

# Sun™ N2000 Series Secure Application Switch: Layer 4 performance test

*Test report prepared under contract from Sun Microsystems*

## Executive summary

Sun Microsystems commissioned VeriTest to measure the Layer 4 performance of their Sun N2000 Series Secure Application Switch. Sun supplied a N2120 switch running version V2\_0R1 firmware and configured with 12 small form factor Gigabit Ethernet ports and one function card. The function card adds support for both hardware-based Layer 4-7 application switching and SSL acceleration.

### Key findings

- The Sun N2000 Series switch averaged 208,424 Layer 4 connections/second in our test configuration.

Sun requested that VeriTest measure the maximum number of Layer 4 HTTP 1.0 connections/second supported by the Sun N2000 Series switch. For the test, the Sun N2000 Series switch was configured as a server load balancer positioned between clients generating HTTP requests and Web servers responding to requests.

We used Ziff Davis Media's WebBench 5.0 benchmark to generate the HTTP request load for the test. WebBench measures Web server throughput performance by using physical test clients to generate an HTTP based workload against a Web server under test. WebBench reports test results in HTTP requests/second and throughput in bytes per second. Please refer to the Test Methodology section for a detailed description of the test methodology. We performed two separate runs of each test to verify consistent results.

Figure 1 shows the results of each individual test run as well as their average for the Layer 4 connections/second test.

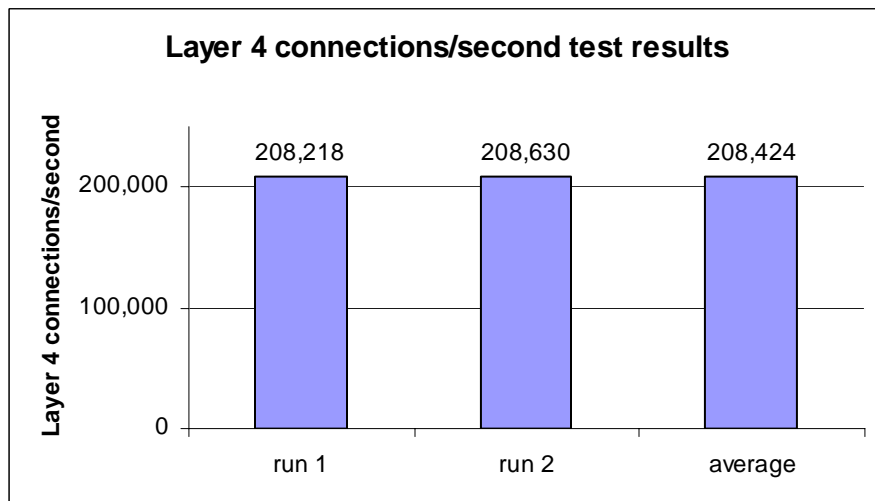


Figure 1: Sun N2000 Series Layer 4 connections/second test results

We created a custom WebBench test suite that sent HTTP 1.0 requests for a 223 byte static object from the WebBench clients to four virtual IP addresses configured on the Sun N2000 Series switch. For each request, the Sun N2000 Series switch received the HTTP request from the client, load balanced the request across a pool of Web servers, and forwarded the response from the Web server to the client.

The Sun N2000 Series switch averaged 208,424 Layer 4 connections/second in our test configuration.

## Testing methodology

Sun Microsystems commissioned VeriTest to measure the maximum number of Layer 4 connections/second supported on their Sun N2000 Series Secure Application Switch. Sun supplied a N2120 switch running version V2\_0R1 firmware and configured with 12 small form factor Gigabit Ethernet ports and one function card. The function card adds support for both hardware-based Layer 4-7 application switching and SSL acceleration.

For the Layer 4 performance test, the Sun N2000 Series switch was configured as a load balancer positioned between clients generating HTTP requests and Web servers responding to those requests. The Sun N2000 Series switch received HTTP requests from the clients, load balanced the requests across the pool of Web servers, and forwarded the responses from the Web servers to the clients.

We used Ziff Davis Media's WebBench 5.0 benchmark to generate the HTTP request load for the test. WebBench measures Web server throughput performance by using physical test clients to generate an HTTP based workload against a Web server under test. As the Web server under test responds to the client requests, each WebBench client records the number of HTTP requests made and the amount of data moved during the test. Once a test completes, WebBench reports test results in HTTP requests/second and throughput in bytes per second. For the Layer 4 connections/second test, we created a custom WebBench test suite that sent HTTP 1.0 requests for a 223 byte static object from the WebBench clients to four separate virtual IP addresses configured on the Sun N2000 Series switch.

