About this Guide

This guide details the configuration of Loadbalancer.org appliances for deployment with Microsoft Internet Information Services (IIS).

For an introduction on setting up the appliance as well as more technical information, please also refer to our quick-start guides and full administration manuals which are available at the following links:

Version 7 Documentation

Version 8 Documentation

Loadbalancer.org Appliances Supported

All our products can be used with IIS. The complete list of models is shown below:

<table>
<thead>
<tr>
<th>Discontinued Models</th>
<th>Current Models *</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise R16</td>
<td>Enterprise R20</td>
</tr>
<tr>
<td>Enterprise VA R16</td>
<td>Enterprise MAX</td>
</tr>
<tr>
<td>Enterprise VA</td>
<td>Enterprise 10G</td>
</tr>
<tr>
<td></td>
<td>Enterprise R320</td>
</tr>
<tr>
<td></td>
<td>Enterprise VA R20</td>
</tr>
<tr>
<td></td>
<td>Enterprise VA MAX</td>
</tr>
<tr>
<td></td>
<td>Enterprise AWS **</td>
</tr>
<tr>
<td></td>
<td>Enterprise AZURE **</td>
</tr>
</tbody>
</table>

* For full specifications of these models please refer to: http://www.loadbalancer.org/products

** Some features may not be supported, please check with Loadbalancer.org support

Loadbalancer.org Software Versions Supported

- v7.3.2 and later
  N.B. this guide includes configuration steps for v7.6 & later. For older versions of the appliance please contact Loadbalancer.org sales or support

Microsoft IIS Software Versions Supported

- Microsoft IIS – all versions
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 release is v8.5 which is part of Windows 2012 R2. IIS v8.5 is 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.

Load Balancing IIS

The Basics

Sharing the Load

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.

A Virtual Service (VIP) is configured on the load balancer and the related IIS servers are then defined. Clients then connect to the VIP rather than individual IIS servers. Incoming requests are then distributed to the IIS servers based on the load balancing algorithm selected (e.g. round robin, least connection).

Adding additional capacity as demand grows then becomes straightforward and can be achieved by simply associating additional IIS servers to the Virtual Service.

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.

Providing Resilience

Typically, two appliances are deployed as an active / passive pair. This ensures that a single point of failure is not introduced. A heartbeat signal between the two units is used to ensure that should the active unit fail, the passive unit takes over.

Ports & Protocols

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

<table>
<thead>
<tr>
<th>Port</th>
<th>Protocol</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>TCP/HTTP</td>
<td>HTTP web traffic</td>
</tr>
<tr>
<td>443</td>
<td>TCP/HTTPS</td>
<td>HTTPS web traffic</td>
</tr>
</tbody>
</table>
**SSL & Certificates**

For secure websites & web pages, SSL / TLS is used. This ensures that data is encrypted between client and server. SSL certificates can either be installed on the load balancer (a.k.a. **SSL off-loading**) or on the IIS (a.k.a. **SSL pass-through**). When installed on the load balancer, it’s possible to enable re-encryption so that the connection from the load balancer to the IIS servers is protected (a.k.a. **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.

**Persistence (a.k.a. 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. For HTTP, this can be either based on source IP address or cookies, both methods ensure that repeated connections from a particular client within a session are always sent to the same back-end IIS server.
Deployment Overview

The following diagram provides a simple illustration to indicate how the load balancer is deployed with multiple IIS servers.

Clients connect to the Virtual Service (VIP) on the load balancer rather than connecting directly to one of the IIS servers. These connections are then load balanced across the IIS servers to distribute the load according to the load balancing algorithm selected.

NOTE: The load balancer can be deployed as a single unit, although Loadbalancer.org strongly recommends a clustered pair for resilience & high availability.
Load Balancer Deployment Methods

Various deployment methods are supported. Each method is explained in the following sections.

Layer 4

**DR Mode (aka Direct Server Return)**

One-arm Direct Routing (DR) mode is a very high performance solution that requires little change to your existing infrastructure. *N.B. Kemp, Brocade & A10 Networks call this Direct Server Return and F5 call it N-Path.*

- Direct Routing 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 it 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 Servers own IP address and the VIP
- The Real Server should not respond to ARP requests for the VIP. Only the load balancer should do this. Configuring the Real Servers in this way is referred to as **Solving the ARP Problem.** Please refer to page 43 in the appendix for solving the ARP problem for Windows 2012
- 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 in DR mode i.e. having a different RIP port than the VIP port
- DR mode is transparent, i.e. the Real Server will see the source IP address of the client
Network Address Translation (NAT Mode)

Sometimes it’s not possible to use DR mode. The two most common reasons being: if the application cannot bind to the RIP & VIP at the same time; or if the host operating system cannot be modified to handle the ARP problem. The second choice is Network Address Translation (NAT) mode. This is also a high performance solution but it requires the implementation of a two arm infrastructure with an internal and external subnet to carry out the translation (the same way a firewall works).

