Appliance Quick Start Guide

v6.18

Table of Contents

Loadbalancer.org terminology

Acronym	Terminology
Load Balancer	An IP based traffic manager for clusters
VIP	The Virtual IP address that a cluster is contactable on (Virtual Server)
RIP	The Real IP address of a back-end server in the cluster (Real Server)
GW	The Default Gateway for a back-end server in the cluster
WUI	Web User Interface
Floating IP	An IP address shared by the master & slave load balancer when in a high-availability configuration (shared IP)
Layer 4	Part of the seven layer OSI model, descriptive term for a network device that can route packets based on TCP/IP header information
Layer 7	Part of the seven layer OSI model, descriptive term for a network device that can read and write the entire TCP/IP header and payload information at the application layer
DR	Direct Routing is a standard load balancing technique that distributes packets by altering only the destination MAC address of the packet
NAT	Network Address Translation – Standard load balancing technique that changes the destination of packets to and from the VIP (external subnet to internal cluster subnet)
SNAT *(HAProxy)*	Source Network Address Translation – Load balancer acts as a proxy for all incoming & outgoing traffic
SSL Termination *(Pound)*	The SSL certificate is installed on the load balancer in order to decrypt HTTPS traffic on behalf of the cluster
MASQUERADE	Descriptive term for standard firewall technique where internal servers are represented as an external public IP address. Sometimes referred to as a combination of SNAT & DNAT rules
One Arm	The load balancer has one physical network card connected to one subnet
Two Arm	The load balancer has two physical network cards connected to two subnets
Eth0	Usually the internal interface also known as Gb0
Eth1	Usually the external interface also known as Gb1

What is a virtual IP address?

Most load balancer vendors use the term virtual IP address (VIP) to describe the address that the cluster is accessed from.

It is important to understand that the virtual IP (VIP) refers both to the physical IP address and also to the logical load balancer configuration. Likewise the real IP (RIP) address refers both to the real servers physical IP address and its representation in the logical load balancer configuration.

What is a floating IP address?

The floating IP address is shared by the master and slave load balancer when in a high-availability configuration. The network knows that the master controls the floating IP address and all traffic will be sent to this address. The logical VIP matches this address and is used to load balance the traffic to the application cluster. If the master has a hardware failure then the slave will take over the floating IP address and seamlessly handle the load balancing for the cluster. In scenarios that only have a master load balancer there can still be a floating IP address, but in this case it would remain active on the master unit only.

What are your objectives?

It is important to have a clear focus on your objectives and the required outcome of the successful implementation of your load balancing solution. If the objective is clear and measurable, you know when you have achieved the goal.

Hardware load balancers have a number of flexible features and benefits for your technical infrastructure and applications. The first question to ask is:

Are you looking for increased performance, reliability, ease of maintenance or all three?

Performance	A load balancer can increase performance by allowing you to utilize several commodity servers to handle the workload of one application.
Reliability	Running an application on one server gives you a single point of failure. Utilizing a load balancer moves the point of failure to the load balancer. At Loadbalancer.org we advise that you only deploy load balancers as clustered pairs to remove this single point of failure.
Maintenance	Using the appliance, you can easily bring servers on and off line to perform maintenance tasks, without disrupting your users.

In order to achieve all three objectives of performance, reliability & maintenance in a web based application, your application must handle persistence correctly (see page 31 for more details).

What is the difference between a one-arm and a two-arm configuration?

The number of 'arms' is a descriptive term for how many physical connections (Ethernet ports or cables) are used to connect the load balancers to the network. It is very common for load balancers that use a routing method (NAT) to have a two-arm configuration. Proxy based load balancers (SNAT) commonly use a one-arm configuration.

NB: To add even more confusion, having a 'one-arm' or 'two-arm' solution may or may not imply the same number of network cards.

Loadbalancer.org topology definition:

One-Arm	The load balancer has **one** physical network card connected to **one** subnet
Two-Arm	The load balancer has **two** physical network cards connected to **two** subnets

What are the different load balancing methods supported?

The Loadbalancer.org appliance is one of the most flexible load balancers on the market. The design of the appliance allows different load balancing modules to utilize the core high availability framework of the appliance. Multiple load balancing methods can be used at the same time or in combination with each other.

Layer 4	DR (Direct Routing)	Ultra-fast local server based load balancing *Requires handling the ARP issue on the real servers*	1 ARM
Layer 4	NAT (Network Address Translation)	Fast Layer 4 load balancing, the appliance becomes the default gateway for the real servers	2 ARM
Layer 4	TUN	Similar to DR but works across IP encapsulated tunnels	1 ARM
Layer 7	SSL Termination *(Pound)*	Usually required in order to process cookie persistence in HTTPS streams on the load balancer - ***Processor intensive***	1 or 2 ARM
Layer 7	SNAT (Source Network Address Translation: HAProxy)	Layer 7 allows great flexibility including full SNAT and WAN load balancing, cookie insertion and URL switching ***Not as fast as Layer 4***	1 or 2 ARM

Key:

Recommended for high performance fully transparent and scaleable solutions

Recommended if HTTP cookie persistence is required, also used for numerous Microsoft applications such as Terminal Services (RDP cookie persistence) and Exchange, that require SNAT mode

Only required for Direct Routing implementation across routed networks (rarely used)

Loadbalancer.org Recommendation:

The one-arm direct routing (DR) mode is the recommended mode for Loadbalancer.org installation because it's a very high performance solution with very little change to your existing infrastructure.

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 issue.

The second choice is Network Address Translation (NAT) mode. This is also a fairly 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).

Network engineers with experience of hardware load balancers will have often used this method.

If your application requires that the load balancer handles cookie insertion then you need to use the SNAT configuration (HAProxy). This also has the advantage of a one-arm configuration and does not require any changes to the application servers. However, as the load balancer is acting as a full proxy it doesn't have the same raw throughput as the routing based methods. Please refer to sections D, E & G in the administration manual for configuration of SSL termination or cookie insertion.

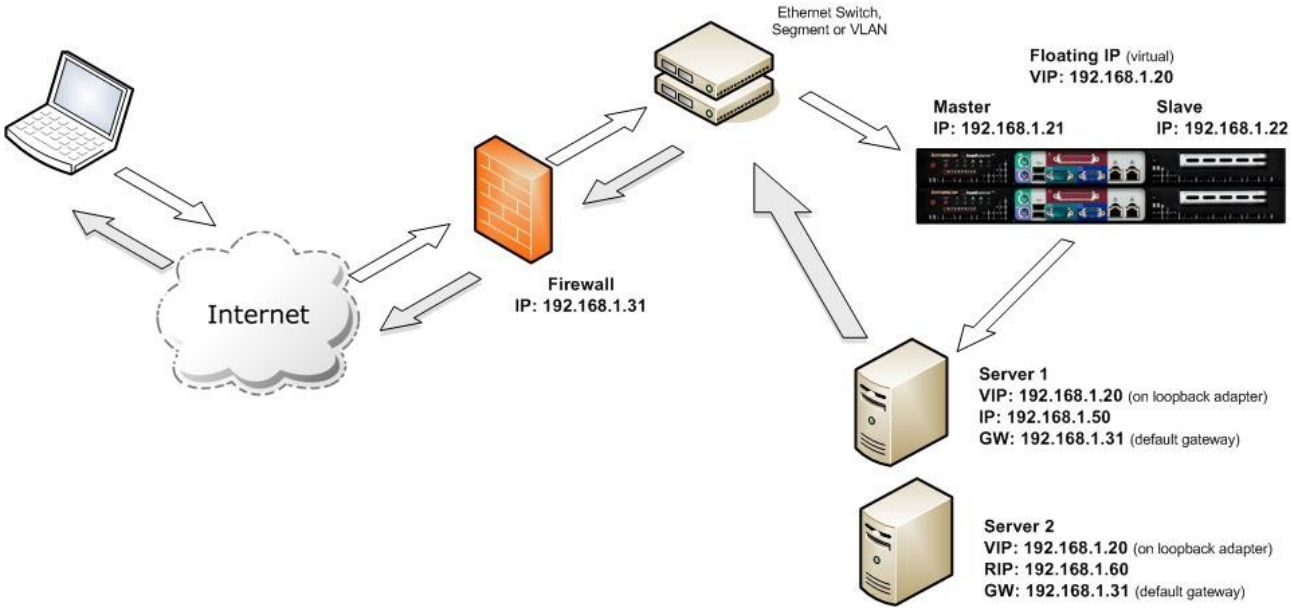
If your application doesn't maintain its own state information then you may need to use cookie insertion, please refer to the full administration manual for configuration details .

The following section describes the different network configuration possibilities for NAT & DR mode in more detail.

High-availability configuration of two Loadbalancer.org appliances

Loadbalancer.org's recommended configuration is to use a clustered pair of load balancers to provide a highly available and resilient load balancing solution. In this configuration, the pair communicates via a heartbeat to determine if the master node is active. Should the master node suffer a failure, the slave will immediately take over any resources hosted on the shared floating IP addresses.

