Load Balancing Microsoft IIS

Version 1.8.0
# Table of Contents

1. About this Guide ................................................................. 4  
2. Loadbalancer.org Appliances Supported .................................. 4  
3. Loadbalancer.org Software Versions Supported .......................... 4  
4. Microsoft IIS Software Versions Supported............................... 4  
5. Microsoft Internet Information Services (IIS) ............................. 4  
6. Load Balancing IIS ................................................................. 4  
   The Basics ........................................................................... 4  
   Ports & Protocols .................................................................. 4  
   IIS Server Health-checks ....................................................... 5  
   SSL Termination & Certificates .............................................. 5  
   Persistence (aka Server Affinity) .......................................... 5  
   Load Balancer Deployment .................................................. 5  
      WAF .............................................................................. 6  
   Load Balancer Deployment Modes ........................................ 6  
      Layer 4 DR Mode ............................................................. 6  
      Layer 4 NAT Mode .......................................................... 7  
      Layer 7 SNAT Mode .......................................................... 9  
   Loadbalancer.org Recommended Mode ................................... 10  
7. Loadbalancer.org Appliance – the Basics .................................... 11  
   Virtual Appliance .................................................................. 11  
   Initial Network Configuration ............................................. 12  
   Accessing the WebUI .......................................................... 12  
      Main Menu Options ....................................................... 13  
   HA Clustered Pair Configuration ......................................... 14  
8. Appliance & IIS Server Configuration – Using Layer 4 DR Mode ... 14  
   Overview ........................................................................... 14  
   Load Balancer Configuration .............................................. 14  
      Configure the Network Interface .................................... 14  
      Configure the Virtual Service (VIP) ................................. 14  
      Configure the Real Servers (RIPs) ..................................... 15  
   IIS Server Configuration ..................................................... 16  
      Solve the ‘ARP Problem’ ................................................. 16  
      Configure IIS Bindings ................................................. 16  
   DR Mode – Key Points ....................................................... 17  
9. Appliance & IIS Server Configuration – Using Layer 4 NAT Mode 17  
   Overview ........................................................................... 17  
   Load Balancer Configuration .............................................. 17  
      Configure the Network Interfaces .................................... 17  
      Configure the Virtual Service (VIP) ................................. 18  
      Configure the Real Servers (RIPs) ..................................... 19  
      Create a Floating IP to use for the IIS server’s Default Gateway 20  
   IIS Server Configuration ..................................................... 21  
      Default Gateway ........................................................... 21  
      NAT Mode – Key Points ................................................. 21  
10. Appliance & IIS Server Configuration – Using Layer 7 SNAT Mode 21  
   Overview ........................................................................... 21  
   Load Balancer Configuration .............................................. 21  
      Configure the Network Interface .................................... 21  
      Configure the Virtual Service (VIP) ................................. 21
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>11. Additional Configuration Options &amp; Settings</td>
<td>23</td>
</tr>
<tr>
<td>SSL Termination</td>
<td>23</td>
</tr>
<tr>
<td>SSL Termination on the IIS servers (SSL Pass-through)</td>
<td>23</td>
</tr>
<tr>
<td>SSL Termination on the Load Balancer (SSL Offloading)</td>
<td>24</td>
</tr>
<tr>
<td>SSL Termination on the Load Balancer with Re-encryption (SSL Bridging)</td>
<td>28</td>
</tr>
<tr>
<td>Real Server (IIS) Health Checks</td>
<td>29</td>
</tr>
<tr>
<td>Layer 4</td>
<td>29</td>
</tr>
<tr>
<td>Layer 7</td>
<td>30</td>
</tr>
<tr>
<td>External Health-Check Scripts</td>
<td>30</td>
</tr>
<tr>
<td>URL Rewriting / Content Switching (ACL’s)</td>
<td>30</td>
</tr>
<tr>
<td>HTTP Header Manipulation</td>
<td>33</td>
</tr>
<tr>
<td>Web Application Firewall (WAF)</td>
<td>33</td>
</tr>
<tr>
<td>Server feedback Agent</td>
<td>35</td>
</tr>
<tr>
<td>Windows Agent</td>
<td>35</td>
</tr>
<tr>
<td>Linux/Unix Agent</td>
<td>37</td>
</tr>
<tr>
<td>Custom HTTP Agent</td>
<td>38</td>
</tr>
<tr>
<td>Load Balancer Transparency</td>
<td>38</td>
</tr>
<tr>
<td>Layer 4</td>
<td>38</td>
</tr>
<tr>
<td>Layer 7</td>
<td>38</td>
</tr>
<tr>
<td>12. Testing &amp; Verification</td>
<td>39</td>
</tr>
<tr>
<td>Testing Load Balanced Services</td>
<td>39</td>
</tr>
<tr>
<td>Diagnosing VIP Connection Problems</td>
<td>39</td>
</tr>
<tr>
<td>Taking IIS Servers Offline</td>
<td>40</td>
</tr>
<tr>
<td>Using Reports &amp; Log Files</td>
<td>41</td>
</tr>
<tr>
<td>13. Technical Support</td>
<td>41</td>
</tr>
<tr>
<td>14. Further Documentation</td>
<td>41</td>
</tr>
<tr>
<td>15. Conclusion</td>
<td>41</td>
</tr>
<tr>
<td>16. Appendix</td>
<td>42</td>
</tr>
<tr>
<td>Solving the ARP Problem</td>
<td>42</td>
</tr>
<tr>
<td>Windows Server 2012, 2016 &amp; 2019</td>
<td>42</td>
</tr>
<tr>
<td>Configuring HA - Adding a Secondary Appliance</td>
<td>46</td>
</tr>
<tr>
<td>17. Document Revision History</td>
<td>50</td>
</tr>
</tbody>
</table>
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. Loadbalancer.org Software Versions Supported

