



NETWORK FIREWALL
TEST METHODOLOGY 3.0

All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the authors. This report shall be treated at all times as a confidential and proprietary report for internal use only.

Please note that access to or use of this Methodology is conditioned on the following:

1. The information in this Methodology is subject to change by NSS Labs without notice.
2. The information in this Methodology is believed by NSS Labs to be accurate and reliable, but is not guaranteed. All use of and reliance on this Methodology are at your sole risk. NSS Labs is not liable or responsible for any damages, losses or expenses arising from any error or omission in this Report.
3. NO WARRANTIES, EXPRESS OR IMPLIED ARE GIVEN BY THE NSS LABS. ALL IMPLIED WARRANTIES, INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT ARE DISCLAIMED AND EXCLUDED BY NSS LABS. IN NO EVENT SHALL NSS LABS BE LIABLE FOR ANY CONSEQUENTIAL, INCIDENTAL OR INDIRECT DAMAGES, OR FOR ANY LOSS OF PROFIT, REVENUE, DATA, COMPUTER PROGRAMS, OR OTHER ASSETS, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.
4. This Methodology does not constitute an endorsement, recommendation or guarantee of any of the products (hardware or software) tested or the hardware and software used in testing the products. The testing does not guarantee that there are no errors or defects in the products, or that the products will meet your expectations, requirements, needs or specifications, or that they will operate without interruption.
5. This Methodology does not imply any endorsement, sponsorship, affiliation or verification by or with any companies mentioned in this report. For PCI-related reports, this does not constitute an endorsement by the PCI Security Standards Council.
6. All trademarks, service marks, and trade names used in this Report are the trademarks, service marks, and trade names of their respective owners, and no endorsement of, sponsorship of, affiliation with, or involvement in, any of the testing, this Report or NSS Labs is implied, nor should it be inferred.

CONTACT INFORMATION

NSS Labs

P.O. Box 130573
Carlsbad, CA 92013 USA
+1 (760) 412-4627
info@nsslabs.com
www.nsslabs.com

CONTENTS

1	<i>Introduction</i>	1
1.1	The Need for the Firewalls	1
1.2	What is a Firewall?.....	1
1.3	About This Test Methodology and Report	1
1.4	Inclusion Criteria	1
1.5	About NSS Labs.....	1
2	<i>Product Guidance</i>	2
2.1	Recommend	2
2.2	Neutral.....	2
2.3	Caution	2
3	<i>Security Effectiveness</i>	3
3.1	Firewall Policy Enforcement.....	3
4	<i>Firewall Performance</i>	6
4.1	Raw Packet Processing Performance (UDP Traffic).....	6
4.2	LATENCY	7
4.3	Maximum Capacity	7
4.4	HTTP Capacity With No Transaction Delays.....	9
4.5	“Real-World” Traffic	10
5	<i>Stability & Reliability</i>	12
6	<i>Management and Configuration Costs</i>	13
6.1	Ease-of-Use.....	13
6.2	Expected Costs.....	13
6.3	Total Cost of Ownership.....	13
	<i>Appendix A: Test Environment</i>	14
	<i>Appendix B: Special Thanks</i>	15

1 INTRODUCTION

1.1 THE NEED FOR THE FIREWALLS

Firewall technology has been around for at least 25 years, and undergone several stages of development; from early packet and circuit firewalls to application layer and dynamic packet firewalls. Across these stages, the goal has continued to be to provide a protective barrier between internal and external networks, while allowing for productive communications to pass from one side to the other.

1.2 WHAT IS A FIREWALL?

As Firewalls which will be deployed at critical choke-points in the network, the stability and reliability of a Firewall is imperative. Therefore prime directive of any Firewall is that it must be as stable, as reliable, as fast, and as flexible as the existing firewall that it is replacing.

In order to establish a secure perimeter, a Firewall must provide granular control based upon the source and destination IP Addresses and ports.

The following capabilities are considered essential as part of a Firewall:

- Basic packet filtering
- Stateful multi-layer inspection
- NAT
- Highly Stable
- Ability to operate at layer 3

1.3 ABOUT THIS TEST METHODOLOGY AND REPORT

NSS Labs' test reports are designed to address the challenges faced by IT professionals in selecting and managing security products. The scope of this particular report is focused on:

- Security effectiveness
- Performance
- Stability
- Total Cost of Ownership (TCO)

1.4 INCLUSION CRITERIA

In order to garner the greatest participation, and allay any potential concerns of bias, we invited all leading firewall vendors, and those claiming Firewall capabilities, to submit products at no cost. Thus vendors with major market share, as well as challengers with new technology and less of a track record were included.