We used 480 Dell PowerEdge 350 systems running a mixture of Windows 2000 Professional SP3 (150 systems) Windows 2000 Professional SP4 (120 systems) and Windows XP Professional SP1 (210 systems) as WebBench clients generating the HTTP request load. The client systems were spread across eight test beds with 60 clients in each test bed. Within each test bed, the 60 clients were evenly distributed between two Extreme Networks Summit48 switches. The network interfaces on the client systems and the ports on the Summit48 switches were set to auto negotiate resulting in 100Mbps full-duplex links. We configured the eight test beds into four groups of two, with the switches in each group connected by Fiber Gigabit connections. We configured the clients in each group into different IP subnets. We connected each of the four groups to the Sun N2000 Series switch with a Fiber Gigabit connection resulting in four Fiber Gigabit connections between the client test beds and the Sun N2000 Series switch. We set the gateway address on the client systems to IP routing interfaces defined on the Sun N2000 Series switch. Finally, we configured the WebBench clients so that each group sent HTTP requests to a different virtual IP address on the switch.

We used a Gateway E-4600 system running Windows XP Professional SP1 as the WebBench 5.0 controller. We installed an Intel PRO/1000 MF Server Adapter in the controller and connected it to a port on the Sun N2000 Series switch. We enabled auto negotiation on the Intel adapter and the Sun N2000 Series switch port.

We used 60 Dell PowerEdge 350 systems as the Web servers for the test. Each system ran Windows 2000 Server SP4 with updated patches and Internet Information Server (IIS) 5.0 configured as the Web Server. Using the Microsoft Management Console Internet Services plug-in, we made the following changes to the default IIS 5.0 configuration on each of the Web servers:

- On the WebSite tab, we disabled logging.
- On the Performance tab, we set the IIS performance parameter to 100,000+ hits per day.
- On the Home Directory tab, we set Execute Permissions to None for the document root directory.

- On the Home Directory tab, we disabled the “Index This Resource” property for the document root directory.
- On the Home Directory tab, we set the Application Protection to “Low (IIS Process)” for the document root directory.

Additionally, we made the following registry modifications to each of the Web servers:

- Set HKLM\System\CurrentControlSet\Services\InetInfo\Parameters\ObjectCacheTTL to 7200 seconds
- Set HKLM\System\CurrentControlSet\Services\InetInfo\Parameters\MaxCachedFileSize to 600000 bytes.

We connected the Web servers to two Extreme Networks Summit48 switches with 30 servers connected to each switch. The network interfaces on the Web servers and the ports on the Summit48 switches were set to auto negotiate resulting in 100Mbps full-duplex links. We configured each Summit48 switch with two VLANs. Each VLAN contained ports connected to 15 Web servers as well as one Fiber Gigabit port. We configured the 15 Web servers in each VLAN into separate IP subnets. We connected the Fiber Gigabit port from each of the four VLANs to the Sun N2000 Series switch resulting in four Fiber Gigabit connections between the Web servers and the Sun N2000 Series switch. We set the gateway address on the Web servers to IP routing interfaces defined on the Sun N2000 Series switch. We copied a 233 byte file to the document root on each Web server and used this as the content for each HTTP request.

Figure 2 illustrates the test bed configuration used for the Layer 4 connections/second test.