- The load balancer translates all requests from the external Virtual Service to the internal Real Servers
- Normally eth0 is used for the internal network and eth1 is used for the external network although this is not mandatory. If the Real Servers require Internet access, Autonat should be enabled using the WUI option: Cluster Configuration > Layer 4 – Advanced Configuration, the external interface should be selected
- NAT mode can be deployed in the following ways:

  2-arm (using 2 Interfaces), 2 subnets (as shown above) - One interface on the load balancer is connected to subnet1 and the second interface and Real Servers are connected to subnet2. The VIP is brought up in subnet1. The default gateway on the Real Servers is set to be an IP address in subnet2 on the load balancer. Clients can be located in subnet1 or any remote subnet provided they can route to the VIP

  2-arm (using 1 Interface), 2 subnets - same as above except that a single interface on the load balancer is allocated 2 IP addresses, one in each subnet

  1-arm (using 1 Interface), 1 subnet - Here, the VIP is brought up in the same subnet as the Real Servers. For clients located on this subnet, 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 clients located in remote networks the default gateway on the Real Servers must also be set to be an IP address on the load balancer. For more details please refer to the administration manual

- If you want Real Servers to be accessible on their own IP address for non-load balanced services, e.g. SMTP or RDP, you will need to setup individual SNAT and DNAT firewall script rules for each Real Server or add additional VIPs for this - please refer to the administration manual
- NAT mode is transparent, i.e. the Real Server will see the source IP address of the client
- Port translation is possible in NAT mode, i.e. VIP:80 → RIP8080 is possible
Layer 7

Source Network Address Translation (HAProxy)

If your application requires that the load balancer handles cookie insertion then you need to use the SNAT configuration. This mode is also used with numerous Microsoft applications such as Exchange, Sharepoint, Lync etc.

This mode has the advantage of a one arm configuration and does not require any changes to the application servers. However, since the load balancer is acting as a full proxy it doesn't have the same raw throughput as the layer 4 methods.

The network diagram for the Layer 7 HAProxy SNAT mode is very similar to the Direct Routing example except that no re-configuration of the Real Servers is required. The load balancer proxies the application traffic to the servers so that the source of all traffic becomes the load balancer.

- As with other modes a single unit does not require a Floating IP, although it is recommended to make adding a slave unit easier
- SNAT is a full proxy and therefore load balanced Real Servers do not need to be changed in any way
- Because SNAT is a full proxy any server in the cluster can be on any accessible subnet including across the Internet or WAN
- SNAT 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 balancers IP address. If required, this can be solved by either enabling TProxy on the load balancer, or for HTTP, using X-forwarded-For headers. Please refer to the administration manual for more details.
**Loadbalancer.org Recommended Method**

One-arm layer 4 DR mode is the fastest option so where possible this is recommended. If this is not feasible for any reason – e.g. the IIS Servers are located on a different subnet to the VIP, then two-arm layer 4 NAT mode is suggested as this also offers high performance.

In situations where the network topology does not allow DR mode to be used (e.g. because VIPs & RIPv not located in different subnets) or NAT mode (e.g. because its not feasible to set the IIS Servers default gateway to be the load balancer), layer 7 SNAT mode is suggested since the IIS Servers can be positioned on any routeable network and no IIS server configuration changes are required. This mode is also ideal when layer 7 functionality such as enhanced persistence methods, URL rewriting, SSL termination etc. is required.

**Helping you Choose**

```
START

Will the VIP and the IIS Servers be located in the same subnet and within the same switch fabric?

YES → Use layer 4 DR Mode

NO

Is it possible to set the default gateway on the IIS Servers to be the load balancer?

YES → Use layer 4 NAT Mode

NO → Use layer 7 SNAT Mode
```
Initial Network Configuration

The IP address, subnet mask, default gateway and DNS settings can be configured in several ways as detailed below:

**Method 1 - Using the Network Setup Wizard at the console**

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

**Method 2 - Using the WUI:**

Using a browser, connect to the WUI on the default IP address/port: [http://192.168.2.21:9080](http://192.168.2.21:9080)

To set the IP address & subnet mask, use: Local Configuration > Network Interface Configuration

To set the default gateway, use: Local Configuration > Routing

To configure DNS settings, use: Local Configuration > Hostname & DNS

**Method 3 - Using Linux commands:**

At the console, set the initial IP address using the following command:

```bash
ip addr add <IP address>/<mask> dev eth0
```

E.g.

```bash
ip addr add 192.168.2.10/24 dev eth0
```

At the console, set the initial default gateway using the following command:

```bash
route add default gw <IP address> <interface>
```

E.g.

```bash
route add default gw 192.168.2.254 eth0
```

At the console, set the DNS server using the following command:

```bash
echo nameserver <IP address> >> /etc/resolv.conf
```

