

VMware VDM 2 Load Balancing Guide

VMware® Virtual Desktop Infrastructure

VMware Virtual Desktop Infrastructure (VDI) transforms the way customers use and manage desktop operating systems. Desktop instances can be deployed rapidly in secure data centers to facilitate high availability and disaster recovery, protect the integrity of enterprise information, and remove data from local devices that are susceptible to theft or loss. Isolating each desktop instance in its own virtual machine eliminates typical application compatibility issues and improves users' personal computing environments.

About This Guide

This guide provides an in-depth view of how to increase the scalability and availability of a VMware VDI solution using the Virtual Desktop Manager 2 (VDM 2) connection broker. Written primarily for architects and system administrators, it describes the components involved in load balancing VDM 2 and provides reference examples using free or low-cost load balancing technologies.

Overview

Once appropriate server-grade hardware has been selected, load balancing become an important consideration for addressing a configuration's scalability and fault tolerance.

In general, load-balanced configurations use multiple VDM Connection Servers installed in a primary-and-replica manner, with the first server installed as the primary and subsequent servers are installed as replicas. VDM Connection Servers provide session management and handle all incoming client requests, directing them to the appropriate virtual desktop session, and VDM Security Servers provide SSL tunneling capabilities for encrypting communication between the client devices and the VDM Connection Servers.

The configuration of a load-balanced solution largely depends on the requirements of the organization for which it is being deployed. Companies that already have a load balancing solution in place may be able to utilize it for VDI since the load generated by the VDI solution is minimal. Both hardware-based load balancing appliances and inexpensive (or free) software-based load balancing products can be considered as candidate solutions.

Deployment Scenarios

VMware VDI-based solutions are usually deployed either with VDM Connection Servers in a LAN or with VDM Security Servers in a DMZ. DMZ deployment is typically used to provide access for remote work-from-home users or branch offices where desktops are deployed and managed from a centralized, secure corporate data center and accessed across a WAN.

Required Components for Deploying a Load-balanced Solution

- Third-party hardware or software load balancer
- VDM 2 Security Servers (optional based on architecture)
- VDM 2 Connection Servers
- VMware Virtual Infrastructure 3
- VMware VirtualCenter 2.x
- Active Directory

Load Balancing VDM Connection Servers

In some cases, such as LAN-based deployments, users can also connect directly with VDM Connection servers. In this case, no VDM Security servers are deployed because the VDM Security Server component is installed as part of a VDM Connection Server.

LAN-based connections can be tunneled or non-tunneled. When tunneling is enabled, all VDI traffic is SSL-encrypted and tunneled through a VDM Connection Server. When Direct Connect to Virtual Desktop is enabled, session traffic is not routed through the VDM Connection servers or SSL encrypted. When a VDM client receives information from the VDM Connection Server about which desktop instance to connect to, it establishes a session directly with the virtual desktop; this enables direct communication between the client device and the virtual desktop instance. In such cases, a load balancing solution should be deployed to ensure the highest level of scalability and availability.

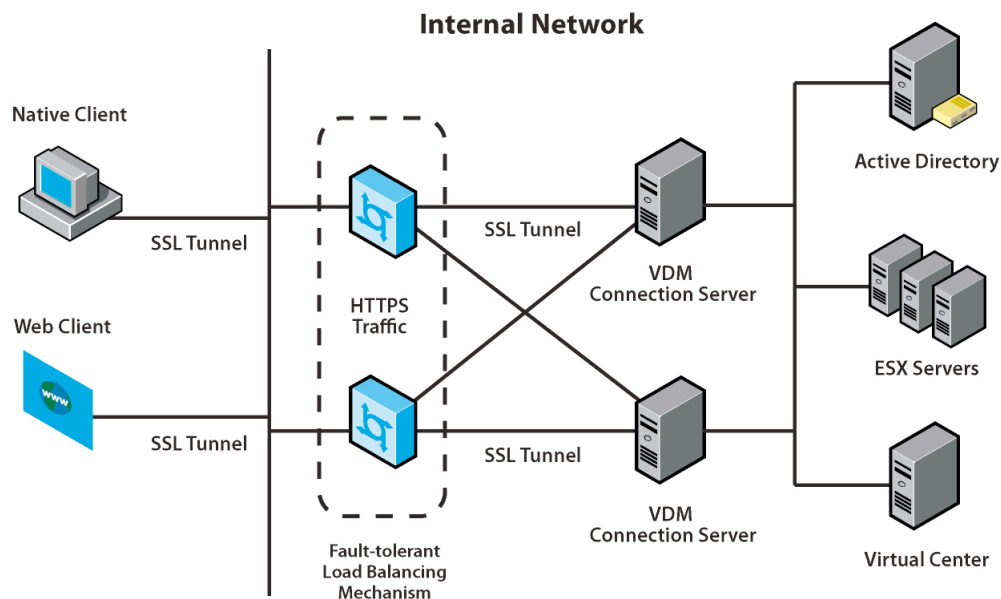


Figure 1. Load Balancing VDM Connection Servers in a LAN

Load Balancing VDM Security Servers Inside a DMZ

When VDM Security Servers are deployed in a DMZ, each security server is associated with a VDM connection server inside the trusted network during the installation process. This architecture requires only a few ports to be opened on the firewall to allow clients to connect with VDM Security Servers inside the DMZ and a few ports to be configured for communication between the VDM Security Servers and the VDM Connection Servers inside the internal network.