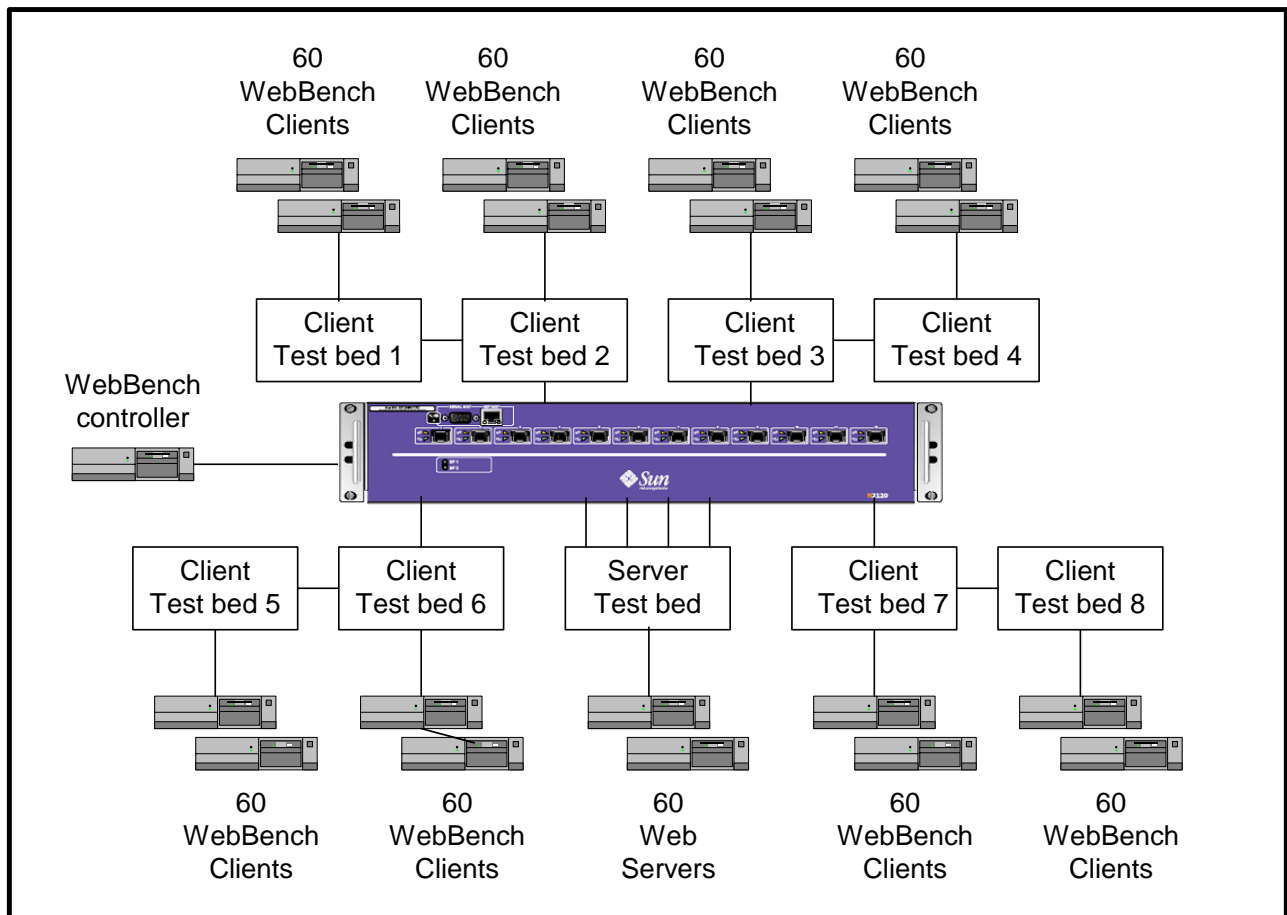


Figure 2: Layer 4 connections/second test bed

Figure 3 shows the port connections on the Sun N2000 Series switch for the Layer 4 connections/second test. All ports on the Sun N2000 Series switch were set to auto negotiate link speed and duplex settings.

Port	Connection
1	Client group 1 (120 WebBench clients)
2	Web Server group 1 (15 Web servers)
3	Client group 4 (120 WebBench clients)
4	unused
5	Client group 2 (120 WebBench clients)
6	Web Server group 2 (15 Web servers)
7	Web Server group 4 (15 Web servers)
8	unused
9	Client group 3 (120 WebBench clients)
10	Web Server group 3 (15 Web servers)
11	unused
12	WebBench 5.0 controller

**Figure 3: Sun N2000 Series port connections for the Layer 4 connections/second test**

An on-site Sun engineer configured the Sun N2000 Series switch for the test. A VeriTest engineer performed the actual test. The Sun N2000 Series switch configuration specified four service groups. Each service group contained 15 Web servers. Three service groups used the weighted hash load balance type and one service group used the round robin load balance type to distribute HTTP requests among the Web servers in the group. Health checking was disabled for all of the service groups. The configuration included four virtual services. Three virtual services had the L4SLB service type and the fourth virtual service had the L4SLB\_ADVANCED service type. Each virtual service mapped to a unique service group. Each virtual service also defined a virtual IP address accessed by the WebBench clients (a total of four distinct virtual IP addresses). The switch configuration enabled IP routing, which allowed the Sun N2000 Series switch to route packets between all of the client subnets.

Figure 4 lists a summary of the Sun N2000 Series switch configuration for the Layer 4 connections/second test. Refer to Appendix B for a listing of the switch configuration file used for the test.