* V8.3.8 and later

4. Microsoft IIS Software Versions Supported

* Microsoft IIS – all versions

5. 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.

6. Load Balancing IIS

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

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 straightforward and can be achieved by simply adding additional IIS servers to the load balanced cluster.

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>Port</td>
<td>Protocol</td>
<td>Use</td>
</tr>
<tr>
<td>------</td>
<td>-----------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>443</td>
<td>TCP/HTTPS</td>
<td>HTTPS web traffic</td>
</tr>
</tbody>
</table>

**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.

**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.

**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.

**Load Balancer Deployment**

The following diagram illustrates how the load balancer is deployed with multiple IIS servers.
WAF = Web Application Firewall

VIP = Virtual 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.

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 N-Path.
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 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 WebUI option: Cluster Configuration > Layer 4 - Advanced Configuration, the external interface should be selected.

NAT mode can be deployed in the following ways:

Two-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.

Two-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.

One-arm (using 1 Interface), 1 subnet - Here, the VIP is brought up in the same subnet as the Real Servers. For clients located in remote networks the default gateway on the Real Servers must be set to be an IP address on the load balancer. For clients located on the same 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 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. 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.
- 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 Server 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:

<table>
<thead>
<tr>
<th>Protocol</th>
<th>VIP</th>
<th>Port</th>
<th>RIP</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP</td>
<td>10.0.0.20</td>
<td>80</td>
<td>192.168.1.50</td>
<td>80</td>
</tr>
</tbody>
</table>

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.
Layer 7 SNAT mode is a full proxy and therefore load balanced Real Servers do not need to be changed in any way.

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.