E.g.

```bash
echo nameserver 192.168.2.250 >> /etc/resolv.conf
```

**N.B. If method 3 is used, you must also configure these settings using the WUI, otherwise the settings will be lost after a reboot**
Accessing the Web User Interface (WUI)

The WUI can be accessed from a browser at:  

http://192.168.2.21:9080/lbadmin

* Note the port number → 9080

(replace 192.168.2.21 with the IP address of your load balancer if changed from the default)

Username: loadbalancer
Password: loadbalancer

Once you have entered the logon credentials the Loadbalancer.org Web User Interface will be displayed as shown below:
Clustered Pair Configuration

Loadbalancer.org recommend that load balancer appliances are deployed in pairs for high availability. In this guide a single unit is deployed first, adding a secondary slave unit is covered in the Appendix.
Load Balancing IIS using Layer 4 DR Mode

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

N.B. The steps presented in the following sections cover versions 7.6 & later of the Appliance. For older versions of the appliance please contact loadbalancer.org sales or support.

**Overview**

- **Configure the Network Interface** — A single Interface is needed, eth0 is normally used
- **Configure the Virtual Service (VIP)** — All IIS servers are accessed via this IP address
- **Configure the Real Servers (RIPs)** — Define the servers that make up the IIS cluster
- **Configure the IIS Servers** — In DR mode, the ‘ARP Problem’ must be solved on each IIS server

**Load Balancer Configuration**

**Configure the Network Interface**

- One interface is required. Pages 11 & 12 of this guide covers the various methods available to configure network settings.

**Configure the Virtual Service (VIP)**

- Using the WUI, go to *Cluster Configuration > Layer 4 – Virtual Services* and click [Add a New Virtual Service]

- Enter the following details:

<table>
<thead>
<tr>
<th>Label</th>
<th>IIS-Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Service IP Address</td>
<td>192.168.2.180</td>
</tr>
<tr>
<td>Ports</td>
<td>80</td>
</tr>
<tr>
<td>Protocol</td>
<td>TCP</td>
</tr>
<tr>
<td>Forwarding Method</td>
<td>Direct Routing</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Enter an appropriate name (Label) for the VIP, e.g. **IIS-Cluster**
- Set the **Virtual Service IP address** field to the required IP address, e.g. **192.168.2.180**
- Set the **Virtual Service Ports** field to **80**
• Leave Protocol set to TCP
• Ensure that Forwarding Method is set to Direct Routing
• Click Update
• Now click [Modify] next to the newly created Virtual Service
• Set Balance Mode (the load balancing algorithm) according to your needs
• Click Update

Configure the Real Servers (RIPs)

• Using the WUI, go to Cluster Configuration > Layer 4 – Real Servers and click [Add a New Real Server] next to the newly created Virtual Service
• Enter the following details:

<table>
<thead>
<tr>
<th>Label</th>
<th>IIS-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real Server IP Address</td>
<td>192.168.2.190</td>
</tr>
<tr>
<td>Weight</td>
<td>100</td>
</tr>
<tr>
<td>Minimum Connections</td>
<td>0</td>
</tr>
<tr>
<td>Maximum Connections</td>
<td>0</td>
</tr>
</tbody>
</table>

• Enter an appropriate name (Label) for the first IIS server, e.g. IIS1
• Change the Real Server IP Address field to the required IP address (e.g. 192.168.2.190)
• Leave other settings at their default values
• Click Update
• Repeat the above steps for your other IIS server(s)
IIS Server Configuration

Solve the ‘ARP Problem’
As mentioned previously, DR mode works by changing the MAC address of the incoming packet. Therefore the load balancer and the IIS servers must both be configured to accept traffic for the same IP address. However, only the load balancer should respond to ARP requests. To achieve this, a loopback adapter is added to the IIS servers. The IP address is then set to be the same as the Virtual Service and is also configured so that it does not respond to ARP requests. For specific configuration steps for Windows 2000/2003 and Windows 2008/2012, please refer to the appropriate full admin manual referenced at the start of this guide and search for 'ARP Problem' and follow the steps for your particular version of Windows.

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.
DR Mode – Key Points

• You must solve the ‘ARP Problem’ on all IIS servers in the cluster (please refer to the administration manual for more details)

• 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
Load Balancing IIS using Layer 4 NAT Mode

NOTE: It’s highly recommended that you have a working IIS environment first before implementing the load balancer.

N.B. The steps presented in the following sections cover versions 7.6 & later of the Appliance. For older versions of the appliance please contact loadbalancer.org sales or support.