Network diagram: One-Arm – DR Direct Routing (clustered pair)

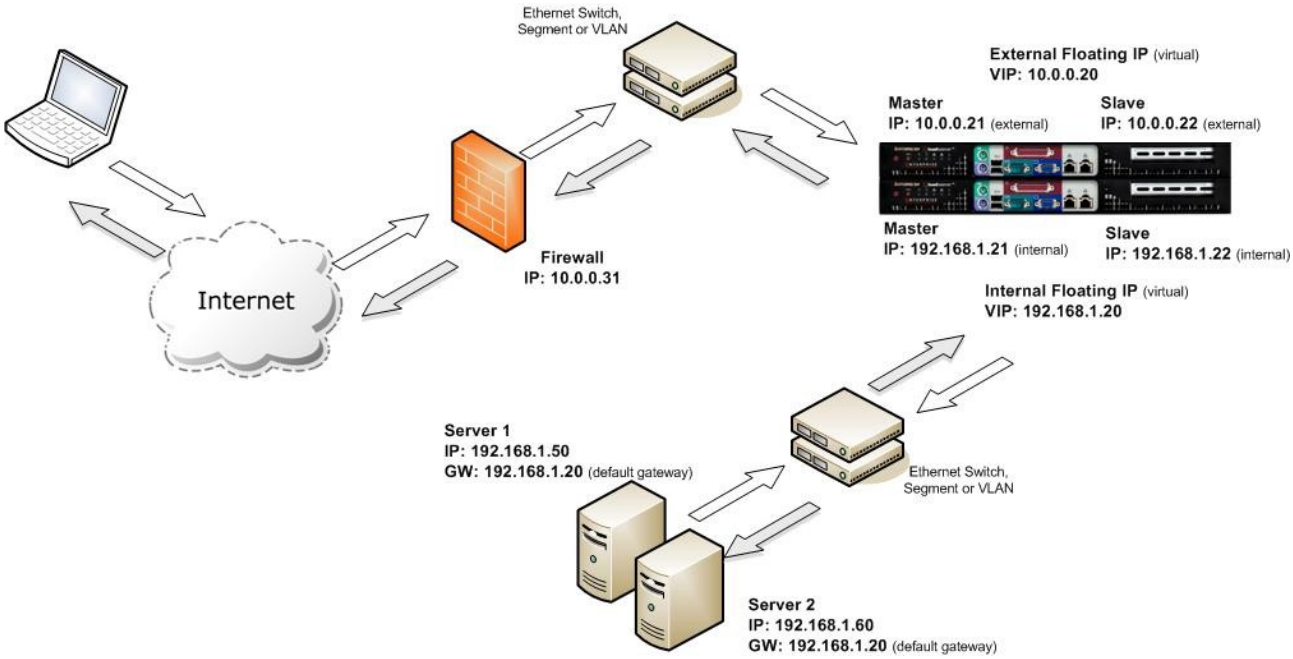


Notes:

- Direct routing works by changing the destination MAC address of the incoming packet on the fly which is very fast
- However, this means that when the packet reaches the real server it expects it to own the VIP. This means you need to make sure the real server responds to the VIP, but does not respond to ARP requests. Please refer to pages 20 - 28 for more details on resolving the ARP issue
- 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
- Direct routing mode enables servers on a connected network to access either the VIPs or RIPs, no extra subnets or routes are required on the network
- The real server must be configured to respond to its own IP address & the VIP
- Port translation is not possible in DR mode i.e. have a different RIP port than the VIP port
- Administration of the load balancers is via any active IP address

When using a clustered pair of load balancers in one-arm DR mode all load balanced services must be configured on a floating IP to enable failover to the slave unit to occur.

Network diagram: Two-Arm - NAT Network Address Translation (clustered pair)



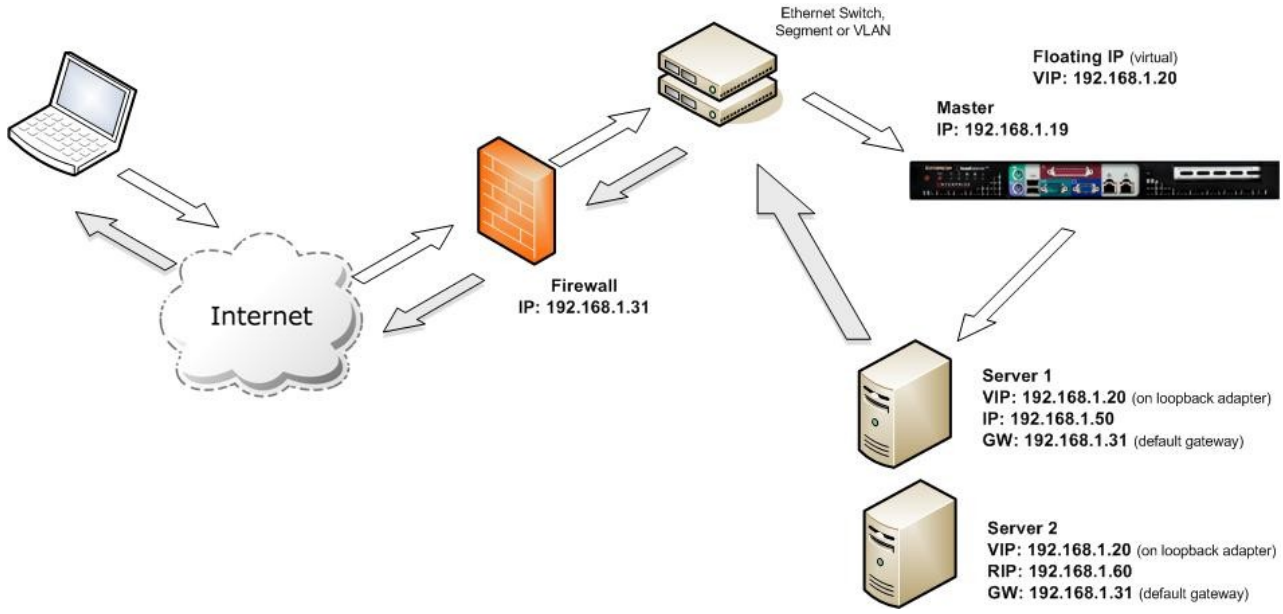
Notes:

- In two-arm NAT mode the load balancer translates all requests from the external virtual server to the internal real servers
- It is a good idea to use *eth1* as your external network and *eth0* as internal, otherwise you will need to change the *autonat* setting in *Edit Configuration* > *Global Settings*
- The real servers *must* have their default gateway configured to point at a floating IP the load balancer
- Real servers are automatically given access to the Internet through the load balancer (via autonat)
- A floating IP must be configured for hosting the virtual server (public access)
- Administration of the load balancers is via any active IP address
- If you want real servers to be accessible on their own IP address for non-load balanced services, e.g. SMTP, you will need to set up individual SNAT and DNAT firewall script rules for each real server. Please refer to Advanced NAT Considerations in section E of the administration manual for more details on this

When using a clustered pair of load balancers in two-arm NAT mode all load balanced services must be configured on a floating IP and the real servers must also have their default gateway directed to a floating IP. This enables failover to the slave unit to occur.

You can also configure the load balancers in one-arm NAT mode, but in order to make the servers accessible from the local network you need to change some routing information on the real servers. Section E of the administration manual provides more detail for one-arm NAT mode. The admin manual is available here : http://www.loadbalancer.org/pdffiles/loadbalanceradministration.pdf

Network diagram: One-Arm – DR Direct Routing (single unit)



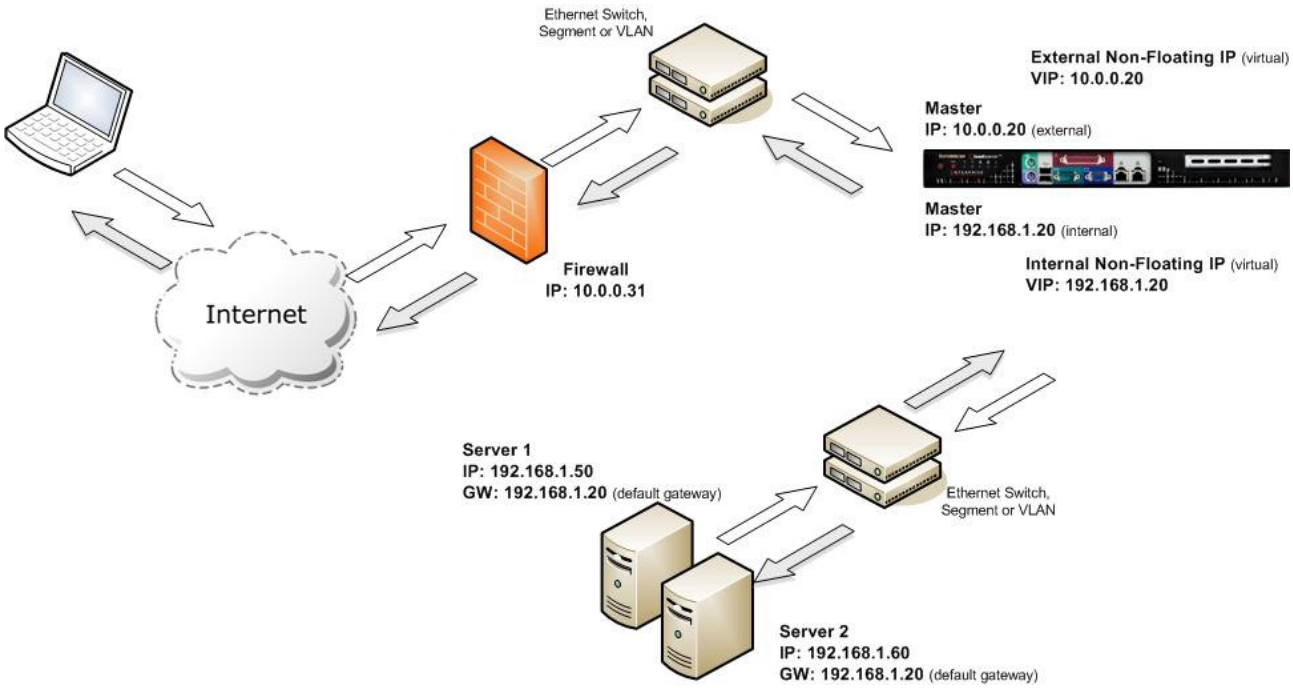
Notes:

- When using a single load balancer only one IP address is required

  NB: Please note however that it is still good practice to use an extra dedicated floating IP to make adding a 2nd unit (to create a clustered pair) much easier should this be needed later
- Direct routing works by changing the destination MAC address of the incoming packet on the fly which is very fast
- However, this means that when the packet reaches the real server it expects it to own the VIP. This means you need to make sure the real server responds to the VIP, but does not respond to ARP requests. Please refer to pages 20 - 28 for more details on resolving the ARP issue
- 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
- Direct routing mode enables servers on a connected network to access either the VIPs or RIPs, no extra subnets or routes are required on the network
- The real server must be configured to respond to its own IP address & the VIP
- Port translation is not possible in DR mode i.e. have a different RIP port than the VIP port
- Administration of the load balancers is via any active IP address

Using a single load balancer introduces a single point of failure for your infrastructure so it's strongly recommended to use two appliances in a clustered pair.