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.
7. Loadbalancer.org Appliance – the Basics

Virtual Appliance

A fully featured, fully supported 30 day trial is available if you are conducting a PoC (Proof of Concept) deployment. The VA is currently available for VMware, Virtual Box, Hyper-V, KVM, XEN and Nutanix AHV and has been optimized for each Hypervisor. By default, the VA is allocated 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 The Virtual Appliance - Hypervisor Deployment and the ReadMe.txt text file included in the VA download for more detailed information on deploying the VA using various Hypervisors.
For the VA, 4 NICs are included but only eth0 is connected by default at power up. If the other NICs are required, these should be connected using the network configuration screen within the Hypervisor.

**Initial Network Configuration**

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

**Important**

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

**Accessing the WebUI**

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

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

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

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


2. Log in to the WebUI:

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

   **Note**

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

Once logged in, the WebUI will be displayed as shown below:
The WebUI for the VA is shown, the hardware and cloud appliances are very similar. The yellow licensing related message is platform & model dependent.

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

**Main Menu Options**
- **System Overview** - Displays a graphical summary of all VIPs, RIPv2 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 & RIPv2
- **Maintenance** - Perform maintenance tasks such as service restarts and taking backups
- **View Configuration** - Display the saved appliance configuration settings
- **Reports** - View various appliance reports & graphs
HA Clustered Pair Configuration
Loadbalancer.org recommend that load balancer appliances are deployed in pairs for high availability. In this guide a single unit is deployed first, adding a secondary unit is covered in Configuring HA - Adding a Secondary Appliance.

8. 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.

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.

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:

   - 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 80,443.


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.

   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).
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. Please refer to Solving the ARP Problem for full details of solving the ARP problem for Windows 2012 & later.

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 (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**

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

**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.

**Load Balancer Configuration**

**Configure the Network Interfaces**

1. Configure the first interface by following the steps in Initial Network Configuration.
2. Using the WebUI, navigate to: Local Configuration > Network Interface Configuration.

3. Define an additional IP address in a different subnet – either by using 2 separate interfaces:

   ![IP Address Assignment Diagram]

   Eth0 is typically used as the internal interface and eth1 is used as the external interface. However, you can use any interface for any purpose.

   Or by adding a VLAN:

   ![IP Address Assignment Diagram]

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.

<table>
<thead>
<tr>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>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).</td>
</tr>
<tr>
<td>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.</td>
</tr>
</tbody>
</table>

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.

<table>
<thead>
<tr>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>For more information about persistence, please refer to Persistence (aka Server Affinity).</td>
</tr>
</tbody>
</table>

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:
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
- 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 (for an HA pair this must be a floating IP address)
- Port translation is possible, e.g. VIP:80 > RIP:8080 is allowed

10. 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.

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.

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.
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.

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.

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).

**IIS Server Configuration**

In layer 7 SNAT 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
- 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

**11. Additional Configuration Options & Settings**

**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.
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)

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.

By default, a self-signed certificate is used for the new SSL VIP. Certificates can be requested on the load balancer or uploaded as described in the section below. The default self-signed certificate can be regenerated if needed using the WebUI menu option: SSL Certificate and clicking the Regenerate Local SSL Certificate button.

The backend for the SSL VIP can be either a Layer 7 SNAT mode VIP or a Layer 4 NAT or SNAT mode VIP. Layer 4 DR mode cannot be used since 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. Alternatively, you can create a Certificate Signing Request (CSR) and send this to your chosen CA to create a new certificate.

**Generating a CSR on the Load Balancer**

CSR’s can be generated on the load balancer to apply for a certificate from your CA.

To generate a CSR:

1. Using the WebUI, navigate to: *Cluster Configuration > SSL Certificates*.
2. Click *Add a new SSL Certificate* and select *Create a New SSL Certificate (CSR)*.

![CSR Form]

3. Enter a suitable label (name) for the certificate, e.g. **Cert1**.
4. Populate the remaining fields according to your requirements.
5. Once all fields are complete click **Create CSR**.
6. To view the CSR click **Modify** next to the new certificate, then expand the Certificate Signing Request (CSR) section.
7. Copy the CSR and send this to your chosen CA.
8. Once received, copy/paste your signed certificate into the *Your Certificate* section.
9. Intermediate and root certificates can be copied/pasted into the *Intermediate Certificate* and *Root Certificate* sections as required.
10. Click **Update** to complete the process.