Example Firewall Rules

	Source	Protocol	Port	Destination	Comments
Outer firewall	Any	TCP	443 (or 80)	VDM Security Servers	443 (HTTPS) by default, 80 (HTTP) if SSL is disabled
Inner firewall	VDM Security Servers	TCP	8009	Associated VDM Connection Server	Forwarded web traffic (AJP)
Inner firewall	VDM Security Servers	TCP	4001	Associated VDM Connection Server	JMS connection
Inner firewall	VDM Security Servers	TCP	3389	Any desktop VM	RDP traffic from client to desktop VM

One benefit of this approach is that if a VDM Security Server is compromised, the intruder cannot attack corporate resources or services such as Active Directory. When VDM Security Servers are deployed inside the DMZ, they should also be load-balanced inside the DMZ to maximize scalability and fault tolerance.

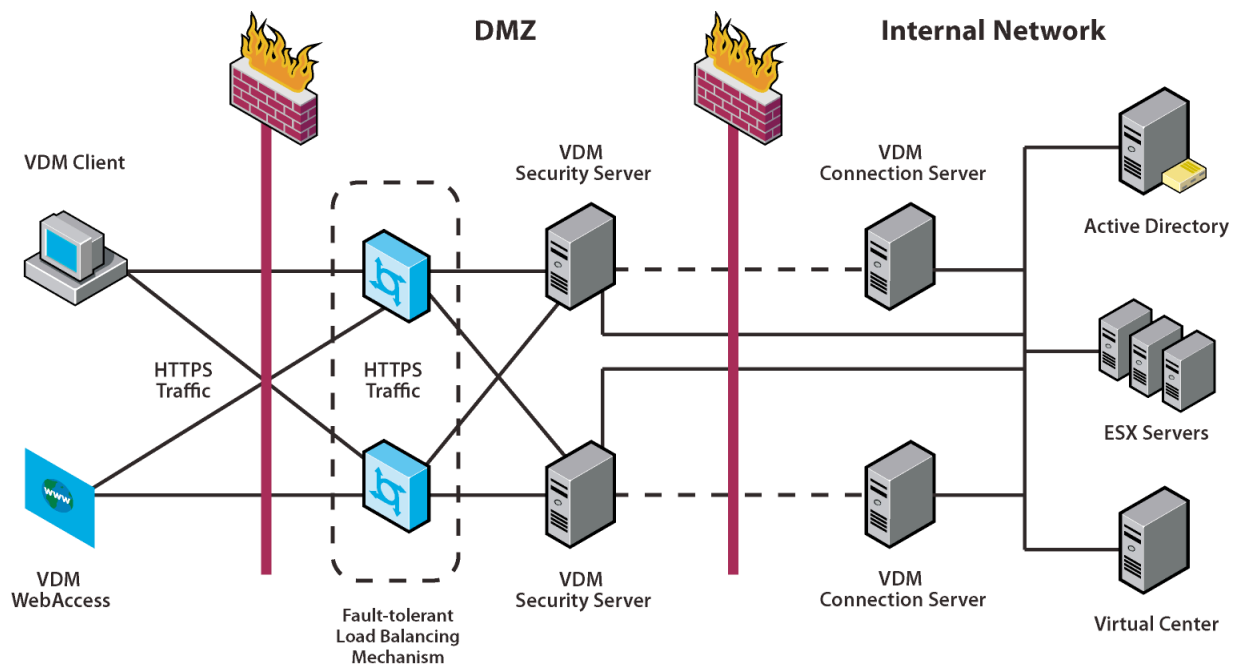


Figure 2. Load Balancing VDM Security Servers in a DMZ

Load Balancing Solution Recommendations

The *easiest* load balancing solution to implement from a technical point of view is Round-robin DNS. However, this approach has a significant disadvantage from a failover perspective: if one of the servers fails, it has to be removed from the DNS list of records corresponding to the load-balanced domain name. It can take several hours before the removal of DNS names from the list of records can be replicated and the cached record expires across the internet DNS servers. Meanwhile, remote access client connections can fail if directed to a server that is out of service.

To avoid becoming a single point of failure, the load balancer should support a redundancy and failover mechanism, typically at the network level. For example, two Pen-based Hercules virtual appliances could use VRRP (Virtual Router Redundancy Protocol) so that, if the main load balancer fails, another in the group automatically starts handling connections.

Windows Network Load Balancing provides a *low-cost* solution. Windows Network Load Balancing, included with Windows 2003 Server, is available on the servers where the VDM Security Server or VDM Connection

Server is installed. WNLB is designed in such a way that there is no single point of failure. Other load balancing solutions may provide other ways to eliminate any single points of failure.