Network diagram: Two-Arm - NAT Network Address Translation (single unit)



Notes:

- In two-arm NAT mode the load balancer translates all requests from the external virtual server to the internal real servers
- It is a good idea to use *eth1* as your external network and *eth0* as internal, otherwise you will need to change the *autonat* setting in *Edit Configuration* > *Global Settings*
- The real servers *must* have their default gateway configured to point at a floating IP the load balancer
- Real servers are automatically given access to the Internet through the load balancer (via autonat)
- Administration of the load balancers is via any active IP address
- If you want real servers to be accessible on their own IP address for non-load balanced services, e.g. SMTP, you will need to set up individual SNAT and DNAT firewall script rules for each real server. Please refer to Advanced NAT Considerations in section E of the administration manual for more details on this

Using a single load balancer introduces a single point of failure for your infrastructure so it's strongly recommended to use two appliances in a clustered pair.

When using a load balancer in two-arm NAT mode, all load balanced services can be configured on the external IP (eth1). The real servers must also have their default gateway directed to the internal IP. Please note that it is good practice to use extra dedicated floating IP's for the VIP and the gateway to make adding 2nd unit (to create a clustered pair) much easier.

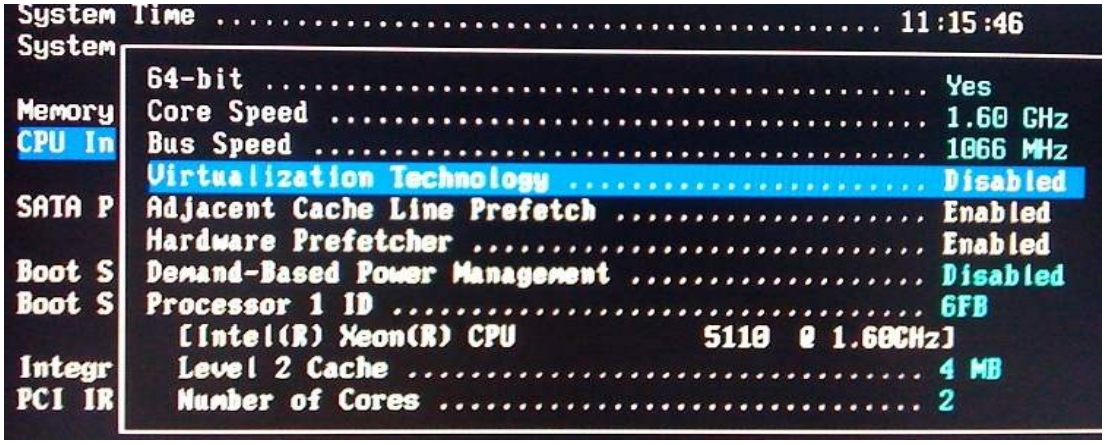
VMware Virtual Appliance

Host Server requirements & preparation for VMware

To be able to successfully run the Loadbalancer.org Enterprise VA under VMware, the following basic server specifications must be met:

- A compatible 64bit CPU
- Virtual Technology hardware support – either Intel-VT or AMD-V compliant CPU's

For an Intel based server, VT may need to be enabled in BIOS as shown in the example below:



If your server is unable to support 64bit guests, a message similar to the following message will be displayed when trying to start the VA:

Appliance download formats

VMware has a number of formats and versions for systems and files. It can get confusing which type / version is needed for your specific environment. The section below explains what is needed for various versions of Vmware. All files can be downloaded from the downloads page on our website.

VMware Player, Workstation & Server

- Download file LBVM.zip (virtualHW.version = 4)
- For VMware server v2.x you can highlight the VA after import and select Upgrade to Hardware v7, for VMware Player, Workstation & Server v1.x no further steps are required

Vsphere Client 4.x & ESX 4.x / ESXi 4.x

- Download file LBVMESX.zip (ovf v1.0, hardware v7)

Virtual Infrastructure Client 2.5.x & ESX 3.x / ESXi 3.x

In this case you have two choices:

- Download LBVM.zip from our downloads page and use the converter for your environment to convert to a compatible VA

  or

- Download file LBVMESX_ovf0.9.zip (ovf v0.9, hardware v4) from the Quick Download Links section of the downloads page

Setting up the Loadbalancer.org Virtual Appliance

1. Download & extract the appropriate file (see previous section)
2. Import the VA:
   - For VMware Server use: ***Virtual Machine > Add VM to Inventory***
   - For Vsphere use: ***File > Deploy ovf Template***
   - For Virtual Infrastructure use: ***File > Virtual Appliance > Import***
3. Start the Virtual Appliance, allow a minute for booting

If two Loadbalancer.org Virtual Appliances are being used, heartbeat is automatically broadcast over eth0 (in a physical appliance, by default heartbeat uses the serial interface)

VMware Tools

Unless you have a specific reason to upgrade the supplied tools then its not really necessary. Our appliances make heavy use of the 64Bit e1000 network driver which is part of the default kernel, the appliance doesn't strictly need any of the extra VMware tools functionality. If you do want to upgrade, detailed instructions are available on our blog at:

http://blog.loadbalancer.org/how-to-upgrade-vmware-tools-on-clusterload-esx-or-loadbalancerorg-va/

Physical Appliance

Unpacking The Appliance

1. Remove all packaging
2. Rack mount the appliance as required using the supplied rails
3. The power supply is an auto sensing unit (115v or 230v)

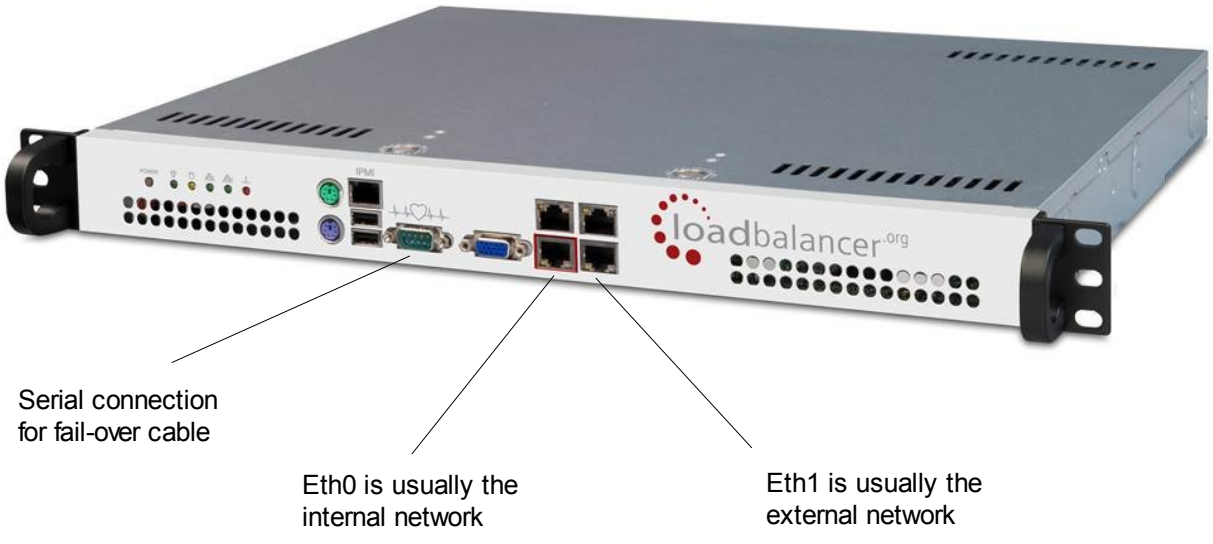
Connections

1. Connect the power lead from the power socket to the mains or UPS
2. Connect your network cable from your switch or hub to the internal network port (*eth0*)
3. If using a two-armed configuration connect a second network cable to the external port (*eth1*)
4. Attach a monitor to the VGA port
5. Attach a keyboard to the USB or PS/2 port
6. Check mains power is on
7. Press the power switch on (fans should start)
8. Allow a minute for booting

If two load balancers are being used, connect a null modem cable (supplied with the appliance) between the two serial fail-over ports and configure the slave first.