Overview

• **Configure the Network Interfaces** – Two interfaces must be used located on different subnets. This can either be two physical interfaces such as eth0 and eth1, or one physical interface such as eth0 and an additional alias/secondary interface

• **Configure the Virtual Service (VIP)** – All IIS servers are accessed via this IP address

• **Configure the Real Servers (RIPs)** – Define the servers that make up the IIS cluster

• **Configure the IIS Servers** – In NAT mode, the IIS servers default gateway must be configured to be an IP address on the load balancer

Load Balancer Configuration

*Configure the Network Interfaces*

- Set the first IP address using one of the methods listed on pages 11 & 12 of this guide
- Using the WUI, define an additional IP address in a different subnet – either by using 2 separate interfaces or a single interface with an additional alias (secondary) address as shown below:

**Using Separate Interfaces**

<table>
<thead>
<tr>
<th>IP Address Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>eth0 192.168.2.170/24</td>
</tr>
<tr>
<td>MTU 1500 bytes</td>
</tr>
<tr>
<td>eth1 192.168.23.170/24</td>
</tr>
<tr>
<td>MTU 1500 bytes</td>
</tr>
</tbody>
</table>
**Using a Single Interfaces with Multiple IPs**

<table>
<thead>
<tr>
<th>IP Address Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.2.170/24</td>
</tr>
<tr>
<td>192.168.23.170/24</td>
</tr>
</tbody>
</table>

MTU: 1500 bytes

**Configure the Virtual Service (VIP)**

- Using the WUI, go to *Cluster Configuration > Layer 4 – Virtual Services* and click [Add a New Virtual Service]
- Enter the following details:
  - **Label**: IIS-Cluster
  - **Virtual Service IP Address**: 192.168.2.180
  - **Ports**: 80
  - **Protocol** set to TCP
  - **Forwarding Method** is to NAT
- Click **Update**
- Now click **Modify** next to the newly created Virtual Service
- Set **Balance Mode** (the load balancing algorithm) mode according to your needs
- Click **Update**
Configure the Real Servers (RIPs)

- Using the WUI, go to Cluster Configuration > Layer 4 – Real Servers and click [Add a New Real Server] next to the newly created Virtual Service
- Enter the following details:

<table>
<thead>
<tr>
<th>Label</th>
<th>IIS-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real Server IP Address</td>
<td>192.168.23.190</td>
</tr>
<tr>
<td>Real Server Port</td>
<td>80</td>
</tr>
<tr>
<td>Weight</td>
<td>100</td>
</tr>
<tr>
<td>Minimum Connections</td>
<td>0</td>
</tr>
<tr>
<td>Maximum Connections</td>
<td>0</td>
</tr>
</tbody>
</table>

- Enter an appropriate name (Label) for the first IIS server, e.g. IIS-1
- Change the Real Server IP Address field to the required IP address, e.g. 192.168.23.190
- Set the Real Server Port field to 80
- Leave other settings at their default values
- Click Update
- Repeat the above steps for your other IIS server(s)

IIS Server Configuration

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 internal IP address on eth0 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 master unit fails, the floating IP will be brought up on the slave and failover will be seamless. To create a floating IP address on the load balancer:

- Go to Cluster Configuration > Floating IP(s)
- Enter the required IP address to be used for the default gateway and click Update. Once added, there will be two floating IPs, one for the Virtual Service (192.168.2.180) and one for the default gateway (e.g. 192.168.23.254) as shown below:
### NAT Mode – Key Points

- Virtual Services & Real Servers (i.e. the IIS servers) must be on different subnets
- The default gateway on the IIS servers should be an IP address on the load balancer
- Port translation is possible, e.g. VIP:80 → RIP:8080 is allowed
Load Balancing IIS using Layer 7 SNAT (HAProxy) Mode

Overview

• **Configure the Network Interface(s)** – HAProxy can be deployed in single-arm or two-arm mode. As with layer 4 NAT mode, with a two-arm Layer 7 configuration, this can be either two physical interfaces such as eth0 and eth1, or one physical interface such as eth0 and an alias/secondary interface such as eth0:0

• **Configure the Virtual Service (VIP)** – All IIS servers are accessed via this IP address

• **Configure the Real Servers (RIPs)** – Define the IIS servers that make up the IIS cluster

• **Configure the IIS Servers** – No Real Server changes are required

Load Balancer Configuration (single-arm example)

**Configure the Network Interface**

• One interface is required. Pages 11 & 12 of this guide covers the various methods available to configure network settings.

**Configure the Virtual Service (VIP)**

• Using the WUI, go to *Cluster Configuration > Layer 7 – Virtual Services* and click [Add a New Virtual Service]

• Enter the following details:

<table>
<thead>
<tr>
<th>Label</th>
<th>IIS-Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Service IP Address</td>
<td>192.168.2.180</td>
</tr>
<tr>
<td>Ports</td>
<td>80</td>
</tr>
<tr>
<td>Layer 7 Protocol</td>
<td>HTTP Mode ▼</td>
</tr>
</tbody>
</table>

• Enter an appropriate name (Label) for the Virtual Service, e.g. **IIS-Cluster**

• Set the Virtual Service IP address field to the required IP address, e.g. **192.168.2.180**