Switch Setting	Value
LoadBalance->serviceGroup->In Line Health Check	Disabled
LoadBalance->virtualService->Advanced->SMM Stream Limit (HTTP)	8xRcvWindow
Resource->serviceBandwidth->Service Percent	100

**Figure 4: Sun N2000 Series switch configuration changes for the Layer 4 connections/second test**

We created a custom WebBench 5.0 test suite that generated HTTP 1.0 requests for a 223 byte static object located on each Web server. We configured all of the clients in each of the four test bed groups to access a different virtual IP address defined on the Sun N2000 Series switch. This evenly distributed the client's HTTP requests across the four virtual IP addresses. The test suite contained one mix that included all 480 WebBench clients in the test. We set the ramp up time and the delay time to 60 seconds. This staggered the start time of the clients for the first 60 seconds of the test. We configured a 10 second ramp down period and an overall test length of 10 minutes. We set the engines per client to 1, the thread inactivity timeout to 120 seconds, and modified the workload file to specify the URL of the 223 byte file located on the Web servers. We manually edited the test suite file and set the WB\_Threads parameter to 20 for the clients mapped to the L4SLB virtual groups and 50 for the clients mapped to the L4SLB\_ADVANCED virtual group.

We performed two separate runs of the test to verify consistent results. Refer to Appendix A for detailed information on the hardware and software used during the tests.

## Appendix

### A. Test Hardware and Software

Sun N2000 Series switch	
Model	N2120
Software version	V2_0R1
Number of Gigabit ports	12 small form factor fiber Gigabit ports
Number of function cards	1

Figure 5: Sun N2000 Series switch

WebBench clients (480 identical systems)	
Machine Type	Dell PowerEdge 350
BIOS	TR440BXA.86B.0034.P12
Processor	850MHz Pentium III
L2 Cache	256KB
Expansion Bus	PCI
Memory	256MB
Disk(s)	10GB Maxtor ST010H1, Microsoft driver version: 5.0.2183.1 (W2K), Microsoft driver version 5.1.2535.0 (XP)
Network Adapter(s)	2 embedded Intel PRO 100+, Intel driver version 8.0.11.0 (W2K), Microsoft driver version 5.41.22.0 (XP)
OS	Windows 2000 Professional SP4 (120 systems) Windows 2000 Professional SP3 (150 systems) Windows XP Professional SP1 (210 systems)

Figure 6: WebBench client systems

Web servers (60 identical systems)	
Machine Type	Dell PowerEdge 350
BIOS	TR440BXA.86B.0034.P12
Processor	850MHz Pentium III
L2 Cache	256KB
Expansion Bus	PCI
Memory	256MB
Disk(s)	10GB Maxtor ST010H1, Microsoft driver version 5.0.2183.1
Network Adapter(s)	2 embedded Intel PRO 100+, Intel driver version 8.0.11.0
OS	Windows 2000 Server SP4

Figure 7: Web server systems

WebBench Controller	
Machine Type	Gateway E-4600
BIOS	GB85010A.15A.0046.P13.0108201551 8/20/2001
Processor	1.3GHz Pentium 4
L2 Cache	256KB
Expansion Bus	PCI
Memory	256MB
Disk(s)	WDC WD400BB-53AUA1, Microsoft driver version 5.1.2535.0
Network Adapter(s)	Intel PRO/100+, Intel driver version 7.1.8.0 Intel PRO/1000 MF Server Adapter, Intel driver version 8.0.57.0
OS	Windows XP Professional
OS Updates	Service Pack 1

Figure 8: WebBench controller system

## B. Sun N2000 Series switch configuration file

```

commandModeEntry on
config
#
# Event configuration and statistics
#
event
#
# Profile rules for event filters
#
filterProfile defaultFile {default filter for saving to file}
filterProfile defaultFile
#
# Profiles to cause event filtering
#
rule 130 drop logLevel warning
rule 200 send true
exit;
filterProfile defaultLog {default log filter}
filterProfile defaultLog
#
# Profiles to cause event filtering
#
rule 90 drop logLevel debug
rule 100 send true
exit;
filterProfile defaultSyslog {default syslog filter}
filterProfile defaultSyslog
#
# Profiles to cause event filtering
#
rule 90 drop logLevel debug
rule 100 send true
exit;
filterProfile defaultTrapd {default trapd filter}
filterProfile defaultTrapd
#
# Profiles to cause event filtering
#
rule 90 drop logLevel warning
rule 100 send true
exit;
exit;
#
# NMON state and status
#
nmon