Configure & Test

1. Configure the load balancer using the web based wizard
2. Configure the load balancer using the console wizard
3. Add extra real servers via the web administration interface
4. Configuring the real servers for either NAT or DR mode
5. Testing the load balancer configuration



Configuring the Loadbalancer.org appliance using the web based wizard

This section deals with the process of configuring a single load balancer appliance via the web based wizard. The wizard enables you to configure a complete working configuration with one virtual server and one real server. You can then continue in the web interface to make modifications to this basic configuration, add additional Virtual IPs (VIPs), Real Servers (RIPs) etc.

Network interface configuration

Log in to the console:

***Username**:* root
***Password**:* loadbalancer

You can access the web interface either via Links at the console or from a web browser on a client connected to the same network (recommended). By default the IP address for eth0 on the physical appliance is set to 192.168.2.21/24. On the VMware Virtual Appliance, eth0 attempts to obtain an IP address via DHCP. The IP address obtained will be displayed on the console of the VM once the boot process is complete. If a DHCP server cannot be contacted, no IP address will be shown and it will therefore need to be set manually. This can be done using the following console command once logged in:

```
ifconfig eth0 <IP address> <netmask> up
```

NB. This is temporary, the IP address MUST be set via the web interface to make this permanent

Accessing the Web User Interface (WUI)

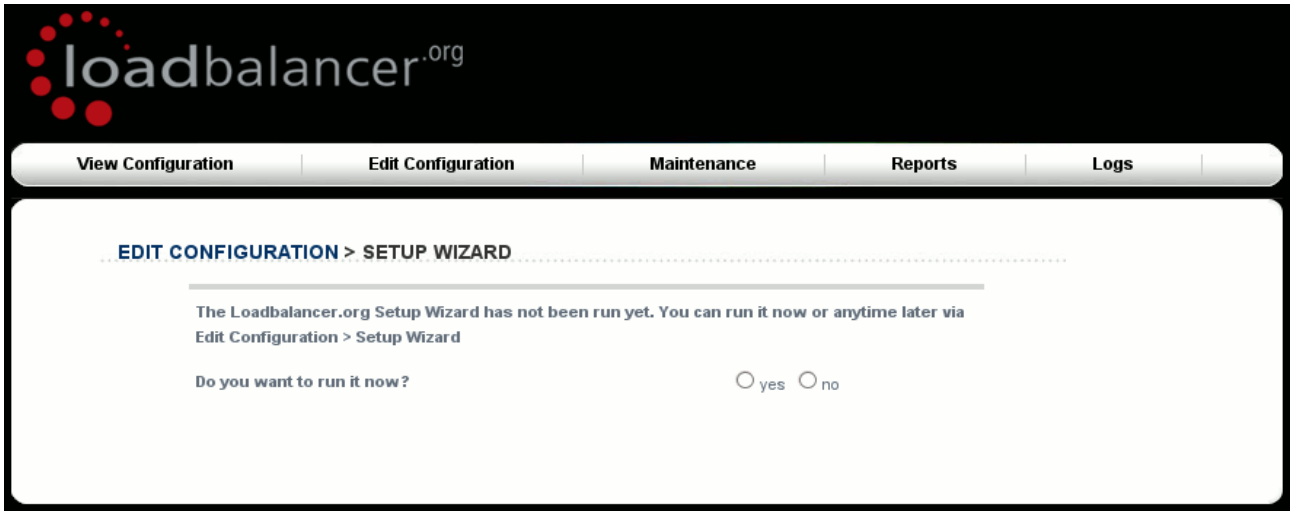
Using a web browser, access the WUI : ***http://<IP address>:9080/lbadmin/***

*(for a physical appliance, the default is **http://192.168.2.21:9080/lbadmin/**)*

***Username**:* loadbalancer
***Password**:* loadbalancer

*NB. If you prefer, you can use the HTTPS administration address : **https://<IP address>:9443/lbadmin/***

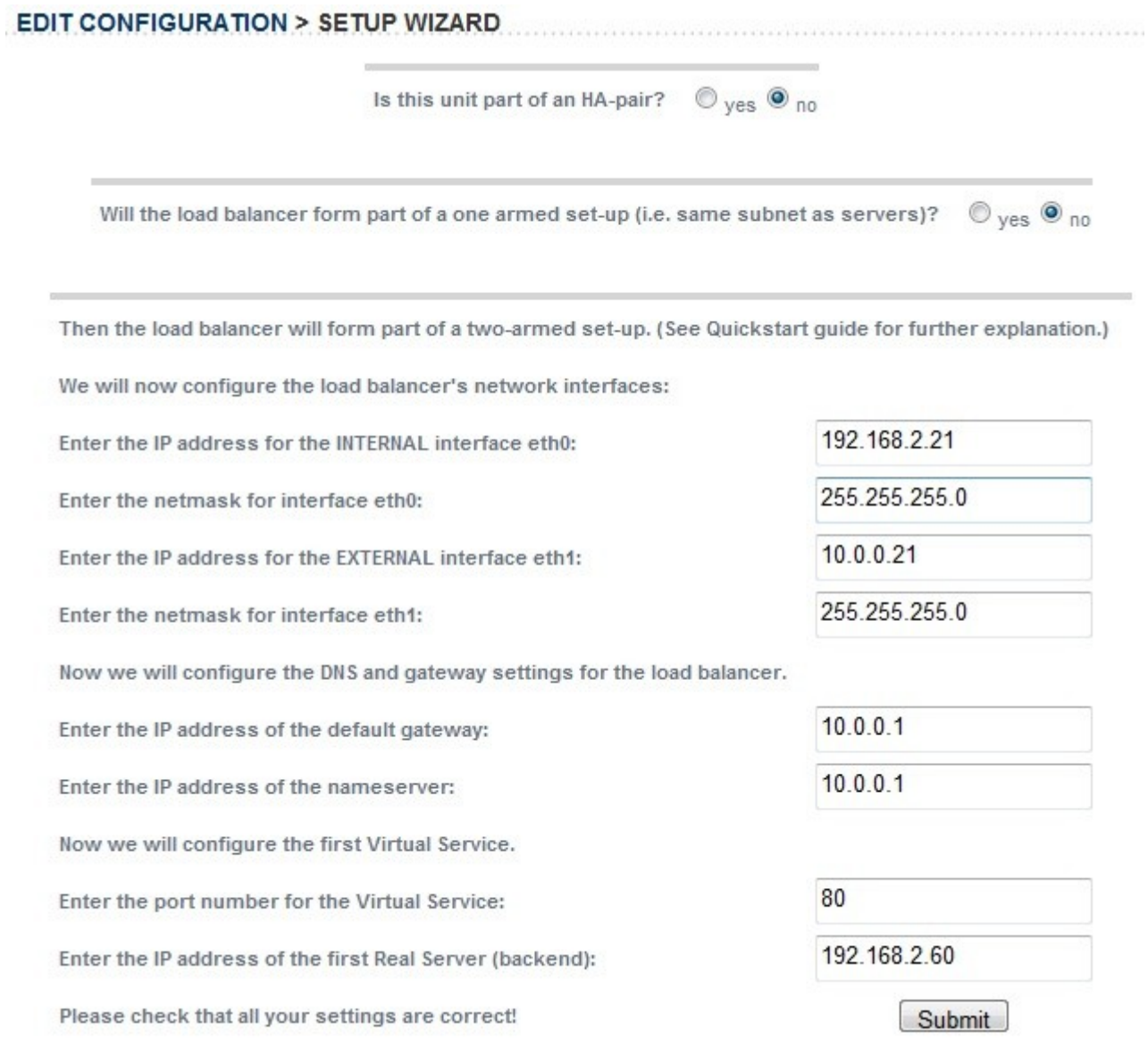
This will take you to the Loadbalancer.org web interface, where the web based configuration wizard will start by default the first time it's accessed. This wizard will ask a series of questions in order to get you started quickly.



All further configuration and administration tasks can then be carried out through the web interface.

Example answers using the wizard for a two-arm NAT configuration

Once you have decided on your load balancing configuration, completing the wizard should be fairly self explanatory. The following example is for a two-arm NAT configuration:



Check that your settings are correct and click **Submit**. Once the wizard is complete the load balancer is configured and ready to use.

For NAT mode you also need to configure the real server to ensure that it uses the internal IP address of the load balancer as its default gateway. Once this is done you can test the virtual server form the external network. By default, the wizard uses the IP address of the external interface for the first virtual server – 10.0.0.21 in this example.

You can now use the *Edit Configuration* menu in the WUI to easily add more virtual or real servers to your configuration.

To restore the manufacturer's settings – at the console use the command **lbrestore** or in the WUI goto *Maintenance* > *Disaster Recovery* > *Restore Manufacturer's Settings*.

Additional Loadbalancer.org configuration (web interface)

This section deals with the configuration of the load balancers via the web interface. The wizard should enable you to have a working virtual server with a single configured real server (back-end). You can use the web interface to add or modify existing virtual and real servers as required.

If you used the web based wizard then you will already be in the web interface. From here all administration tasks can be carried out.

If you chose to use the console wizard then you can now access the web interface either via links at the console or from a web browser on a client connected to the same network (recommended).