• Set the Virtual Service Ports field to **80**

• Leave Layer 7 Protocol set to **HTTP Mode**

• Click **Update**

• Now click [Modify] next to the newly created Virtual Service

• Set **Balance mode** (the load balancing algorithm) mode according to your needs

• Click **Update**
Configure the Real Servers (RIPs)

- Using the WUI, go to Cluster Configuration > Layer 7 – Real Servers and click [Add a New Real Server] next to the newly created Virtual Service
- Enter the following details:
  - Enter an appropriate name (Label) for the first IIS server, e.g. **IIS-1**
  - Change the **Real Server IP Address** field to the required IP address (e.g. **192.168.2.190**)
  - Set the **Real Server Port** field to **80**
  - Click **Update**
  - Repeat the above steps for your other IIS server(s)

IIS Server Configuration

In SNAT (HAProxy) mode, no IIS server configuration changes are required.

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
- Not as fast as Layer 4 DR mode or NAT mode
Additional Configuration Options & Settings

SSL Termination

1 – Certificates Installed on the IIS Servers (aka SSL Pass-through)

When certificates are installed on the IIS servers:

- It's not possible to use HTTP cookie persistence since packets are encrypted and therefore the cookie cannot be read. If persistence via the load balancer is required, IP persistence must be used.
- Data is encrypted from client to server. This provides full end-to-end data encryption as shown in the diagram below:

![Diagram of SSL Termination]

Creating a CSR (Steps shown are for Windows 2008 R2)

To generate a certificate for IIS the first step is to create a Certificate Signing Request (CSR):

1. Select the IIS server in IIS Manager then double-click Server Certificates
2. In the actions section on the right hand side of the screen, select *Create Certificate Request*, fill in the relevant details as per the example below, then click **Next**

![Certificate Request Form](image1)

3. Leave the default settings and click **Next**

![Cryptographic Service Provider Properties](image2)

4. Where prompted on the following screen enter a suitable filename, e.g. `c:\csr.txt` and click **Finish**

5. Use this saved CSR with your chosen Certificate Authority to obtain your certificate

6. Once you've received your certificate from the CA, save it as a text file
7. To install the certificate on the IIS server select *Complete Certificate Request* in the action section of Server Certificates in IIS Manager, then specify the filename and a friendly name and click **OK**.

8. At this point depending on your specific version of Windows, you may receive the message shown below. This is a known issue that occurs because the friendly certificate name entered in step 7 above in not being read correctly.  

For more details please refer to [http://support.microsoft.com/kb/959216](http://support.microsoft.com/kb/959216). Note that the certificate has been installed and can be seen in IIS Manager under Server Certificates.

9. Now amend the site bindings to include HTTPS and the newly installed certificate.
2 – Certificates Installed on the Load balancer (aka SSL off-loading)

- When certificates are installed on the load balancer:

  • It’s possible to use HTTP cookie based persistence
  • Since SSL is terminated on the load balancer, data from the load balancer to the IIS servers is not encrypted as shown in the diagram below. This may or may not be an issue depending on the network structure between the load balancer and IIS servers and your security requirements

  N.B. From v7.6 re-encryption is possible between the load balancer and the Real Servers (aka. SSL bridging). To use this, enable the ‘Re-encrypt to Backend’ option for each RIP and click Update. Each server must be correctly configured for HTTPS for this to work and an appropriate certificate must also be installed (this can be a self signed server certificate). See the administration manual for more details.

  • A Pound or STunnel SSL VIP can be used to terminate SSL (N.B. STunnel in only available in v7.5 and later). The backend for this VIP can be either a Layer 4 NAT mode Virtual Service or a Layer 7 HAProxy Virtual Service. The following diagram shows this:

  ![Diagram showing SSL termination]

  **N.B. It’s not possible to use a layer 4 DR mode Virtual Service in this scenario**

---

**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.

---

Exchanging Certificates from Windows / Importing to the load balancer (Steps shown for Windows 2008 R2)

It’s often easiest to get the certificate working on the IIS server first, then export the certificate and import this to the load balancer. The steps for Windows 2008 R2 for this process are as follows:
1. Once the certificate is working correctly on your Windows server, run `mmc`, and add the certificates snap-in. Expand the Personal folder and click on `Certificates` – your certificate should be here. Right-click the certificate and select **All Tasks > Export**. This will start the Certificate Export Wizard as shown below:

![Certificate Export Wizard](image)

2. Click **Yes** to export the private key and click **Next**

![Certificate Export Wizard](image)
3. Check *Include all Certificates in the certification path if possible* and click **Next**

![Certificate Export Wizard](image)

4. Enter a password to secure the private key and click **Next**

![Certificate Export Wizard](image)
5. Enter a folder & filename for the exported certificate and click **Next**

![Certificate Export Wizard](image)

6. Now click Finish to complete the wizard, the following confirmation should be shown:

![Certificate Export Wizard](image)

7. Now create either a Pound or STunnel (v7.5 and later) VIP on the load balancer. For the VIP you can use the same IP address as your HAProxy or NAT VIP created earlier with port 443 for HTTPS. The IP address & port for the Backend Cluster should be set the same as your HAProxy or NAT mode VIP as detailed below:

   V7.5 and later supports STunnel for SSL termination. This is the default, but Pound can also be selected if required using the **SSL Terminator** radio button.