Finally, a *fault tolerant* load balancing solution needs to be able to remove failed VDM Connection Server nodes from the load-balanced group. The method used to detect failed nodes may vary from solution to solution; however, regardless of the method used to remove or blacklist an unresponsive VDM Connection Server, the solution should ultimately ensure that new incoming requests are not directed to the unresponsive server.

If a VDM Connection Server fails or becomes unresponsive during an active established session, the user does not lose any data; the desktop state is preserved in the virtual desktop instance. When the user is reconnected to a different VDM Connection Server in the group, the desktop session continues exactly where it left off when the failure occurred. In a LAN deployment, where Direct Connect may be used, users are unaffected by any VDM Connection Server disruption because their session is established directly with the virtual desktop.

Reference Example 1: Load-balanced Configuration Using Hercules

Hercules is a Linux-based virtual appliance that can be downloaded from the VMTN Appliances Directory (see “References” on page 12). It is a stripped-down version of Linux that runs in 32 MB of memory and uses a very small virtual hard drive. The load balancing component is provided by the mature, open-source project Pen, which provides TCP-level load balancing, working as proxy. When a TCP session is established between a VDM client and Pen, Pen establishes a mapped session to one of the load-balanced VDM servers, based on the configuration file provided by the administrator. Pen also supports session affinity or stickiness: originating IP information is hashed and used to determine which server to connect to so that all requests from the same client go to the same server. Pen also supports failover on the configured port level. If Pen cannot establish the outgoing connection to one of the servers during configuration time, the server is removed from the list, and the connection is forwarded to another server without timing out.

Hercules and Pen are very easy to use. Since the memory and processor footprint of the virtual appliance is small, you can run it using the free VMware Player on one (or two) of the VDM Security Servers inside the DMZ (see Figure 2). To load balance VDM Connection Servers inside the LAN, run the virtual appliance on VI3 as a virtual appliance to handle client connection requests made directly with the VDM Connection Servers (See Figure 1). For Hercules-based virtual appliances, modify the included Pen service startup configuration file, `/etc/init.d/pen`. Hercules includes the vi text editor, so configuration changes can be made directly from the appliance. For example:

```
#!/bin/sh
#
# startup script for pen
# pchaganti@gmail.com

LOGFILE=/var/log/pen.log
PIDFILE=/var/run/pen.pid
CONTROLPORT=8888
CHROOTDIR=/chroot/pen
LBSERVER=192.168.1.101:https
SERVER1=192.168.1.102:https
SERVER2=192.168.1.103:https

case "$1" in
    start)
        if [ -x /bin/pen ] ; then
            echo -n "Starting pen: "
            /bin/pen -C $CONTROLPORT -X -l $LOGFILE -p $PIDFILE 443 $SERVER1 $SERVER2
            echo "OK"
        fi
        ;;
    stop)
        kill `cat /var/run/pen.pid`
        ;;
    *)
        echo "usage: $0 { start | stop }" >&2

```

```

    exit 1
    ;;
esac

```

In this case, the load balancing appliance was configured to use 192.168.1.101 (the address of the network adapter eth0 of the virtual appliance). Two VDM Connection Servers were added, using 192.168.102 and 103, with the HTTPS protocol. You need to add the domain names for the Hercules appliance and the VDM Connection Servers to your DNS servers separately and verify that all three addresses are accessible through any existing firewalls using HTTPS and the VDM client or VDM WebAccess.

This configuration provides a load-balanced connection to the VDM Connection Servers using sticky sessions. In addition, an automatic failover is configured. During session establishment, if the server does not reply within five seconds, it will be redirected. If you wish to eliminate the load balancer as a single point of failure, you should also configure Virtual Router Redundancy Protocol (VRRP). When VRRP is configured, if a master Hercules appliance fails to serve the incoming client request, the backup takes over. When using VRRP, configure a global DNS name and IP address by modifying the `/etc/init.d/pen` file. The following example configuration file enables VRRPD support.

```

#!/bin/sh
#
# startup script for pen
# pchaganti@gmail.com

LOGFILE=/var/log/pen.log
PIDFILE=/var/run/pen.pid
CONTROLPORT=8888
CHROOTDIR=/chroot/pen
VSERVER=192.168.1.100
LBSERVER=192.168.1.101:https
SERVER1=192.168.1.102:https
SERVER2=192.168.1.103:https

case "$1" in
    start)
        if [ -x /bin/pen ] ; then
            echo -n "Starting pen: "
            /bin/pen -C $CONTROLPORT -X -l $LOGFILE -p $PIDFILE 443 $LBSERVER $SERVER1 $SERVER2
            /sbin/vrrpd -n -i eth0 -v 1 $VSERVER
            echo "OK"
        fi
        ;;
    stop)
        kill `cat /var/run/pen.pid`
        ;;
    *)
        echo "usage: $0 { start | stop }" >&2
        exit 1
        ;;
esac

```

This configuration is an example uses a proxy-type load balancer appropriate for testing and small deployments. The next section describes a load-balanced configuration using the Windows Network Load Balancing service, an example of a commercial product more appropriate for production deployments.