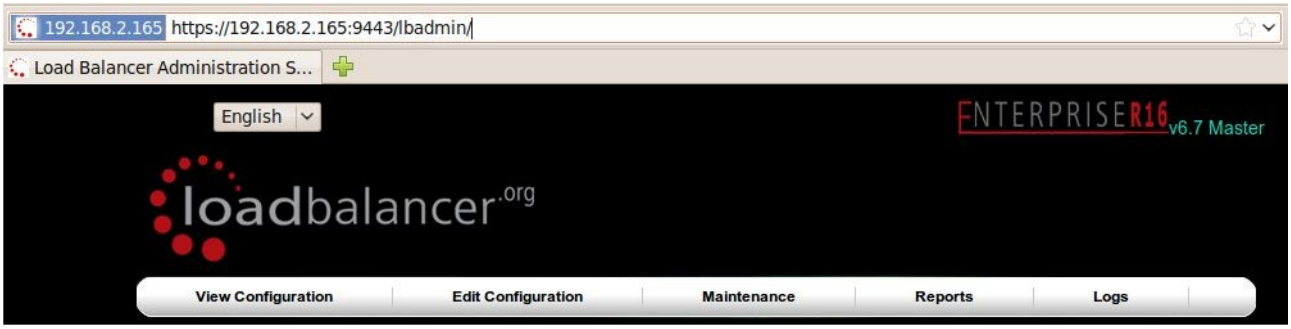
With a web browser access the web interface : ***http://192.168.2.21:9080/lbadmin/***

(replace 192.168.2.21 with the correct address)

Log in to the console:

***Username**:* root
***Password**:* loadbalancer

NB. If you prefer you can use the HTTPS administration address : https://192.168.2.21:9443/lbadmin/

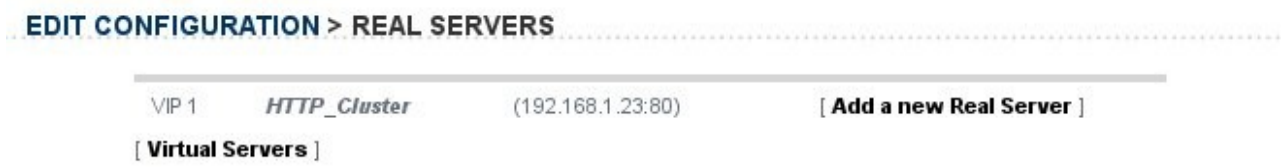


All administration tasks can be carried out through the web interface.

Additional real servers (web interface)

The wizard sets up one virtual server with one real server (back-end server) to send the traffic to. You will need to add any extra servers through the web administration interface:

- Use *Edit Configuration* > *Layer 4 Configuration* > *Real Servers* and you should see your logical virtual servers listed, select the one you want and click on **Add a new Real Server**



- You just need to give the IP address and port number of your web server
- Correctly specify your real servers IP address and service port
- Weight defaults to 1 making real servers active immediately
- Leave the minimum & maximum connections as 0 for unrestricted

- The forwarding method will default to NAT if you have a two-arm configuration or DR if you have a one-arm configuration

You have now finished the configuration of both load balancers for the cluster. Now you must configure the back-end web servers to respond to the load balancer's requests.

Real server configuration for NAT mode

If you are using a two-arm NAT load balancing method the real server configuration is a simple case of configuring the load balancer as the default gateway. The real server must also have a valid IP address in the internal subnet behind the load balancer.

Failure to correctly configure the real servers default gateway is the most common problem in NAT configurations. Please refer to the administration manual for more details.

Real server configuration for DR mode (Linux)

If you are using a one-arm DR load balancing method each web server requires the ARP problem to be handled. Every real server must be configured to respond to the VIP address as well as the RIP address. This is because in DR mode load balanced traffic arrives on the VIP address but the load balancer health checks are on the real server's IP address.

Solving for Linux (with iptables)

This is the recommended method for Linux. You can use iptables (netfilter) on the real server to re-direct incoming packets destined for the virtual server IP address. This is a simple case of adding the following command to your start up script (rc.local):

```
iptables -t nat -A PREROUTING -p tcp -d 10.0.0.21 -j REDIRECT
```

i.e. Redirect any incoming packets destined for 10.0.0.21 (virtual server) to my local address.

(Don't forget to change the IP address to be the same as your virtual server)

Solving for Linux – alternative method (with arp_ignore sysctl values)

Each real server needs a loopback IP address to be configured as the VIP. This address needs to be stopped from responding to ARP requests and the web server needs to be configured to respond to this IP address.

With most modern Linux kernels (>2.6) you can alter the ARP behavior allowing you to configure a loopback adapter without worrying about ARP issues. To do this just add the following lines to /etc/sysctl.conf and re-boot, or run /sbin/sysctl.conf -p to reload the file:

```
net.ipv4.conf.all.arp_ignore=1
net.ipv4.conf.eth0.arp_ignore=1
net.ipv4.conf.eth1.arp_ignore=1
net.ipv4.conf.all.arp_announce=2
net.ipv4.conf.eth0.arp_announce=2
net.ipv4.conf.eth1.arp_announce=2
```

Alternatively, the following commands may be used to change the settings interactively during runtime:

```
echo 1 > /proc/sys/net/ipv4/conf/all/arp_ignore
echo 1 > /proc/sys/net/ipv4/conf/eth0/arp_ignore
echo 1 > /proc/sys/net/ipv4/conf/eth1/arp_ignore
echo 2 > /proc/sys/net/ipv4/conf/all/arp_announce
echo 2 > /proc/sys/net/ipv4/conf/eth0/arp_announce
echo 2 > /proc/sys/net/ipv4/conf/eth1/arp_announce
```

Once you have configured your Linux real server so that it won't respond to ARP requests for the loopback adapter you can configure your VIPs as follows:

```
ifconfig lo:0 VIP netmask 255.255.255.255 up
```

To make this permanent and reboot safe you may include this command in rc.firewall or in a equivalent customizable start-up script.

Failure to correctly configure the real servers to handle the ARP problem is the most common problem in DR configurations. Please refer to the administration manual for more details.

Real server configuration for DR mode (Windows)

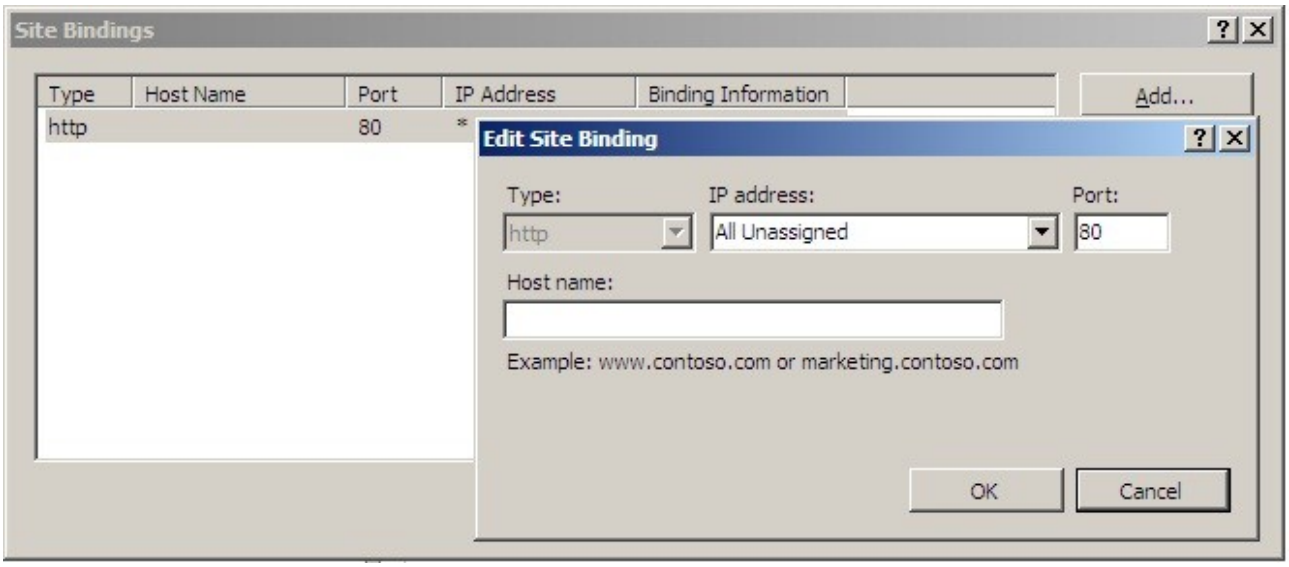
If you are using a one-arm DR load balancing method, each web server requires the ARP problem to be handled:

- Each real server must be configured to respond to the VIP address as well as the RIP address. This is because in DR mode load balanced traffic arrives on the VIP address but the load balancer health checks are on the real IP address
- Each real server must have the MS loopback adapter installed and configured
- The Microsoft loopback adapter must be configured to deal with the ARP problem

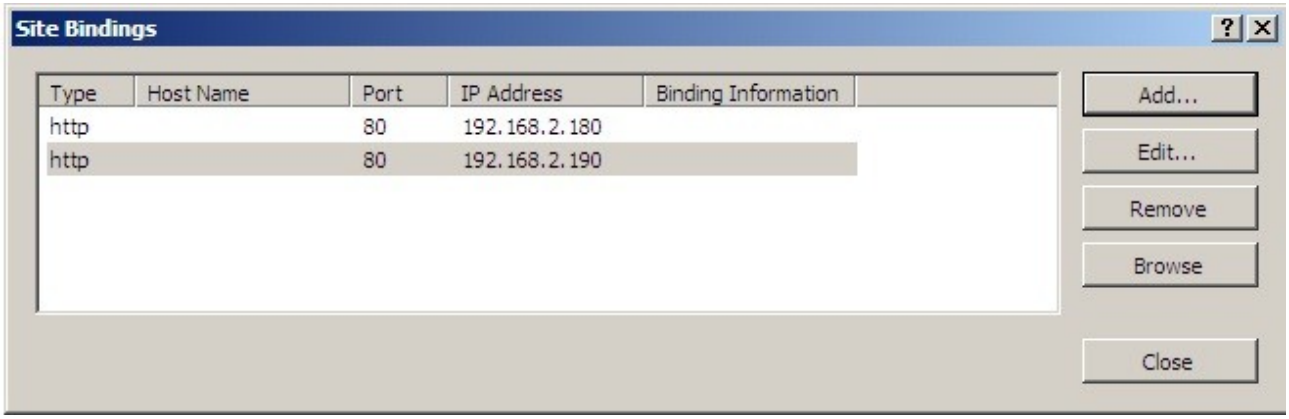
Remember that for all real servers in Direct Routing mode the load balanced application must respond to both the virtual IP as well as the servers real IP. With Windows the IP address must either be set to (All Unassigned) or use the Advanced tab to add a second IP address.

