 of 3: Configure the strong/weak host behavior

The strong/weak host behavior can be configured using either of the following 2 methods:

- Option 1 - Using network shell (netsh) commands
- Option 2 - Using PowerShell cmdlets

The commands in this section assume that the LAN Adapter is named "**net**" and the Loopback Adapter is named "**loopback**" as shown in the example below:

> **Important** Either adjust the commands to use the names allocated to your LAN and loopback adapters, or rename the adapters before running the commands. Names are case sensitive so make sure

that the interface names used in the commands match the adapter names exactly.

Option 1 - Using Network Shell (netsh) Commands

To configure the correct strong/weak host behavior run the following commands:

For IPv4 addresses:

```
netsh interface ipv4 set interface "net" weakhostreceive=enabled
netsh interface ipv4 set interface "loopback" weakhostreceive=enabled
netsh interface ipv4 set interface "loopback" weakhostsend=enabled
```

For IPv6 addresses:

```
netsh interface ipv6 set interface "net" weakhostreceive=enabled
netsh interface ipv6 set interface "loopback" weakhostreceive=enabled
netsh interface ipv6 set interface "loopback" weakhostsend=enabled
netsh interface ipv6 set interface "loopback" dadtransmits=0
```

Option 2 - Using PowerShell Cmdlets

For IPv4 addresses:

```
Set-NetIpInterface -InterfaceAlias loopback -WeakHostReceive enabled -WeakHostSend enabled
-DadTransmits 0 -AddressFamily IPv4
```

```
Set-NetIpInterface -InterfaceAlias net -WeakHostReceive enabled -AddressFamily IPv4
```

For IPv6 Addresses:

```
Set-NetIpInterface -InterfaceAlias loopback -WeakHostReceive enabled -WeakHostSend enabled
-DadTransmits 0 -AddressFamily IPv6
```

```
Set-NetIpInterface -InterfaceAlias net -WeakHostReceive enabled -AddressFamily IPv6
```

14.2. Configuring HA - Adding a Secondary Appliance

Our recommended configuration is to use a clustered HA pair of load balancers to provide a highly available and resilient load balancing solution.

We recommend that the Primary appliance is configured first and then the Secondary should be added. Once the Primary and Secondary are paired, all load balanced services configured on the Primary are automatically replicated to the Secondary 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 Secondary. 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 Secondary) will take over.

Non-Replicated Settings

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

WebUI Main Menu Option	Sub Menu Option	Description
Local Configuration	Hostname & DNS	Hostname and DNS settings
Local Configuration	Network Interface Configuration	All network settings including IP address(es), bonding configuration and VLANs
Local Configuration	Routing	Routing configuration including default gateways and static routes
Local Configuration	System Date & time	All time and date related settings
Local Configuration	Physical – Advanced Configuration	Various settings including Internet Proxy, Management Gateway, Firewall connection tracking table size, NIC offloading, SMTP relay, logging and Syslog Server
Local Configuration	Security	Appliance security settings
Local Configuration	SNMP Configuration	Appliance SNMP settings
Local Configuration	Graphing	Appliance graphing settings
Local Configuration	License Key	Appliance licensing
Maintenance	Software Updates	Appliance software update management
Maintenance	Firewall Script	Appliance firewall (iptables) configuration
Maintenance	Firewall Lockdown Wizard	Appliance management lockdown settings

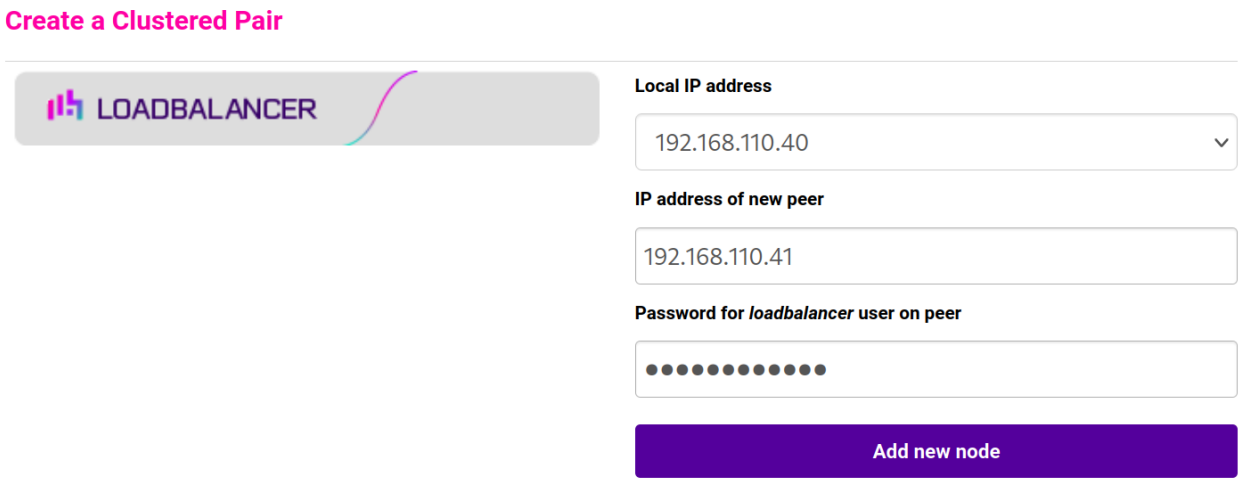
> **Important** Make sure that if these settings/updates have been configured on the Primary appliance, they're also configured on the Secondary appliance.