Reference Example 2: Windows Network Load Balancing

The Windows Network Load Balancing Service is available with Windows 2003. It can be configured to use MultiCast or UniCast, each of which has its pros and cons. One benefit of using MultiCast is the need for only one NIC per server. During the configuration, a shared cluster IP address is established on each computer in the cluster. Client requests sent to the cluster IP address are received by all servers in the cluster, but only one responds based on the load balancing algorithm. If a server becomes unresponsive, it is removed from the cluster, thus providing a failover mechanism. NLB supports stickiness with a configuration parameter called *affinity - single*, based on the IP address. When single affinity is enabled, all requests originating from a single IP address are guaranteed to land on the same server.

When configuring NLB, enable the Network Load Balancing service in the properties of the network adapter and configure its settings, either by configuring each VDM server or by using the Network Load Balancing Manager MMC utility from a central location. Creating and managing the cluster using the Network Load Balancing Manager can help reduce human error during the configuration. When you create a cluster using the NLBM and add servers to the cluster, the NLBM enables the NLB service on each server and configures the cluster information. Use the same shared cluster IP address on all load-balanced servers and register a single domain name for this address. Once fully configured, each host will use the shared cluster IP address as well as a unique private IP address (see [Figure 3](#)).

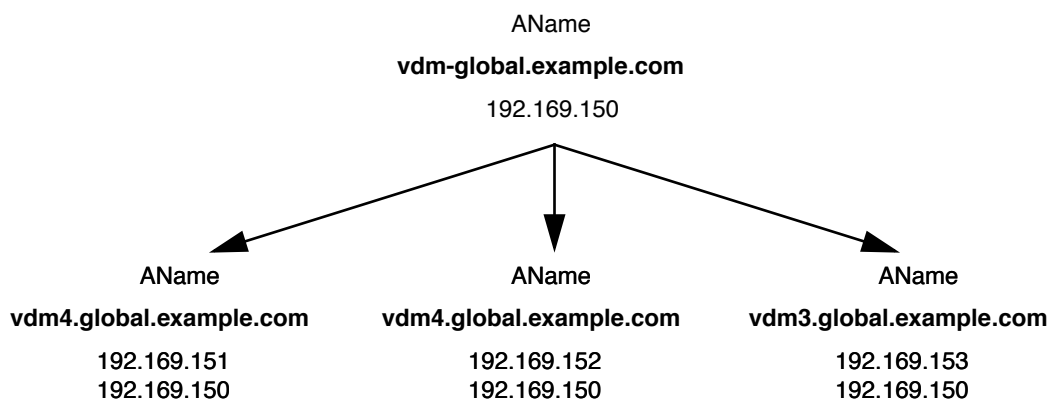


Figure 3. Example NLB DNS Configuration

VDM WebAccess or VDM Clients establish an initial connection using this global load-balanced cluster DNS name. Client requests are then directed to establish an SSL tunnel to the respective VDM Connection Server.

VDM Connection Servers must have an additional, dedicated IP address and registered DNS name. The cluster IP address can either share the same physical adapter or use separate adapters. Using two adapters allows the load-balanced traffic to be separated from the outbound traffic. For complete configuration information, see the Network Load Balancing documentation available from Microsoft Technet.

When using NLB, configure load balancing across all the participating hosts for HTTPS port 443. (Load balancing across multiple hosts with an equal load is advised.) Configure single affinity to support sticky sessions and Web-based administration to configure VDM Connection Servers. The NLB parameters should be identical on all VDM Connection Servers, except for host priority and the dedicated address setting if using a single network adapter in multicast mode. The host with a priority of 1 receives all traffic in the cluster, for which no rules are defined. Sample configuration screen shots are shown in [Figure 4](#) through [Figure 9](#).