Configuring IIS to respond to both the RIP and VIP

By default, IIS listens on all configured IP addresses, this is shown in the example below. As can be seen the IP address field is set to “All Unassigned”.



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 Server IP address (VIP) as shown below:

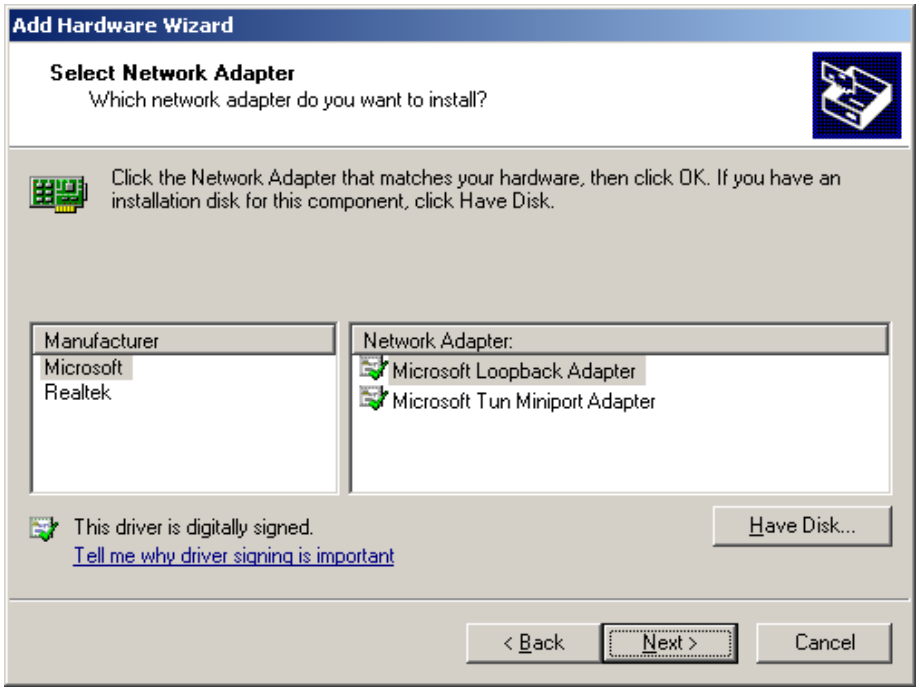


Resolving ARP issues for Windows server 2000 / 2003 (DR mode only)

Windows server supports the direct routing (DR) method through the use of the MS loopback adapter to handle the traffic. The IP address on the loopback adapter must be set to be the same as the Virtual Servers IP address (VIP). If the real server is included in multiple VIPs, you can add additional IP addresses to the loopback adapter that correspond to each VIP.

Installing the Microsoft loopback adapter

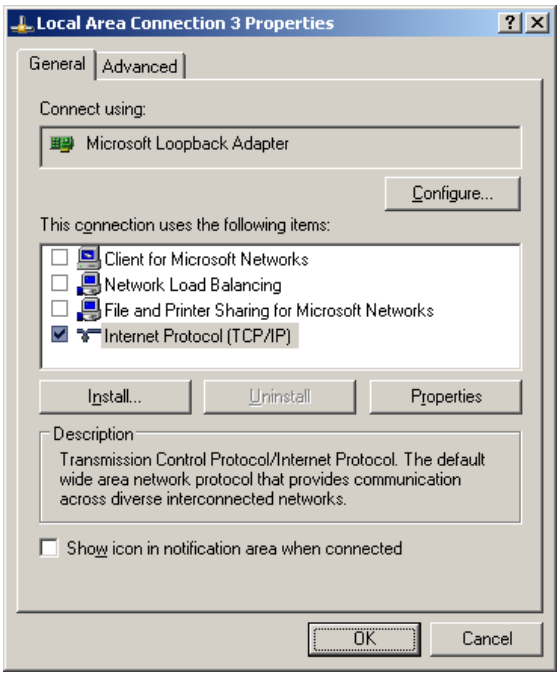
1. Open the Control Panel and double-click Add Hardware
2. Once the Hardware Wizard opens, click Next
3. Select 'Yes, I have already connected the hardware', click Next
4. Scroll to the bottom of the list, select 'Add a new hardware device' and click Next
5. Select 'Install the hardware that I manually select from a list (Advanced)', click Next
6. Select 'Network adapters', click Next
7. Select 'Microsoft' & 'Microsoft Loopback Adapter', click Next



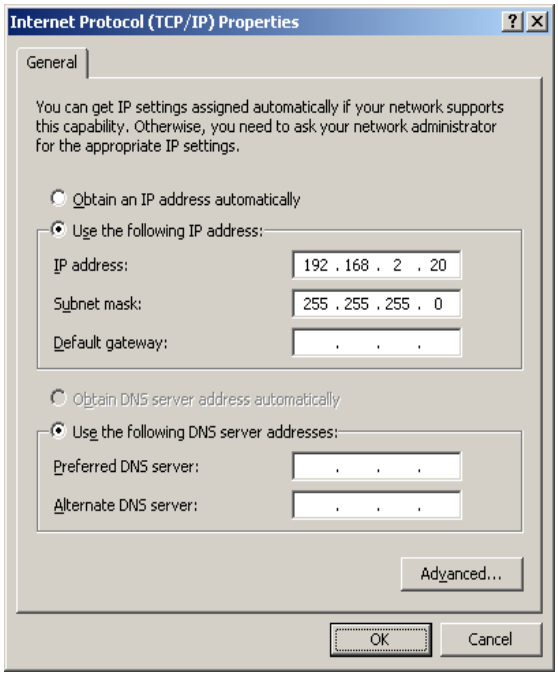
8. Click Next to start the installation, when complete click Finish

Configuring the loopback adapter

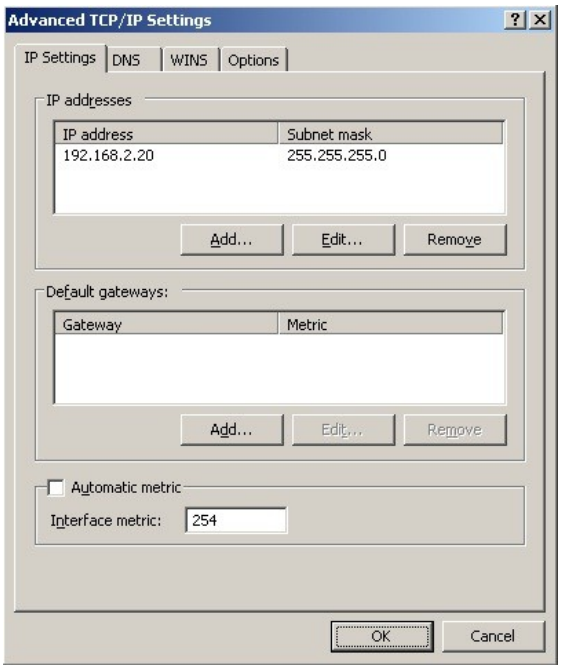
1. Open the Control Panel and double-click Network Connections
2. Right click the new loopback adapter and select properties



3. Un-check all items except Internet Protocol (TCP/IP)
4. Select Internet Protocol (TCP/IP), click Properties and configure the IP address to be the same as the Virtual Server (VIP), e.g. 192.168.2.20/24



5. Click on the *Advanced* button and change the Interface Metric to 254 (This stops the adapter responding to ARP requests).