   - Using the WUI, go to **Cluster Configuration > SSL Termination** and click [Add a New Virtual Service]
   - Enter the following details (shows STunnel example which is the default terminator from v7.5):
• Enter an appropriate name (Label) for the VIP, e.g. IIS-SSL
• Change the Virtual Service IP address field to the required value, e.g. 192.168.2.180
• Change the Virtual Service Port field to the required value, e.g. 443
• Change the Backend Virtual Service IP Address field to the required value, e.g. 192.168.2.180
• Change the Backend Virtual Service Port field to the required value, e.g. 80
• The remaining options can be left at their default values
• Click Update

8. Now upload the PFX format certificate to the newly created SSL VIP:

• Using the WUI, go to Cluster Configuration > SSL Termination and click [Certificate] next to the relevant VIP
• Using the browse option, select the .pfx file created earlier
• Click Upload PEM/PFX file
• A message will be displayed confirming the upload
• The Certificate State field will change to Signed Certificate Installed as shown below:

<table>
<thead>
<tr>
<th>SSL1</th>
<th>192.168.2.100:443</th>
<th>192.168.2.100:30</th>
</tr>
</thead>
<tbody>
<tr>
<td>Certificate State:</td>
<td>Signed certificate installed</td>
<td>[Delete]</td>
</tr>
</tbody>
</table>

9. Now restart STunnel or Pound:

• Go to: Maintenance > Restart Services and click Restart STunnel or Restart Pound

Once restarted your secure website should be accessible at: https://<Virtual IP Address>
Grouping Multiple Ports on a Single Virtual Service (VIP)

In certain circumstances it may be desirable to combine multiple ports in a single Virtual Service. For example, if your IIS server has both HTTP and HTTPS content, you may want clients to connect to the same IIS server for both. This is especially useful if you need persistence as clients move from HTTP to HTTPS, e.g. an e-commerce web site without a proper back end database for session state.

Layer 4 – Using Firewall Marks

The concept is to create a firewall rule that matches incoming packets to a particular IP and port, and mark them with an arbitrary integer. A Virtual Service is then configured or modified, specifying the firewall mark instead of an IP and port.

Firewall Marks are configured automatically when multiple ports are defined for a Layer 4 VIP. For example, to configure an HTTP/HTTPS Virtual Service, simply specify port 80 & 443 separated by a comma in the ‘Virtual Service Ports’ field as shown below:

```
Label: IIS-Cluster
Virtual Service IP Address: 192.168.2.180
Ports: 80,443
Protocol: TCP
Forwarding Method: Direct Routing
```

N.B. Firewall marks can also be configured manually if needed, for details on this please refer to the administration manual.

Layer 7 – By Defining Multiple Ports

For Layer 7 VIPs, multiple ports can be defined. Simply specify all the required ports in the Virtual Service Ports field separated by commas as shown below:

```
Label: IIS-Cluster
Virtual Service IP Address: 192.168.2.180
Ports: 80,443
Layer 7 Protocol: TCP Mode
```

N.B. Make sure you leave the Real Server port field blank if using multiple ports at the VIP level.
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

At layer 4, IIS server health checking is provided by Ildirectord. This allows a full range of options to check that the IIS servers are operational, and if not what steps to take. The default check-type for new VIPs is a TCP connect to the port defined for the VIP.

For full details on the options available, please refer to the administration manual referenced at the start of this guide and search for the section `Health Monitoring`.

Layer 7

By default layer 7 (HAProxy) VIPs also use a connect type health check on the same port specified in the Virtual Service.

For full details on the options available, please refer to the administration manual referenced at the start of this guide and search for the section `Health Monitoring`. 
Using Server Feedback Agents

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.

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 \((92/100*\text{requested_weight})\) to find the new optimized weight. Using this method an idle Real Server will get 10 times as many new connections as an overloaded server.