**Uploading Certificates**

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

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

2. Click Add a new SSL Certificate 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:

```
Information: cert1 SSL Certificate uploaded successfully.
```

**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**

Using a text editor such as vi or vim under Linux or Notepad under Windows, create an empty file (e.g. pem.txt) then copy/paste the entire contents of each of the following items into this file in the order listed:

* Private Key
* SSL Certificate
* Intermediate Certificate
* Root CA Certificate

Make sure you include the beginning and end tags. The resulting file should look like this:
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.

   ![Virtual Service Table]

<table>
<thead>
<tr>
<th>Label</th>
<th>SSL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Associated Virtual Service</td>
<td>IIS-Cluster</td>
</tr>
<tr>
<td>Virtual Service Port</td>
<td>443</td>
</tr>
<tr>
<td>SSL Operation Mode</td>
<td>High Security</td>
</tr>
<tr>
<td>SSL Certificate</td>
<td>Cert1</td>
</tr>
</tbody>
</table>

2. Enter a suitable Label (name) for the VIP, e.g. SSL.

3. Set Associated Virtual Service to the appropriate VIP, e.g. IIS-Cluster.

   **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.

4. Leave Virtual Service Port set to 443.


6. Select the required certificate from the SSL Certificate drop-down.

7. Click Update.
8. Reload STunnel to apply the new settings using the link provided in the blue box.

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

**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.

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.

**Note**
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*.

   ![Image of WebUI menu](image)

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).

**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:

<table>
<thead>
<tr>
<th>Check Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Negotiate</td>
<td>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.</td>
</tr>
<tr>
<td>Connect to port</td>
<td>Just do a simple connect to the specified port/service &amp; verify that it’s able to accept a connection.</td>
</tr>
<tr>
<td>Ping server</td>
<td>Sends an ICMP echo request packet to the Real Server.</td>
</tr>
<tr>
<td>External check</td>
<td>Use a custom script for the health check.</td>
</tr>
<tr>
<td>No checks, always Off</td>
<td>All Real Servers are off.</td>
</tr>
<tr>
<td>No checks, always On</td>
<td>All Real Servers are on (no checking).</td>
</tr>
<tr>
<td>5 Connects, 1 Negotiate</td>
<td>Do 5 connect checks and then 1 negotiate check.</td>
</tr>
<tr>
<td>Check Type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>10 Connects, 1 Negotiate</td>
<td>Do 10 connect checks and then 1 negotiate check.</td>
</tr>
</tbody>
</table>

**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:

<table>
<thead>
<tr>
<th>Check Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Negotiate HTTP/HTTPS (GET)</td>
<td>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.</td>
</tr>
<tr>
<td>Negotiate HTTP/HTTPS (HEAD)</td>
<td>Scan the returned page headers defined as the Request to Send, and check the returned data in the Response Expected string. HEAD would return the page headers that would usually be returned by a GET.</td>
</tr>
<tr>
<td>Connect to port</td>
<td>Just do a simple TCP connect to the specified port/service &amp; verify that it’s able to accept a connection.</td>
</tr>
<tr>
<td>External Script</td>
<td>Use a custom script for the health check.</td>
</tr>
<tr>
<td>MySQL</td>
<td>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</td>
</tr>
<tr>
<td>No checks, always On</td>
<td>All Real Servers are assumed on (i.e. no checking)</td>
</tr>
</tbody>
</table>

---

**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 provides an example script that performs an HTTP GET, checks that the Application Pool specified is running and writes the status to a text file on the IIS server which the load balancer then reads.

**URL Rewriting / Content Switching (ACL’s)**

The WebUI supports the ability to create ACL’s which can be used to control and direct HTTP traffic based on the rules defined. This option can be accessed by clicking the Edit ACL Rules button when modifying a VIP.
Multiple rules can be defined using the Add button.