Adding a Secondary Appliance - Create an HA Clustered Pair

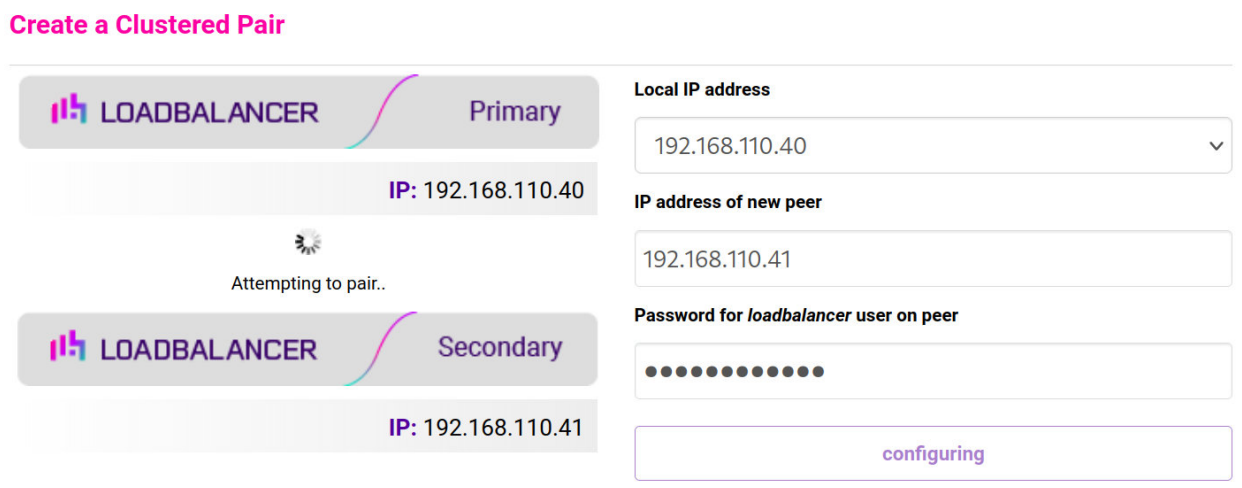
> **Note** If you have already run the firewall lockdown wizard on either appliance, you'll need to ensure that it is temporarily disabled on both appliances whilst performing the pairing process.

1. Deploy a second appliance that will be the Secondary and configure initial network settings.

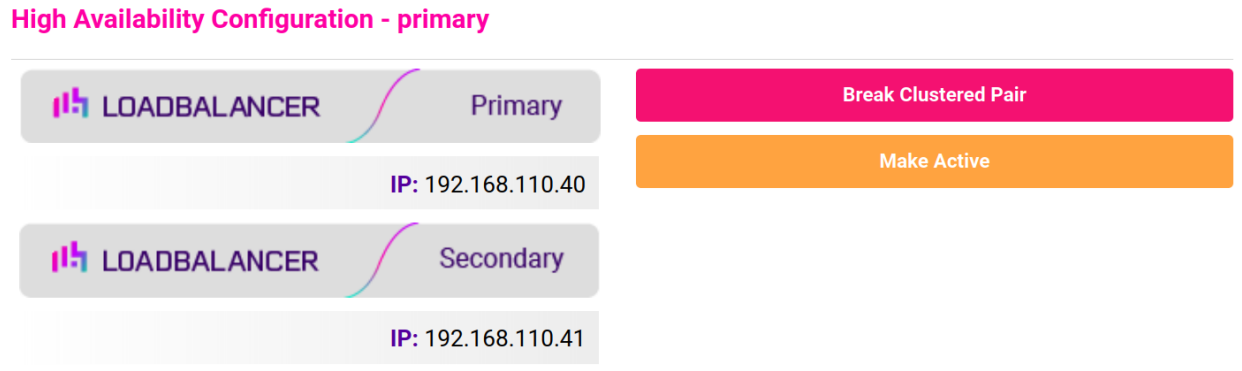
2. Using the WebUI on the Primary appliance, navigate to: *Cluster Configuration > High-Availability Configuration*.



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



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



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

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

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

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

15. Document Revision History

Version	Date	Change	Reason for Change	Changed By
1.7.0	9 August 2019	Styling and layout	General styling updates	RJC
1.7.1	1 June 2020	New title page Updated Canadian contact details New screenshot for creating a layer 4 VIP	Branding update Change to Canadian contact details Changes to the appliance WebUI	AH
1.7.2	17 June 2021	Various minor updates		RJC
1.8.0	1 January 2022	Converted the document to AsciiDoc	Move to new documentation system	AH, RJC, ZAC
1.8.1	13 April 2022	Updated ACL instructions Updated HTTP header manipulation instructions	Changes to the appliance WebUI	AH
1.8.2	26 April 2022	Updated SSL related content to reflect latest software version	New software release	RJC
1.8.3	28 September 2022	Updated layer 7 VIP and RIP creation screenshots	Reflect changes in the web user interface	AH
1.8.4	5 January 2023	Combined software version information into one section Added one level of section numbering Added software update instructions Added table of ports used by the appliance Reworded 'Further Documentation' section Removed references to the colour of certain UI elements	Housekeeping across all documentation	AH
1.8.5	10 January 2023	Updated Testing & Verification section	General Improvements	RJC
1.8.6	2 February 2023	Updated screenshots	Branding update	AH

Version	Date	Change	Reason for Change	Changed By
1.8.7	7 March 2023	Removed conclusion section	Updates across all documentation	AH
1.9.0	24 March 2023	New document theme Modified diagram colours	Branding update	AH

LOADBALANCER

Visit us: www.loadbalancer.org

Phone us: +44 (0)330 380 1064

Phone us: +1 833 274 2566

Email us: info@loadbalancer.org

Follow us: @loadbalancer.org

About Loadbalancer.org

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