```

```

exit;
#
# Port configuration
#
port eth.1.1 normal auto fullDuplex disabled 1000M fullDuplex discard
port eth.1.1
exit;
port eth.1.2 normal auto fullDuplex disabled 1000M fullDuplex discard
port eth.1.2
exit;
port eth.1.3 normal auto fullDuplex disabled 1000M fullDuplex discard
port eth.1.3
exit;
port eth.1.4 normal auto fullDuplex disabled 1000M fullDuplex discard
port eth.1.4
exit;
port eth.1.5 normal auto fullDuplex disabled 1000M fullDuplex discard
port eth.1.5
exit;
port eth.1.6 normal auto fullDuplex disabled 1000M fullDuplex discard
port eth.1.6
exit;
port eth.1.7 normal auto fullDuplex disabled 1000M fullDuplex discard
port eth.1.7
exit;
port eth.1.8 normal auto fullDuplex disabled 1000M fullDuplex discard
port eth.1.8
exit;
port eth.1.9 normal auto fullDuplex disabled 1000M fullDuplex discard
port eth.1.9
exit;
port eth.1.10 normal auto fullDuplex disabled 1000M fullDuplex discard
port eth.1.10
exit;
port eth.1.11 normal auto fullDuplex disabled 1000M fullDuplex discard
port eth.1.11
exit;
port eth.1.12 normal auto fullDuplex disabled 1000M fullDuplex discard
port eth.1.12
exit;
#
# vSwitch configuration
#
vSwitch Veritest
#
# Host configuration
#
loadBalance host testbed1-1 192.168.1.1 vRouter Veritest:default
loadBalance host testbed1-2 192.168.1.2 vRouter Veritest:default
loadBalance host testbed1-3 192.168.1.3 vRouter Veritest:default

<repetitive entries for hosts 4 – 14 removed>

loadBalance host testbed1-15 192.168.1.15 vRouter Veritest:default
loadBalance host testbed2-16 192.168.2.16 vRouter Veritest:default
loadBalance host testbed2-17 192.168.2.17 vRouter Veritest:default
loadBalance host testbed2-18 192.168.2.18 vRouter Veritest:default

<repetitive entries for hosts 19 – 29 removed>

loadBalance host testbed2-30 192.168.2.30 vRouter Veritest:default
loadBalance host testbed3-31 192.168.3.31 vRouter Veritest:default
loadBalance host testbed3-32 192.168.3.32 vRouter Veritest:default
loadBalance host testbed3-33 192.168.3.33 vRouter Veritest:default

<repetitive entries for hosts 34 – 44 removed>

loadBalance host testbed3-45 192.168.3.45 vRouter Veritest:default
loadBalance host testbed4-46 192.168.4.46 vRouter Veritest:default
loadBalance host testbed4-47 192.168.4.47 vRouter Veritest:default
loadBalance host testbed4-48 192.168.4.48 vRouter Veritest:default

```

<repetitive entries for hosts 49 – 59 removed>

```
loadBalance host testbed4-60 192.168.4.60 vRouter Veritest:default
#
# Real service parameters
#
loadBalance realService rs-testbed1-1 testbed1-1 certType Literal
loadBalance realService rs-testbed1-1
exit; exit;
loadBalance realService rs-testbed1-2 testbed1-2 certType Literal
loadBalance realService rs-testbed1-2
exit; exit;
loadBalance realService rs-testbed1-3 testbed1-3 certType Literal
loadBalance realService rs-testbed1-3
exit; exit;
```

<repetitive entries for real services 4 – 14 removed>

```
loadBalance realService rs-testbed1-15 testbed1-15 certType Literal
loadBalance realService rs-testbed1-15
exit; exit;
loadBalance realService rs-testbed2-16 testbed2-16 certType Literal
loadBalance realService rs-testbed2-16
exit; exit;
loadBalance realService rs-testbed2-17 testbed2-17 certType Literal
loadBalance realService rs-testbed2-17
exit; exit;
loadBalance realService rs-testbed2-18 testbed2-18 certType Literal
loadBalance realService rs-testbed2-18
exit; exit;
```

<repetitive entries for real services 19 – 30 removed>

```
loadBalance realService rs-testbed2-30 testbed2-30 certType Literal
loadBalance realService rs-testbed2-30
exit; exit;
loadBalance realService rs-testbed3-31 testbed3-31 certType Literal
loadBalance realService rs-testbed3-31
exit; exit;
loadBalance realService rs-testbed3-32 testbed3-32 certType Literal
loadBalance realService rs-testbed3-32
exit; exit;
loadBalance realService rs-testbed3-33 testbed3-33 certType Literal
loadBalance realService rs-testbed3-33
exit; exit;
```