6. Click OK on the Advanced and TCP/IP popup windows, then click Close on the Local Area Connection window to save the new settings

7. Now repeat the above process for all other Windows 2000 / 2003 real servers

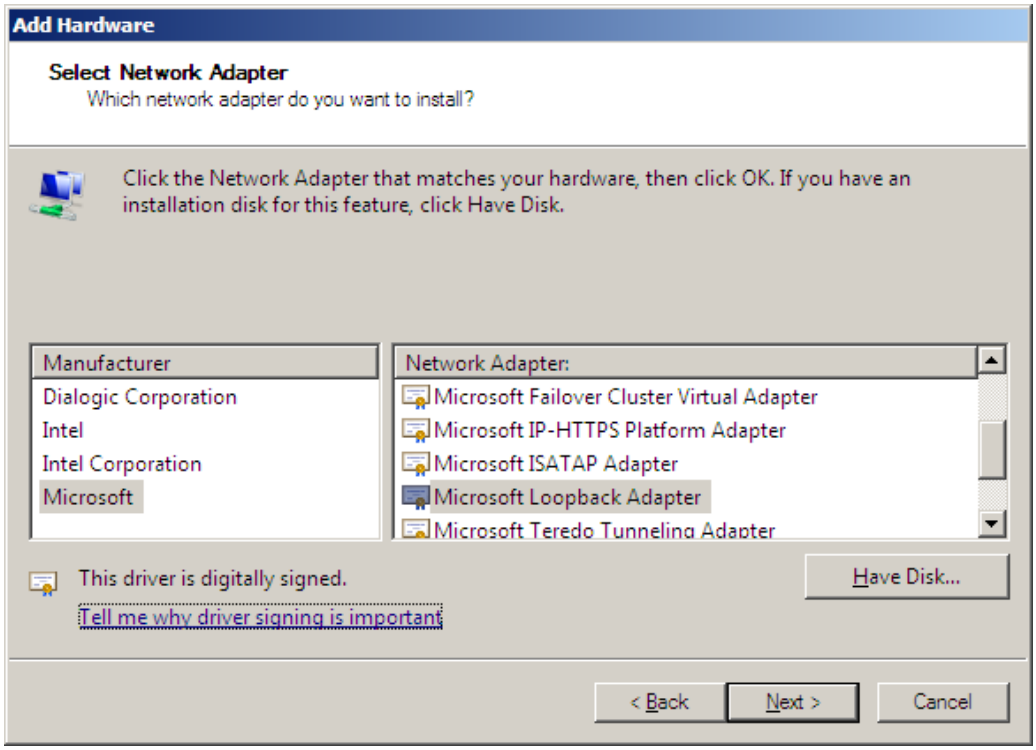
For Windows server 2003 SP1 & above, if you have enabled the built-in firewall, you will need to create an exception to enable access to the web server. This exception by default will allow traffic on both the network and loopback adapters.

Resolving ARP issues for Windows server 2008 (DR mode only)

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.

Installing the Microsoft loopback adapter

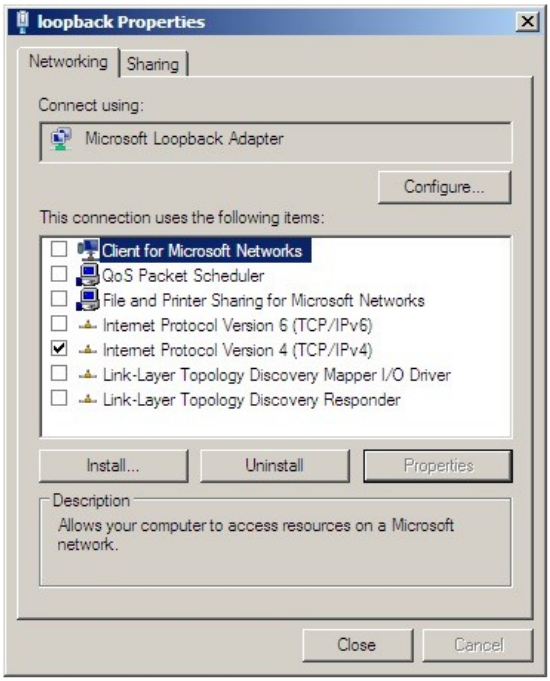
1. Click Start, select Run and enter ***hdwwiz*** to start the Hardware Installation Wizard
2. When the Wizard has started, click Next
3. Select 'Install the hardware that I manually select from a list (Advanced)', click Next
4. Select 'Network adapters', click Next
5. Select 'Microsoft' & 'Microsoft Loopback Adapter', click Next



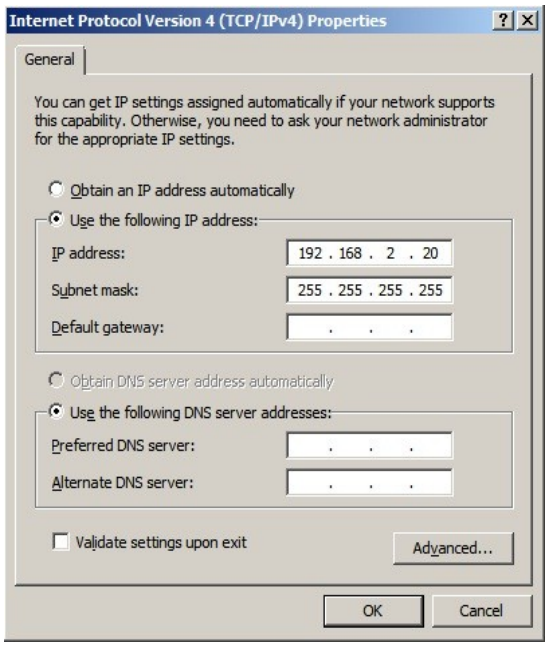
6. Click Next to start the installation, when complete click Finish

Configuring the loopback adapter

1. Open Control Panel and double-click Network and Sharing Centre
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)
5. Select Internet Protocol Version (TCP/IPv4), click Properties and configure the IP address to be the same as the Virtual Server (VIP), with a full subnet mask e.g. 192.168.2.20/32



6. Click OK on the TCP/IP popup window, then click Close on the Local Area Connection window to save the new settings

7. For Windows 2008, its not necessary to modify the interface metric on the advanced tab and should be left set to Automatic

8. Now repeat the above process for all other Windows 2008 real servers

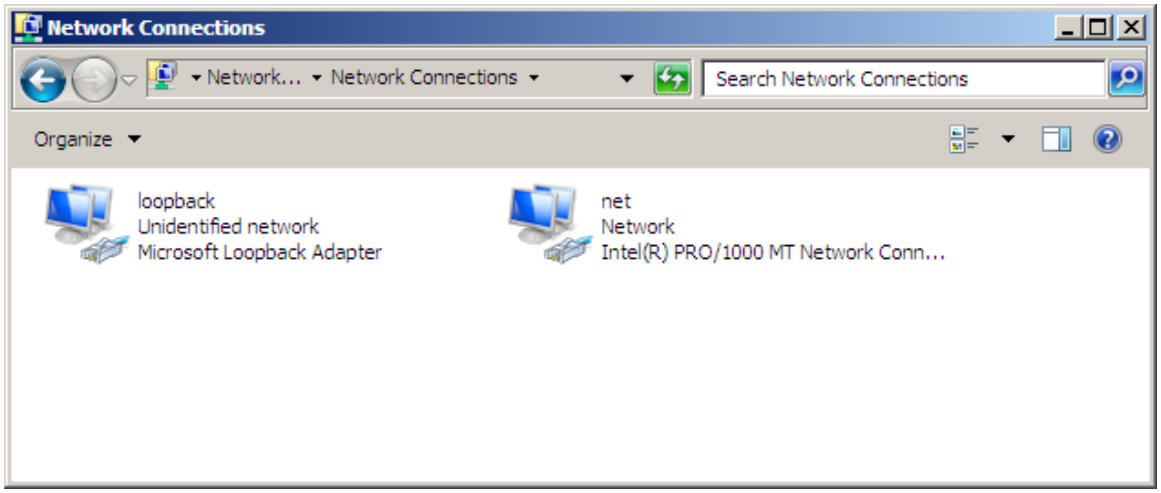
Configuring strong / weak host behaviour

Windows XP 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 Vista and Windows Server 2008 supports strong host sends and receives for both IPv4 and IPv6 by default.

To ensure that the Windows 2008 is running in the correct mode to respond to the VIP, the following commands must be run in a command window on the real server :

```
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”. If you prefer to leave your current NIC names, then the commands above must be modified accordingly.



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.

If you prefer to use the index number for the interface, you can look up the index number using the following command:

```
netsh interface ipv4 show interface
```

then substitute the relevant index number for “net” and “loopback” in the three netsh commands

e.g. if the index number for the loopback adapter is 12, the first netsh command would be:

```
netsh interface ipv4 set interface 12 weakhostreceive=enabled
```

Verifying netsh Settings

To verify that settings have been configured correctly, run the following command on each real server to clearly list the settings that have been applied to the interface:

```
netsh interface ipv4 show interface <interface name>
```

i.e. for the 'loopback' adapter run : `netsh interface ipv4 show interface loopback`
i.e. for the 'net' adapter run : `netsh interface ipv4 show interface net`

e.g.

```
C:\Users\Administrator>netsh interface ipv4 show interface loopback

Interface loopback Parameters
----------------------------------------------
IfLuid                                 : ethernet_9
IfIndex                                : 15
State                                  : connected
Metric                                 : 30
Link MTU                               : 1500 bytes
Reachable Time                         : 28500 ms
Base Reachable Time                    : 30000 ms
Retransmission Interval                : 1000 ms
DAD Transmits                          : 3
Site Prefix Length                     : 64
Site Id                                : 1
Forwarding                             : disabled
Advertising                            : disabled
Neighbor Discovery                     : enabled
Neighbor Unreachability Detection      : enabled
Router Discovery                       : dhcp
Managed Address Configuration          : enabled
Other Stateful Configuration           : enabled
Weak Host Sends                        : enabled
Weak Host Receives                     : enabled
Use Automatic Metric                   : enabled
Ignore Default Routes                  : disabled
Advertised Router Lifetime             : 1800 seconds
Advertise Default Route                : disabled
Current Hop Limit                      : 0
Force ARPND Wake up patterns           : disabled
Directed MAC Wake up patterns          : disabled