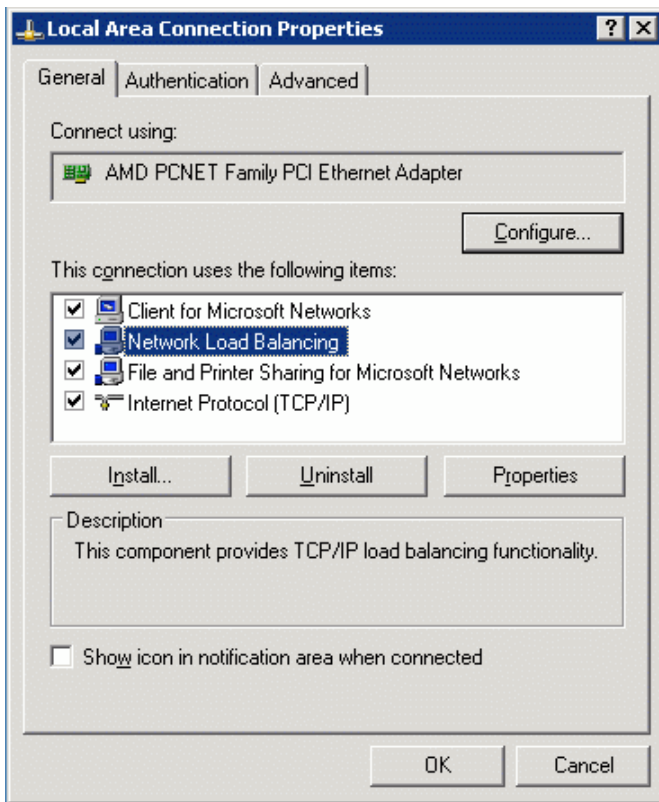


Figure 4. Enabling Network Load Balancing

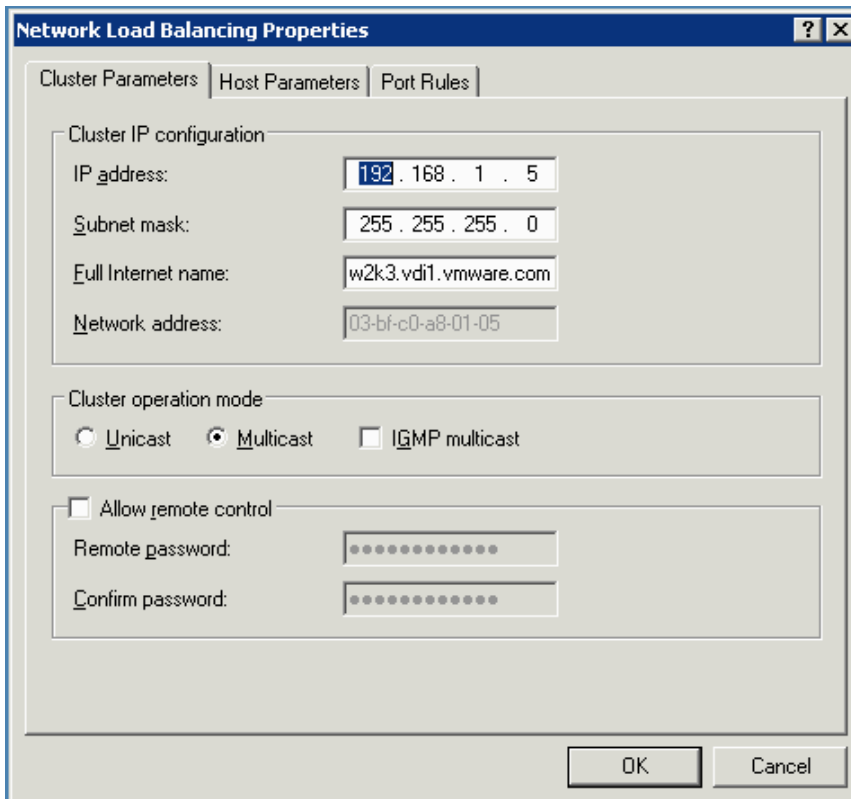


Figure 5. Configuring the Cluster IP Address

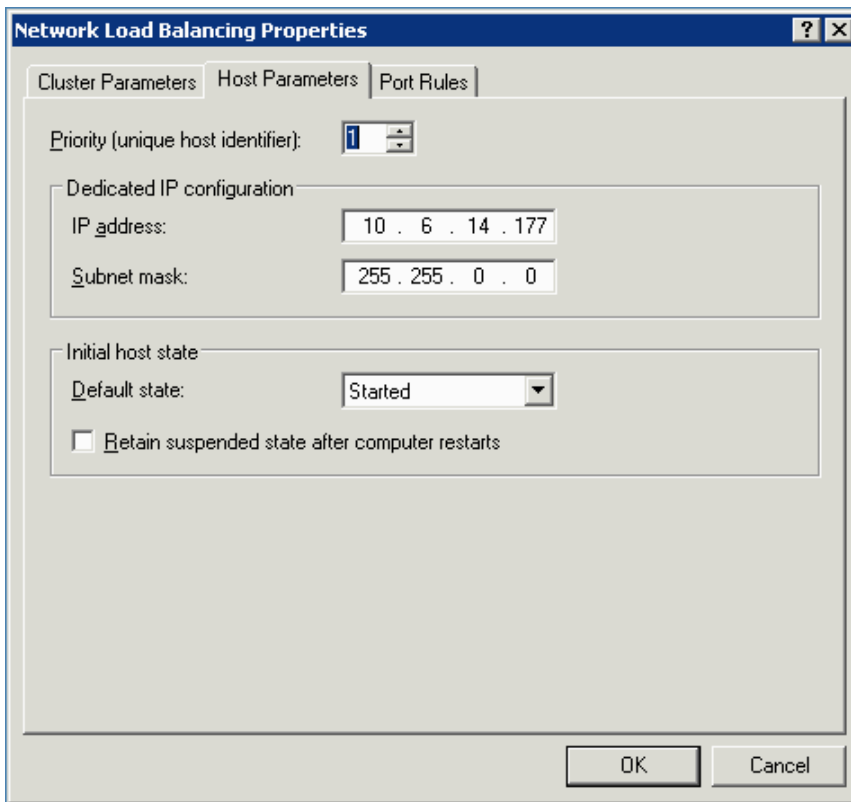


Figure 6. Host Parameters — Dedicated IP Address

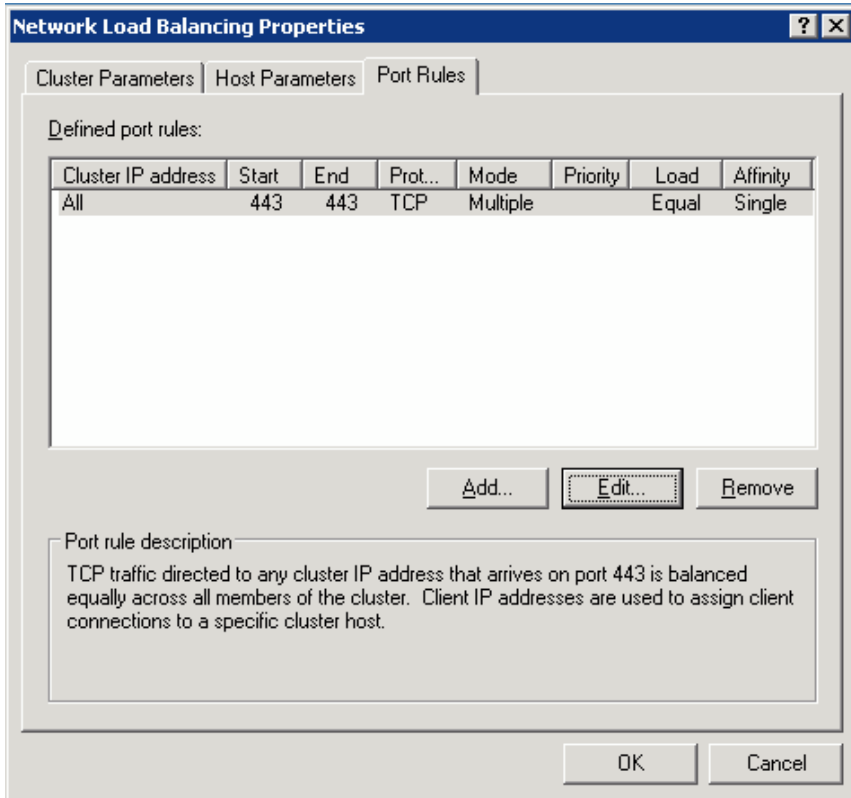


Figure 7. Port Rules

NLB cannot tell if the application (VDM Connection Server or VMware VDMDS service) has failed or is not running on the load-balanced server. NLB detects failures using a heartbeat: a server that responds to heartbeat requests is still sent HTTPS requests even when the VMware VDM Connection Server or VMware VDMDS services have failed. This causes new client requests to time out, and sessions already assigned by the load balancing algorithm associated with this server continue to be directed to this server as they attempt to reconnect.

To work around this problem, stop the NLB service when the VMware VDM Connection Server or VMware VDMDS service is not running, then restart the NLB service once the Connection Server or VDMDS has been restored. You can automate this sequence by configuring the recovery options for both the VMware Connection Server and VMware VDMDS services using the `wlbs.exe` stop and start commands (see [Figure 8](#)). In a more advanced setup, you might use scripting on the server to start and top the NLB service.