<repetitive entries for real services 34 – 44 removed>

```
loadBalance realService rs-testbed3-45 testbed3-45 certType Literal
loadBalance realService rs-testbed3-45
exit; exit;
loadBalance realService rs-testbed4-46 testbed4-46 certType Literal
loadBalance realService rs-testbed4-46
exit; exit;
loadBalance realService rs-testbed4-47 testbed4-47 certType Literal
loadBalance realService rs-testbed4-47
exit; exit;
loadBalance realService rs-testbed4-48 testbed4-48 certType Literal
loadBalance realService rs-testbed4-48
exit; exit;
```

<repetitive entries for real services 49 – 59 removed>

```
loadBalance realService rs-testbed4-60 testbed4-60 certType Literal
loadBalance realService rs-testbed4-60
exit; exit;
#
# Service group configuration
#
loadBalance serviceGroup SG-testbed1 weightedHash \
```

```

    {rs-testbed1-1; rs-testbed1-10; rs-testbed1-11; rs-testbed1-12; rs-testbed1-13; rs-testbed1-14; rs-testbed1-15; rs-testbed1-2; rs-
testbed1-3; rs-testbed1-4; rs-testbed1-5; rs-testbed1-6; rs-testbed1-7; rs-testbed1-8; rs-testbed1-9} \
    inlineHealthCheck disabled
    loadBalance serviceGroup SG-testbed1
    exit; exit;
    loadBalance serviceGroup SG-testbed2 weightedHash \
    {rs-testbed2-16; rs-testbed2-17; rs-testbed2-18; rs-testbed2-19; rs-testbed2-20; rs-testbed2-21; rs-testbed2-22; rs-testbed2-23; rs-
testbed2-24; rs-testbed2-25; rs-testbed2-26; rs-testbed2-27; rs-testbed2-28; rs-testbed2-29; rs-testbed2-30} \
    inlineHealthCheck disabled
    loadBalance serviceGroup SG-testbed2
    exit; exit;
    loadBalance serviceGroup SG-testbed3 weightedHash \
    {rs-testbed3-31; rs-testbed3-32; rs-testbed3-33; rs-testbed3-34; rs-testbed3-35; rs-testbed3-36; rs-testbed3-37; rs-testbed3-38; rs-
testbed3-39; rs-testbed3-40; rs-testbed3-41; rs-testbed3-42; rs-testbed3-43; rs-testbed3-44; rs-testbed3-45} \
    inlineHealthCheck disabled
    loadBalance serviceGroup SG-testbed3
    exit; exit;
    loadBalance serviceGroup SG-testbed4 roundRobin \
    {rs-testbed4-46; rs-testbed4-47; rs-testbed4-48; rs-testbed4-49; rs-testbed4-50; rs-testbed4-51; rs-testbed4-52; rs-testbed4-53; rs-
testbed4-54; rs-testbed4-55; rs-testbed4-56; rs-testbed4-57; rs-testbed4-58; rs-testbed4-59; rs-testbed4-60} \
    inlineHealthCheck disabled
    loadBalance serviceGroup SG-testbed4
    exit; exit;
    #
    # Virtual Service configuration
    #
    loadBalance virtualService VS-Testbed1 L4SLB 10.0.1.250 SG-testbed1
    loadBalance virtualService VS-Testbed1
    exit; exit;
    loadBalance virtualService VS-Testbed2 L4SLB 10.0.2.250 SG-testbed2
    loadBalance virtualService VS-Testbed2
    exit; exit;
    loadBalance virtualService VS-Testbed3 L4SLB 10.0.3.250 SG-testbed3
    loadBalance virtualService VS-Testbed3
    exit; exit;
    loadBalance virtualService VS-Testbed4 L4SLB_ADV 10.0.4.250 SG-testbed4
    loadBalance virtualService VS-Testbed4
    #
    # Virtual service advanced settings
    #
    advanced smmStreamLimit 8xRcvWnd initParseWithData true rxUseLongTime true clientFirstProtocol true
    exit; exit;
    #
    # Port Bandwidth configuration
    #
    resource portBandwidth eth.1.1 100 100 65534 65535
    resource portBandwidth eth.1.2 100 100 65534 65535
    resource portBandwidth eth.1.3 100 100 65534 65535
    resource portBandwidth eth.1.5 100 100 65534 65535
    resource portBandwidth eth.1.6 100 100 65534 65535
    resource portBandwidth eth.1.7 100 100 65534 65535
    resource portBandwidth eth.1.9 100 100 65534 65535
    resource portBandwidth eth.1.10 100 100 65534 65535
    resource portBandwidth eth.1.12 100 100 65534 65535
    #
    # Service Engine Bandwidth Configuration
    #
    resource serviceBandwidth functionCard1 100 100
    #
    # vRouter configuration
    #
    vRouter default {Default vRouter}
    vRouter default
    #
    # Display IP layer configuration
    #
    ip forwarding enabled
    ip
    #
    # IP Interfaces
    #