*N.B. The 'Requested Weight' is the weight set in the WUI for each Real Server.*

For more information please also refer to the following blog article:

http://blog.loadbalancer.org/open-source-windows-service-for-reporting-server-load-back-to-haproxy-load-balancer-feedback-agent/

Windows Agent

The latest Windows feedback agent can be downloaded from:

http://downloads.loadbalancer.org/agent/loadbalanceragent.msi

To install the agent, run loadbalanceragent.msi on each server

Click **Next**

*N.B. The agent should be installed on all Real Serves in the cluster*
Select the installation folder and click **Next**

Click **Next** to start the installation

_N.B. .NET Framework v3.5 is required by the agent and .NET Framework v4.0 is required by the Monitor_

**Starting the Agent**

Once the installation has completed, you'll need to start the service on the Real Servers. The service is controlled by the Feedback Agent Monitor program that is also installed along with the Agent. The monitor can be accessed on the Windows server using: **All Programs > Loadbalancer.org > Monitor.** It's also possible to start the service using the services snap-in – the service is called 'Loadbalancer CPU monitor'.
To start the service, click **Start**
To stop the service, click **Stop**

*N.B. The agent should be installed on all IIS servers in the cluster*

**To Configure the Virtual Service to use the Agent**

As mentioned, from v7.6 both layer 4 and layer 7 VIPs are supported. To Configure Virtual Services to use Agent / HTTP Feedback follow the steps below:

- Go to **Cluster Configuration > Layer 4 - Virtual Services or Layer 7 - Virtual Services**
- Click [**Modify**] next to the Virtual Service

  ![Feedback Agent Window](image)

  - Change the Feedback Method to either **Agent** or **HTTP** for layer 4 VIPs
  - Change the Feedback Method to either **Agent** for layer 7 VIPs
  - Click **Update**
Load Balancer Transparency

Layer 4 – DR & NAT Mode
By default both Layer 4 modes are transparent. This means that IIS will log the actual IP address of the client rather than the IP address of the load balancer.

Layer 7 – TProxy
When using HAProxy, the load balancer is not transparent by default. This means that the IP address of the load balancer will be captured and stored in the IIS logs. To get around this, TProxy can be enabled. TProxy enables the IIS servers behind a layer 7 HAProxy configuration to see the client source IP address. For this to work, the load balancer must be in a NAT configuration (i.e. both internal and external subnets) and the IIS servers must be configured to use the load balancer as their default gateway.

To Enable Tproxy:

• Using the WUI, go to Cluster Configuration > Layer 7 – Advanced Configuration
• Change Transparent Proxy to On
• Click Update

Layer 7 – X-Forwarded-For Headers
Since the load balancer must be in a NAT configuration (i.e. the Virtual Service and the IIS servers in different subnets) to utilize TProxy, it’s not always an appropriate solution. In situations such as this, it’s possible to use the X-Forwarded-For header that is included by default in all layer 7 VIPs.

To enable IIS to support XFF headers, it does depend on the version on Windows being used. For IIS7 and later, IIS Advanced Logging can be installed and used. For IIS6, a 3rd party application must be installed. Several options are available – some free and some that must be paid for. One free solution that works very well is F5’s X-Forwarded-For ISAPI Filter.

Both options are covered in the following blog: http://blog.loadbalancer.org/iis-and-x-forwarded-for-header/
Testing & Validation

Testing Load Balanced Services

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://192.168.110.10

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 WUI. For a single appliance, the status bar should report **Master & Active** as shown below:

   ![Master Status](image)

2. **Check that the VIP/floating IP is up** - Using View Configuration > Network Configuration verify that the VIP is active on the load balancer, if not check Logs > Heartbeat for errors.

   ```
   2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qdisc 1000
       link/ether 00:6c:29:cf:18:03 brd ff:ff:ff:ff:ff:ff
       inet 192.168.110.85/18 brd 192.168.127.255 scope global eth0
           valid_lft forever preferred_lft_forever
       inet 192.168.110.90/18 brd 192.168.127.255 scope global secondary eth0
           valid_lft forever preferred_lft_forever
   ```

   The above example shows that the interface address (192.168.110.85) and the VIP address (192.168.110.90) are both up.

3. **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)
4. **Check the connection state**

For Layer 4 DR mode VIPs check Reports > Layer 4 Current Connections to view the current traffic in detail. Any packets with state `SYN_RECV` imply that the 'ARP Problem' has not been correctly solved on the Real Servers. See page 43 for more details on solving the ARP problem.

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 imply that the default gateway on the Real Servers has not been set to be an IP address on the load balancer.

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

- **username**: loadbalancer
- **password**: loadbalancer

This will open a second tab in the browser and display a statistics/status report as shown in the example below:

![Statistics Report for pid 3261](image)

**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. Remove the network cable from one of the web servers or stop the web service/process, 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. Replace the network cable, 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** will also show the updated 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.

**NOTE**: From v7.6.4 the System Overview supports sorting of VIPs. This can be done by clicking on the column headings or by drag & drop. For more details please refer to the administration manual.

### 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 WUI. Details of both can be found in the administration manual.

### Technical Support

For more details or assistance with your deployment please don’t hesitate to contact the support team:

support@loadbalancer.org

### Conclusion

Loadbalancer.org appliances provide a very cost effective and flexible solution for highly available load balanced Microsoft IIS environments.
Appendix

1 - Solving the ARP Problem - Windows 2012

This process enables each server to be able to receive traffic destined for the VIP. It also also ensures that each server does not respond to ARP requests for the VIP address – only the load balancer should do this.