C:\Users\Administrator>
```

This shows that the settings have been applied correctly.

For Windows server 2008, if you want to leave the built-in firewall enabled, you'll either need to enable the relevant default firewall exceptions or create your own to enable access to the web server. By default these exceptions will allow traffic on both the network and loopback adapters.

Failure to correctly configure the real servers to handle the ARP problem is the most common mistake in DR configurations.

Real server configuration for SNAT mode

When using Layer7 (HAproxy) Virtual Servers, no changes are required to the real servers.

Testing the load balancer configuration

For testing add a page to each real web servers root directory e.g. test.html and put the server name on this page.

Now you need a couple of clients to do the testing. Open up a web browser on two different clients and enter the URL for the VIP i.e. http://192.168.1.20/.

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 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.*

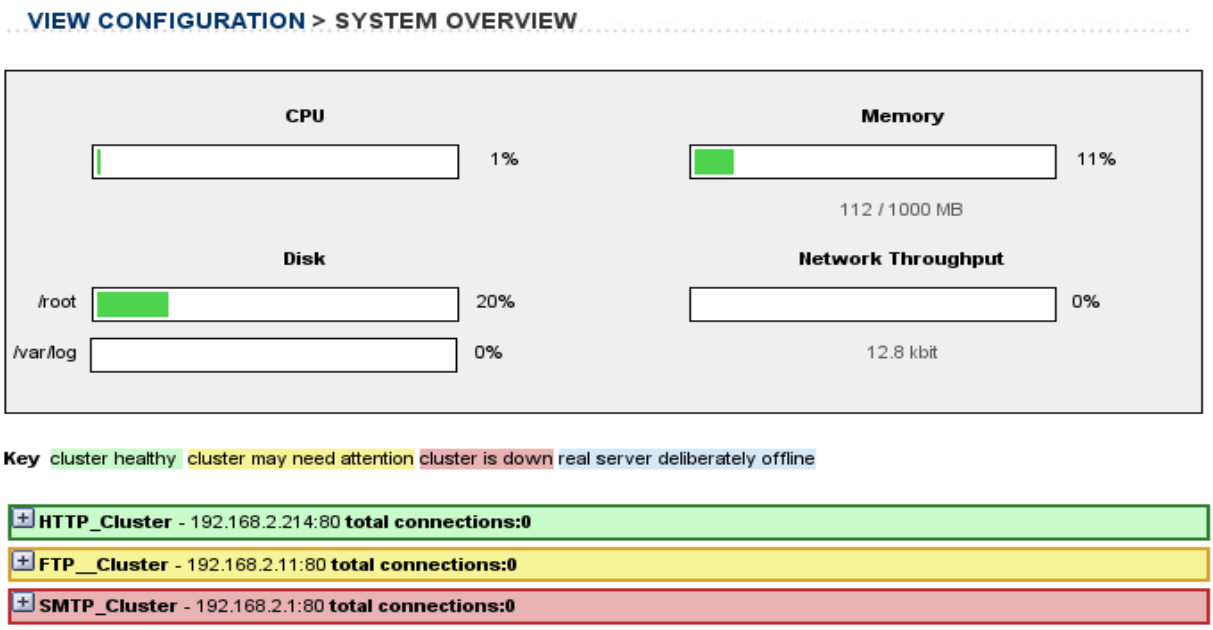
When using a two-arm NAT load balancing method the test client must be in the external subnet.

Connection error diagnosis

If you get a connection error when trying to access the VIP then:

1. Check *View Configuration* > *Network Configuration* and make sure that the VIP is active on the load balancer, if not check *Logs* > *Heartbeat* for errors.

2. Check *Maintenance* > *System Overview* and make sure none of your VIPs are highlighted in red. If they are, your cluster is down and you should see health check diagnosis (next page). Green indicates a healthy cluster, yellow indicates that your cluster may need attention (one real server may be down), and blue indicates a real server has been deliberately taken offline.



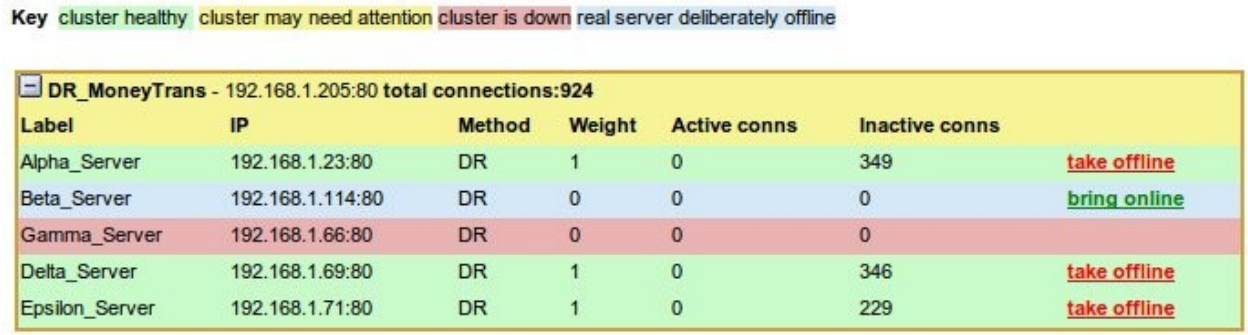
3. If the VIP is still not working then check *Reports* > *Current Connections* to see the current traffic in detail, any packets marked SYN_RECV imply incorrect real server configuration. Check that any NAT mode servers have the correct default gateway and any DR mode servers are responding to the VIP as well as their own IP.

Health check diagnosis

Go to the *Maintenance* > *System Overview* section of the web interface and check that when you use 'take offline' the connections are redirected to the rest of the cluster as expected.

Pull the network cable out of one of the web servers, wait a few seconds (for the load balancer to detect the change) and then refresh the browsers on both clients. They should now both switch to the same server (as one has been removed from the load balancing list).

Put the network cable back in to the web server, wait a few seconds and then refresh the browsers again. They should now show different web servers again.



Key cluster healthy cluster may need attention cluster is down real server deliberately offline

DR_MoneyTrans - 192.168.1.205:80 **total connections:924**

Label	IP	Method	Weight	Active conns	Inactive conns	
Alpha_Server	192.168.1.23:80	DR	1	0	349	take offline
Beta_Server	192.168.1.114:80	DR	0	0	0	bring online
Gamma_Server	192.168.1.66:80	DR	0	0	0	
Delta_Server	192.168.1.69:80	DR	1	0	346	take offline
Epsilon_Server	192.168.1.71:80	DR	1	0	229	take offline

The example above shows that the requested status of Gamma_Server is down (red). This implies that the real server has failed a health check; you can investigate this using *Logs* > *Ldirectord*. If you know the real server should be active, you may need to increase the health check time-out *Edit Configuration* > *Global Settings*.

The Beta_Server however is blue, this indicates that it is deliberately in maintenance mode. You can use 'bring online' to make it active.

Testing high-availability for a Loadbalancer.org HA-pair

To test fail-over of a clustered pair of load balancers, power down the master and check that the slave unit takes over all the floating IP(s).

THIS IS IMPORTANT! and should always be done before going live. This proves the resilience of the cluster and makes you aware of the failover / failback process. Please also refer to the admin guide for details of the hb_takeover command which can be used to force a failover to the device where the command is run from.

If fail-over does not occur correctly check *Logs* > *Heartbeat* on both nodes for any errors.

When testing load balancer fail-over, don't just pull the serial cable and network cable out. These will not cause a fail-over and will invalidate the cluster configuration (split brain). You can configure fail-over on network failure but it is not enabled by default.

Does your application cluster correctly handle its own state?

Load balancers work most effectively if the application servers are completely stateless. This means that if a web server fails and is automatically taken out of the cluster; then all the current user sessions will be transferred to other servers in the cluster without the users needing to re-login to the application again. *If your application doesn't have a persistent data store then you can't have seamless fail over for your back-end servers*.

Web based applications are inherently stateless and an ideal candidate for load balancing. However, Do your web servers store persistent information on local drives?

- Images (jpeg, png, gif etc.)
- Files (html, php, asp etc.)

If so these files either need to be on shared storage such as an NFS/CIFS mount, or they need to be replicated to all of the nodes in the cluster.

Replication solutions for shared data

On UNIX you can use the RSYNC command to replicate files, on Windows Server you can use RSYNC as well but you may prefer ROBOCOPY from the Windows Server Resource Kit. Usually you will upload your content to one master server and then replicate it to the other servers in the cluster.

Solutions for session data

Standard ASP and PHP session data is stored locally by default, leaving your session data in a local store will prevent you from implementing seamless application server fail-over in your cluster. If an application server fails, all of the local session data will be lost and your user will need to re-log in and possibly lose shopping baskets etc.

This problem is easily resolvable by implementing a shared persistent data store for the cluster. This is usually either done with a shared back-end database or a shared memory solution.

What do you do if your application is not stateless?

Some applications require state to be maintained such as:

- Terminal Server
- SSH
- FTP (upload)
- SMTP (incoming)

You may also find that you are unable to modify your HTTP/HTTPS based application to handle shared session data.

For these cases, you can use persistence based on source IP address. You lose the ability to have transparent fail-over, but you do still get increased capacity and manageability. This persistence problem occurs with all load balancers and all vendors use standard methods and technologies to mitigate the issue.

Loadbalancer.org persistence methods

- Source IP (subnet)
- Cookie (Active or Passive)

The standard Layer 4 persistence method is source IP persistence, you can handle millions of persistent connections at Layer 4. Just modify your virtual server to be persistent if you require source IP persistence.

Cookies are a Layer 7 based persistence method that can offer more even traffic distribution and also handle any clients where the source IP address may change during the session (e.g. mega proxies).

NB. Cookies can only be used in HTTP/HTTPS based applications (see the administration manual, example 3 in Section D for more details).

Loadbalancer.org technical support

If you have any questions regarding the appliance don't hesitate to contact the support team support@loadbalancer.org or your local reseller.

For more detailed explanations and complex configuration details please refer to our full administration manual which is available at: http://www.loadbalancer.org/pdffiles/loadbalanceradministration.pdf