```

```

interface eth.1.2
interface eth.1.2 linkUpDownTrap disabled
interface eth.1.6
interface eth.1.6 linkUpDownTrap disabled
interface eth.1.7
interface eth.1.7 linkUpDownTrap disabled
interface eth.1.10
interface eth.1.10 linkUpDownTrap disabled
#
# IP Interface Address
#
address eth.1.2 192.168.1.254 255.255.255.0
address eth.1.6 192.168.2.254 255.255.255.0
address eth.1.7 192.168.4.254 255.255.255.0
address eth.1.10 192.168.3.254 255.255.255.0
#
# ICMP configuration
#
icmp true true false true
exit;
#
# Interfaces
#
interfaces sock.Veritest:default linkUpDownTrap disabled
interfaces sock.Veritest:default/ip.Veritest:default linkUpDownTrap disabled
interfaces ip.Veritest:default linkUpDownTrap disabled
interfaces ip.Veritest:default/eth.1.2 linkUpDownTrap disabled
interfaces ip.Veritest:default/eth.1.6 linkUpDownTrap disabled
interfaces ip.Veritest:default/eth.1.10 linkUpDownTrap disabled
interfaces ip.Veritest:default/eth.1.7 linkUpDownTrap disabled
interfaces loopback linkUpDownTrap disabled
#
# VRRP configuration
#
vrrp
exit;
exit;
vSwitch system {System vSwitch}
vSwitch system
#
# Port Bandwidth configuration
#
resource portBandwidth eth.1.1 100 100 65534 65535
resource portBandwidth eth.1.3 100 100 65534 65535
resource portBandwidth eth.1.5 100 100 65534 65535
resource portBandwidth eth.1.9 100 100 65534 65535
resource portBandwidth eth.1.12 100 100 65534 65535
#
# vRouter configuration
#
vRouter management {System Management vRouter}
vRouter management
#
# Display IP layer configuration
#
ip forwarding disabled
ip
#
# IP Interfaces
#
interface ethMgmt.1
interface ethMgmt.1 linkUpDownTrap disabled
#
# IP Interface Address
#
address ethMgmt.1 10.10.1.1 255.0.0.0
#
# ICMP configuration
#
icmp true true false true

```

```

#
# TCP Connections
#
tcp connections 0.0.0.0 80 0.0.0.0 0 listen
tcp connections 10.10.1.1 80 10.10.1.2 1453 established
tcp connections 10.10.1.1 80 10.10.1.2 1454 established
exit;
#
# Interfaces
#
interfaces sock.system:management linkUpDownTrap disabled
interfaces sock.system:management/ip.system:management linkUpDownTrap disabled
interfaces ip.system:management linkUpDownTrap disabled
interfaces ip.system:management/ethMgmt.1 linkUpDownTrap disabled
interfaces ethMgmt.1 linkUpDownTrap disabled
interfaces loopback linkUpDownTrap disabled
#
# VRRP configuration
#
vrrp
exit;
exit;
vRouter shared {Shared vRouter}
vRouter shared
#
# Display IP layer configuration
#
ip forwarding enabled
ip
#
# IP Interfaces
#
interface eth.1.1
interface eth.1.1 linkUpDownTrap disabled
interface eth.1.3
interface eth.1.3 linkUpDownTrap disabled
interface eth.1.5
interface eth.1.5 linkUpDownTrap disabled
interface eth.1.9
interface eth.1.9 linkUpDownTrap disabled
interface eth.1.12
interface eth.1.12 linkUpDownTrap disabled
#
# IP Interface Address
#
address eth.1.1 10.0.1.254 255.255.255.0
address eth.1.3 10.0.4.254 255.255.255.0
address eth.1.5 10.0.2.254 255.255.255.0
address eth.1.9 10.0.3.254 255.255.255.0
address eth.1.12 10.0.5.254 255.255.255.0
#
# ICMP configuration
#
icmp true true false true
#
# Static route configuration
#
route static 10.0.1.0 255.255.255.0 10.0.1.254 unspecified
route static 10.0.2.0 255.255.255.0 10.0.2.254 unspecified
route static 10.0.3.0 255.255.255.0 10.0.3.254 unspecified
route static 10.0.4.0 255.255.255.0 10.0.4.254 unspecified
exit;
#
# Interfaces
#
interfaces sock.system:shared linkUpDownTrap disabled
interfaces sock.system:shared/ip.system:shared linkUpDownTrap disabled
interfaces ip.system:shared linkUpDownTrap disabled
interfaces ip.system:shared/eth.1.1 linkUpDownTrap disabled
interfaces ip.system:shared/eth.1.5 linkUpDownTrap disabled
interfaces ip.system:shared/eth.1.9 linkUpDownTrap disabled