Once all rules have been defined, click Save to save the rules, then 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, requests are redirected to the URL location http://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:

http://www.example.com

Other Examples:

In the example above, requests are redirected to the URL prefix http://www.domain3.com if the host header value is www.domain1.com.

e.g. if the requested URL is:

http://www.domain1.com/contract

the request is redirected to:
In the example above, requests are forwarded to the backend called **Blog** if the path begins with `/blog`

e.g. if the requested URL is:

http://www.domain1.com/blog

the request is forwarded to the backend called "Blog".

Requests to http://www.domain1.com/<other locations> are forwarded to the IIS servers that were defined using the WebUI menu option: *Cluster Configuration > Layer 7 – Real Servers*.

The Backend can be defined in the following 2 ways:

1 – As a Manually defined Backend

using the WebUI menu option: *Cluster Configuration > Layer 7 – Manual Configuration*, the backend 'Blog' can be defined as shown below:

```
backend Blog
mode http
balance roundrobin
option forwardfor
server rip3 192.168.110.242:80 weight 1 check
server rip4 192.168.110.243:80 weight 1 check
```

2 – As a VIP with the required backend (Real) Servers

Here, 'Blog' has been defined as an additional VIP with 2 Real Servers:
When defining ACL’s that have their Rule Action Type set to **Backend** or **Use Server**, the relevant Backend, VIP or Real Server must exist before HAProxy can be successfully restarted. Note also that names used are case sensitive.

### HTTP Header Manipulation

The appliance enables HTTP headers to be added, set and deleted as described below. This option can be accessed by clicking the **Edit HTTP Headers** button when modifying a VIP.

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>Allows you to append a HTTP header who’s name is controlled by the 'Header Name' input box. The value of the header is controlled by the 'Header Value' Box.</td>
</tr>
<tr>
<td>set</td>
<td>Does the same as add but the header is removed/replaced if it already exists.</td>
</tr>
<tr>
<td>Delete</td>
<td>Removes all HTTP header fields that match the header name specified in the Header Name Box.</td>
</tr>
</tbody>
</table>

Multiple headers can be defined using the **Add** button.

- Once all headers have been defined, click **Save** to save the headers, then 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:

- **[HTTP_X_CLIENTDEST_PORT]**, i.e. the port that the client connected to
- **[HTTP_X_CLIENTDEST]**, i.e. the IP address that the client connected to
- **[HTTP_X_SOURCE]**, i.e. the client’s source IP address.

### 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:

![Data flow before WAF](image)

Modified data flow once WAF is deployed:

![Modified data flow](image)

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 Chapter 7 – Web Application Firewall (WAF) in the Administration Manual.
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.

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.

Note: The agent should be installed on all Real Servers in the cluster.

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 & control program that is also installed along with the Agent. This can be accessed on the Windows server from: Start> Loadbalancer.org > Loadbalancer.org Feedback Agent. It's also possible to start the service using the services snap-in - the service is called LBCPUMon.

- To start the service, click the Start button
- To stop the service, click the Stop button

Linux/Unix Agent

The Linux feedback agent files can be downloaded using the following links:
**Installation & Testing**