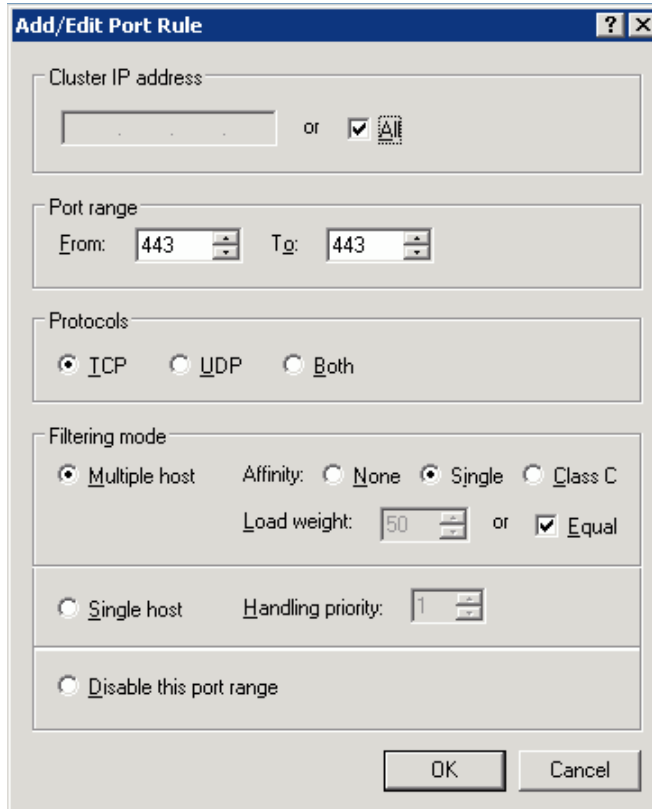


Figure 8. Configuration of the Port Rule

NLB cannot tell if the application (VDM Connection Server or VMware VDMDS service) has failed or is not running on the load-balanced server. NLB detects failures using a heartbeat: a server that responds to heartbeat requests is still sent HTTPS requests even when the VMware VDM Connection Server or VMware VDMDS services have failed. This causes new client requests to time out, and sessions already assigned by the load balancing algorithm associated with this server continue to be directed to this server as they attempt to reconnect.

To work around this problem, stop the NLB service when the VMware VDM Connection Server or VMware VDMDS service is not running, then restart the NLB service once the Connection Server or VDMDS has been restored. You can automate this sequence by configuring the recovery options for both the VMware Connection Server and VMware VDMDS services using the `wlbs.exe` stop and start commands (see [Figure 8](#)). In a more advanced setup, you might use scripting on the server to start and top the NLB service.

In a DMZ setup, (see [Figure 2](#)), VDM Security Servers act as a proxy for the VDM Connection Servers running inside the protected network. If the VDM Connection Server fails and the Security Server is still running, NLB cannot detect this condition, even with the service recovery settings described above. In this case, the configuration can be more fault tolerant if another application or script running on the VDM Security Server

periodically polls the logon page on the server. If it cannot load the page, it can remove the associated VDM Security Server from the NLB cluster.

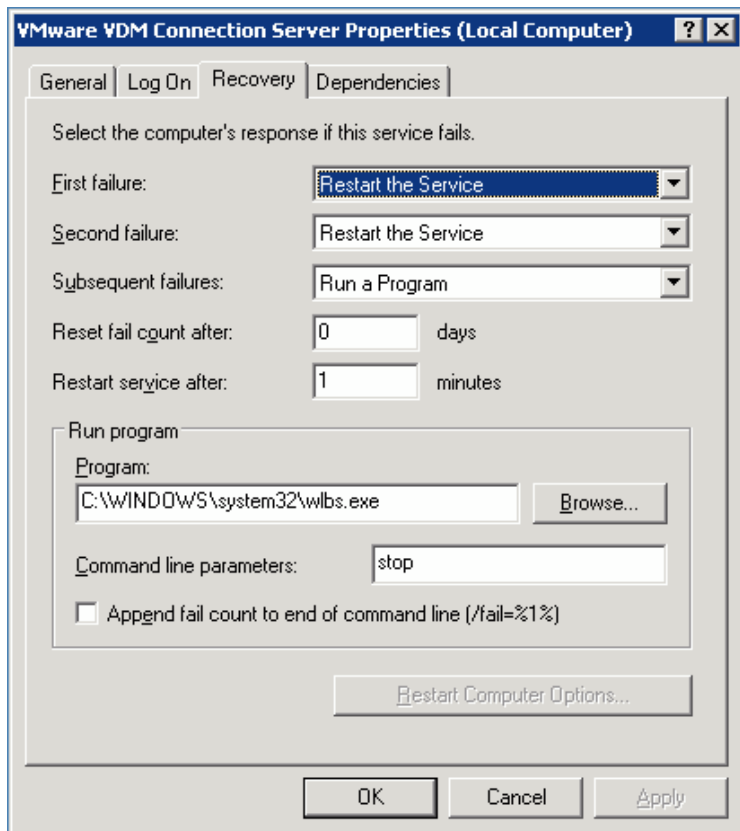


Figure 9. Stopping and Starting the NLB Service in the Event of a Failure

Additional Considerations

The reference configurations provided above are just references. Every production environment will vary based on needs and requirements.

More advanced commercial load balancing solutions, such as those offered from F5, Foundry, Cisco, Nortel, or others, may provide the ability to query the status of the VDM Connection Servers at the application layer. If connection requests are unsuccessful or other advanced algorithms are being used to make load balancing decisions, the load balancer can adjust its decisions accordingly. For example, the Zeus commercial software load balancer allows configuration of an HTTP monitor, which queries the URL of the load-balanced applications and applies a regular expression to the retrieved results. If the results do not match, the server can be removed from the list or the load reduced (see [Figure 10](#)). In the event of a failure or change, an email is sent to the administrator. This functionality can be used to manage a group of load-balanced servers more efficiently: simply checking the logon page can be sufficient to determine whether the VDM Connection Server is running.