```

```

    interfaces ip.system:shared/eth.1.3 linkUpDownTrap disabled
    interfaces ip.system:shared/eth.1.12 linkUpDownTrap disabled
    interfaces loopback linkUpDownTrap disabled
    #
    # VRRP configuration
    #
    vrrp
    exit;
exit;
#
# HTTP configuration and status
#
switchServices httpd enabled
switchServices httpd
exit; exit;
#
# Global NTP configuration parameters
#
switchServices ntp
#
# Global NTP configuration parameters (advanced)
#
advanced
exit;
exit; exit;
#
# SNMP configuration
#
switchServices snmp
#
# SNMP engine configuration
#
systemInfo {} {Sun Application Switch} {}
exit; exit;
#
# Software key
#
switchServices software key {}
#
# SSHd configuration and operation
#
switchServices sshd confEncryption {des3Cbc; blowfishCbc; des} confHmac \
{md5; sha1; md5b96; sha1b96} userAuthentication {publicKey; password}
switchServices sshd
#
# SSHd configuration and operation (advanced)
#
advanced
exit;
exit; exit;
#
# Telnetd configuration and current status
#
switchServices telnetd
exit; exit;
#
# TFTPd configuration and session statistics
#
switchServices tftpd
exit; exit;
#
# TideRunner Configuration
#
switchServices tideRunner initkeys functionCard1 20000 statPollPeriod 5 smmPageSize 2 dleMaxHdrLen 8192
#
# Configuration and status for the trap process
#
switchServices trap
exit; exit;

```

VeriTest ([www.veritest.com](http://www.veritest.com)), the testing division of Lionbridge Technologies, Inc., provides outsourced testing solutions that maximize revenue and reduce costs for our clients. For companies who use high-tech products as well as those who produce them, smoothly functioning technology is essential to business success. VeriTest helps our clients identify and correct technology problems in their products and in their line of business applications by providing the widest range of testing services available.

VeriTest created the suite of industry-standard benchmark software that includes WebBench, NetBench, Winstone, and WinBench. We've distributed over 20 million copies of these tools, which are in use at every one of the 2001 Fortune 100 companies. Our Internet BenchMark service provides the definitive ratings for Internet Service Providers in the US, Canada, and the UK.

Under our former names of ZD Labs and eTesting Labs, and as part of VeriTest since July of 2002, we have delivered rigorous, objective, independent testing and analysis for over a decade. With the most knowledgeable staff in the business, testing facilities around the world, and almost 1,600 dedicated network PCs, VeriTest offers our clients the expertise and equipment necessary to meet all their testing needs.

**For more information** email us at [info@veritest.com](mailto:info@veritest.com) or call us at 919-380-2800.

#### **Disclaimer of Warranties; Limitation of Liability:**

VERITEST HAS MADE REASONABLE EFFORTS TO ENSURE THE ACCURACY AND VALIDITY OF ITS TESTING, HOWEVER, VERITEST SPECIFICALLY DISCLAIMS ANY WARRANTY, EXPRESSED OR IMPLIED, RELATING TO THE TEST RESULTS AND ANALYSIS, THEIR ACCURACY, COMPLETENESS OR QUALITY, INCLUDING ANY IMPLIED WARRANTY OF FITNESS FOR ANY PARTICULAR PURPOSE. ALL PERSONS OR ENTITIES RELYING ON THE RESULTS OF ANY TESTING DO SO AT THEIR OWN RISK, AND AGREE THAT VERITEST, ITS EMPLOYEES AND ITS SUBCONTRACTORS SHALL HAVE NO LIABILITY WHATSOEVER FROM ANY CLAIM OF LOSS OR DAMAGE ON ACCOUNT OF ANY ALLEGED ERROR OR DEFECT IN ANY TESTING PROCEDURE OR RESULT.

IN NO EVENT SHALL VERITEST BE LIABLE FOR INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH ITS TESTING, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN NO EVENT SHALL VERITEST'S LIABILITY, INCLUDING FOR DIRECT DAMAGES, EXCEED THE AMOUNTS PAID IN CONNECTION WITH VERITEST'S TESTING. CUSTOMER'S SOLE AND EXCLUSIVE REMEDIES ARE AS SET FORTH HEREIN.