Install xinetd:

```
apt-get install xinetd (if not already installed)
```

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:

```
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 all Real Servers, **not** the load balancer.

**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 applications requirements.

**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.

12. Testing & Verification

   Note For additional general guidance please also refer to Testing Load Balanced Services.

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 WebUI. For a single appliance, the status bar should report Primary and Active as shown below:

   Master | Slave    | Active | Passive | Link

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 qlen 1000
      link/ether 09:0c: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 IIS 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 IIS Servers). Green indicates a healthy cluster, yellow indicates that your cluster may need attention (one or more of the IIS Servers may be down), and blue indicates all IIS 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 IIS Servers. Refer to **Solving the ARP Problem** 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 IIS Servers has not been set to be an IP address on the load balancer.

   For Layer 7 VIPs, the Layer 7 statistics page can be used. To access the page, navigate to: **Reports > Layer 7 Status** - a new tabbed window will be displayed:

![Statistics Report for pid 3281](image)

**Taking IIS Servers Offline**

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

2. Remove the network cable from one of the IIS 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:

**RIP1** is green. This indicates that it’s operating normally.

**RIP2** is blue. This indicates that it has been either Halted or Drained. In this example, Halt has been used as indicated by Online (Halt) being displayed. If it had been drained it would show as Online (Drain).

**RIP3** is red. This indicates that it has failed a health check.

Using Reports & Log Files
The appliance includes several logs and reports that are very useful when diagnosing issues. Both are available as main menu options in the WebUI. Details of both can be found in chapter 13 in the Administration Manual.

13. 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.

14. Further Documentation

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

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, 2016 & 2019

The basic concept is the same as for Windows 2000/2003. However, additional steps are required to set the strong/weak host behavior. This is used to either block or allow interfaces receiving packets destined for a different interface on the same server. As with Windows 2000/2003/2008, if the Real Server is included in multiple VIPs, you can add additional IP addresses to the Loopback Adapter that correspond to each VIP.

Step 1 of 3: 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 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**.

4. Uncheck all items except **Internet Protocol Version 4 (TCP/IPv4)** and **Internet Protocol Version 6 (TCP/IPv6)** as shown below:

   ![Loopback Properties](image)

   **Note** Leaving both checked ensures that both IPv4 and IPv6 are supported. Select one if preferred.

5. If configuring IPv4 addresses select **Internet Protocol Version (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 (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.

Note: For Windows 2012/2016/2019, it’s not necessary to modify the interface metric on the advanced tab and should be left set to Automatic.

Step 3 of 3: Configure the strong/weak host behavior
To configure the correct strong/weak host behavior for Windows 2012/2016/2019, 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
```

Note: 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.

• Start PowerShell or use a command window to run the appropriate netsh commands as shown in the example below:
This shows an IPv6 example, use the IPv4 commands if you’re using IPv4 addresses.

Repeat steps 1 - 3 on all remaining Windows 2012/2016/2019 Real Server(s).

For Windows 2012/2016/2019 you can also use the following PowerShell Cmdlets:

The following example configures both IPv4 and IPv6 at the same time:

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

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

To configure just IPv4:

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

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

To configure just IPv6:

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

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

**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 should be configured first, 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.
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.

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:

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

To add a Secondary node - *i.e. create a highly available clustered pair*:

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 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 blue message box at the top of the screen.
Note: Clicking the Restart Heartbeat button on the Primary appliance will also automatically restart heartbeat on the Secondary appliance.

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

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Change</th>
<th>Reason for Change</th>
<th>Changed By</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.7.0</td>
<td>9 August 2019</td>
<td>Styling and layout</td>
<td>General styling updates</td>
<td>RJC</td>
</tr>
<tr>
<td>1.7.1</td>
<td>1 June 2020</td>
<td>New title page</td>
<td>Branding update</td>
<td>AH</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Updated Canadian contact details</td>
<td>Change to Canadian contact details</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>New screenshot for creating a layer 4 VIP</td>
<td>Changes to the appliance WebUI</td>
<td></td>
</tr>
<tr>
<td>1.7.2</td>
<td>17 June 2021</td>
<td>Various minor updates</td>
<td></td>
<td>RJC</td>
</tr>
<tr>
<td>1.8.0</td>
<td>1 January 2022</td>
<td>Converted the document to AsciiDoc</td>
<td>Move to new documentation system</td>
<td>AH, RJC, ZAC</td>
</tr>
</tbody>
</table>
About Loadbalancer.org

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

© Copyright Loadbalancer.org • www.loadbalancer.org