Step 1: Install the Microsoft Loopback Adapter

1. Click Start, then run hdwwiz to start the Hardware Installation Wizard
2. When 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: 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
4. Un-check all items except Internet Protocol Version 4 (TCP/IPv4) and Internet Protocol Version
6 (TCP/IPv6) as shown below

N.B. leaving both checked ensures that both IPv4 and IPv6 are supported. If preferred, only the protocol to be used can be checked

5. If configuring IPv4 addresses select Internet Protocol Version 4 (TCP/IPv4), click Properties and configure the IP address to be the same as the Virtual Service (VIP) with a subnet mask of 255.255.255.255, e.g. 192.168.2.20 / 255.255.255.255 as shown below

6. If configuring IPv6 addresses select Internet Protocol Version 6 (TCP/IPv6), 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
7. Click **OK** on TCP/IP Properties, then click **Close** on Ethernet Properties to save and apply the new settings.

8. Now repeat the above process on the other Windows 2012 Real Servers

*N.B. For Windows 2012, it's not necessary to modify the interface metric on the advanced tab and should be left set to Automatic*

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

Windows Server 2000 and Windows Server 2003 use the weak host model for sending and receiving for all IPv4 interfaces and the strong host model for sending and receiving for all IPv6 interfaces. You cannot configure this behavior.

The Next Generation TCP/IP stack in Windows 2008 and later supports strong host sends and receives for both IPv4 and IPv6 by default. To ensure that Windows 2012 is running in the correct mode to be able to respond to the VIP, the following commands must be run on each Real Server:

**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 these commands to work, the LAN connection NIC must be named “net” and the loopback NIC must be named “loopback” as shown below. If you prefer to leave your current NIC names, then the commands above must be modified accordingly. For example, if your network adapters are named “LAN” and “LOOPBACK”, the commands required would be:

```
netsh interface ipv4 set interface "LAN" 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
```

For these commands to work, the LAN connection NIC must be named “net” and the loopback NIC must be named “loopback” as shown below. If you prefer to leave your current NIC names, then the commands above must be modified accordingly. For example, if your network adapters are named “LAN” and “LOOPBACK”, the commands required would be:

```
netsh interface ipv6 set interface "LAN" 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
```

N.B. The names for the NICs are case sensitive, so make sure that the name used for the interface and the name used in the commands match exactly.

1. Start Powershell or use a command window to run the appropriate netsh commands as shown in the example below

```
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
```

N.B. This shows an IPv6 example, use the IPv4 commands if you're using IPv4 addresses.

2. Now repeat these 4 commands on the other Windows 2012 Real Servers

N.B. Solving the ARP problem for other version of Windows is similar. For full details, please refer to the administration manual.
2 – Clustered Pair Configuration – Adding a Slave Unit

If you initially configured just the master unit and now need to add a slave - our recommended procedure, please refer to the relevant document referenced below for more details:

**Version 7**

Please refer to Chapter 8 – Appliance Clustering for HA in the v7 Administration Manual.

**Version 8**

Please refer to Chapter 9 – Appliance Clustering for HA in the v8 Administration Manual.

Don’t hesitate to contact our support team if you need further assistance: support@loadbalancer.org
# 3 – Company Contact Information

<table>
<thead>
<tr>
<th>Website</th>
<th>URL : <a href="http://www.loadbalancer.org">www.loadbalancer.org</a></th>
</tr>
</thead>
</table>
| **North America (US)** | Loadbalancer.org, Inc.  
270 Presidential Drive  
Wilmington,  
DE 19807  
USA  
Tel : +1 888.867.9504 (24x7)  
Fax : +1 302.213.0122  
Email (sales) : sales@loadbalancer.org  
Email (support) : support@loadbalancer.org |
| **North America (Canada)** | Loadbalancer.org Ltd.  
300–422 Richards Street  
Vancouver, BC  
V6B 2Z4  
Canada  
Tel : +1 855.681.6017 (24x7)  
Fax : +1 302.213.0122  
Email (sales) : sales@loadbalancer.org  
Email (support) : support@loadbalancer.org |
| **Europe (UK)** | Loadbalancer.org Ltd.  
Portsmouth Technopole  
Kingston Crescent  
Portsmouth  
PO2 8FA  
England, UK  
Tel : +44 (0)330 3801064 (24x7)  
Fax : +44 (0)870 4327672  
Email (sales) : sales@loadbalancer.org  
Email (support) : support@loadbalancer.org |
| **Europe (Germany)** | Loadbalancer.org GmbH  
Alt Pempelfort 2  
40211 Düsseldorf  
Germany  
Tel : +49 (0)30 920 383 6494  
Fax : +49 (0)30 920 383 6495  
Email (sales) : vertrieb@loadbalancer.org  
Email (support) : support@loadbalancer.org |