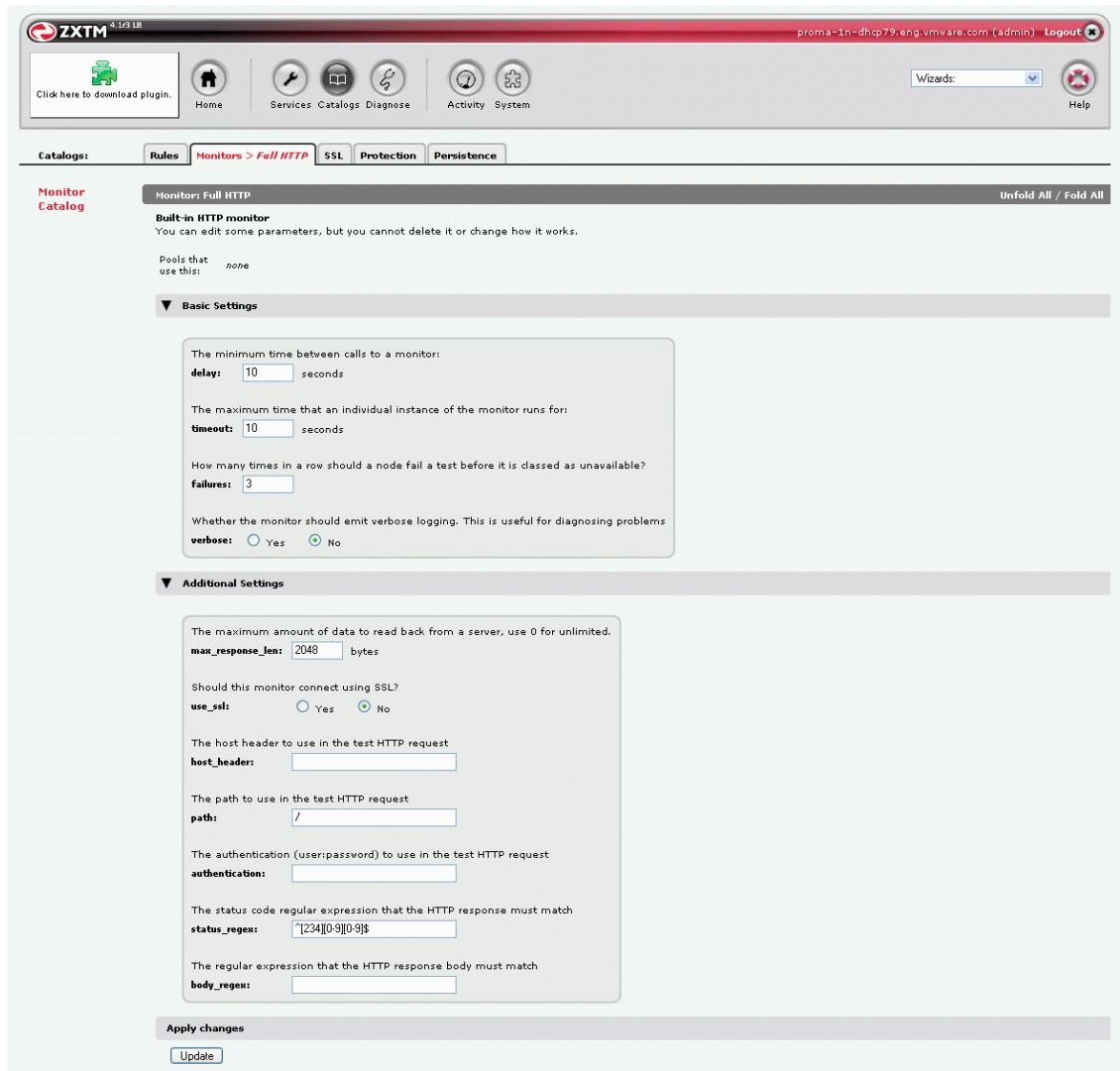


Figure 10. Zeus HTTP Monitor Settings

Conclusion

You can easily leverage free or low-cost commercial load balancing software to provide additional scalability and fault tolerance in any VDM 2-based solution. As with any complex configuration, however, it is always best to give careful consideration to your design requirements up front.

About the Authors

Alexey Orlovsky is a Senior Member of Technical Staff at VMware.

Warren Ponder is a Senior Technical Marketing Engineer at VMware.

Acknowledgements

The authors would like to thank, Tommy Armstrong, Frank Taylor, Anthony Wilkinson, Mark Benson, Chris Duffy, Nick Gibson, and Ian Gibbs for their contributions.

References

Microsoft – Network Load Balancing Clusters

<http://technet2.microsoft.com/windowsserver/en/library/750d3a40-af67-411d-828b-fc1f718a06fb1033.mspx?mfr=true>

Microsoft – *Planning Your Network Load Balancing Configuration*

<http://technet2.microsoft.com/windowsserver/en/library/750d3a40-af67-411d-828b-fc1f718a06fb1033.mspx?mfr=true>

Pen – *Pen Load Balancing*

<http://siag.nu/pen/>

Hercules – *Hercules Load Balancer Virtual Appliance*

<http://www.vmware.com/appliances/directory/300>

Zeus – *Zeus Load Balancing*

<http://www.zeus.com/products/zxtmlb/>