1.5 ABOUT NSS LABS

NSS Labs performs expert, independent security product evaluations and certifications to assist IT teams in selecting and managing the right security products for their environment. We operate the largest security and performance lab in the world. Our test reports and analysis are highly regarded by information security professionals for their rigor, depth, and integrity, and are used to validate purchasing decisions in global organizations.

2 PRODUCT GUIDANCE

NSS Labs issues summary product guidance based on evaluation criteria that is important to information security professionals. The evaluation criteria are weighted as follows:

1. **Security effectiveness** - The primary reason for buying a Firewall is to separate internal trusted networks from external untrusted networks while allowing select controlled traffic to flow between trusted and untrusted.
2. **Resistance to Evasion**- Failure in any evasion class permits attackers to circumvent protection.
3. **Stability** - Long term stability is particularly important for an in-line device, where failure can produce network outages
4. **Performance** – Correctly sizing a firewall is essential
5. **Value** – Customers should seek low TCO and high effectiveness and performance rankings.

Products are listed in rank order according to their guidance rating.

2.1 RECOMMEND

A Recommend rating from NSS Labs indicates that a product has performed well and deserves strong consideration. Only the top technical products earn a Recommend rating from NSS Labs—regardless of market share, company size, or brand recognition.

2.2 NEUTRAL

A Neutral rating from NSS Labs indicates that a product has performed reasonably well and should continue to be used if it is the incumbent within an organization. Products that earn a Neutral rating from NSS Labs deserve consideration during the purchasing process.

2.3 CAUTION

A Caution rating from NSS Labs indicates that a product has performed poorly. Organizations using one of these products should review their security posture and other threat mitigation factors, including possible alternative configurations and replacement. Products that earn a Caution rating from NSS Labs should not be short-listed or renewed.

3 SECURITY EFFECTIVENESS

This section verifies that the DUT is capable of enforcing a specified security policy effectively. NSS Labs' Firewall (Firewall) analysis is conducted by incrementally building upon a baseline configuration (simple routing with no policy restrictions and no content inspection) to a complex real world multiple zone configuration supporting many addressing modes, policies, applications, and inspection engines.

At each level of complexity, test traffic is passed across the Firewall to ensure that only specified traffic is allowed and the rest is denied, and that appropriate log entries are recorded.

The Firewall must support stateful firewalling either by managing state tables to prevent "traffic leakage" or as a stateful proxy. The ability to manage firewall policy across multiple interfaces/zones is a required. At a minimum, the Firewall must provide a "trusted" internal interface, an "untrusted" external/Internet interface, and (optionally) one or more DMZ interfaces. In addition, a dedicated management interface (virtual or otherwise) is preferred.

3.1 FIREWALL POLICY ENFORCEMENT

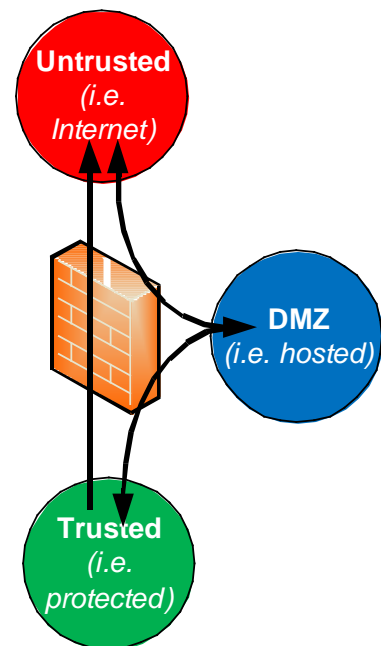
Policies are rules that are configured on a firewall to permit or deny access from one network resource to another based on identifying criteria such as: source, destination, and service. A term typically used to define the demarcation point of a network where policy is applied is a *demilitarized zone* (DMZ). Policies are typically written to permit or deny network traffic from one or more of the following zones:

- **Untrusted** – This is typically an external network and is considered to be an unknown and non-secure. An example of an untrusted network would be the Internet.
- **DMZ** – This is a network that is being *isolated* by the firewall restricting network traffic to and from hosts contained within the isolated network.
- **Trusted** – This is typically an internal network; a network that is considered secure and protected.

The NSS Labs Firewall certification tests performance and the ability to enforce policy between the following:

- Trusted to Untrusted
- Untrusted to DMZ
- Trusted to DMZ

Note: Firewalls must provide at a minimum one DMZ interface in order to provide a DMZ or "transition point" between untrusted and trusted networks



3.1.1 BASELINE POLICY

Routed configuration with an "allow all" policy

3.1.2 SIMPLE POLICIES

Simple outbound and inbound policies allowing basic browsing and e-mail access for internal clients and no external access

3.1.3 COMPLEX POLICIES

Complex outbound and inbound policies consisting of many rules, objects, and services.

3.1.4 STATIC NAT (NETWORK ADDRESS TRANSLATION)

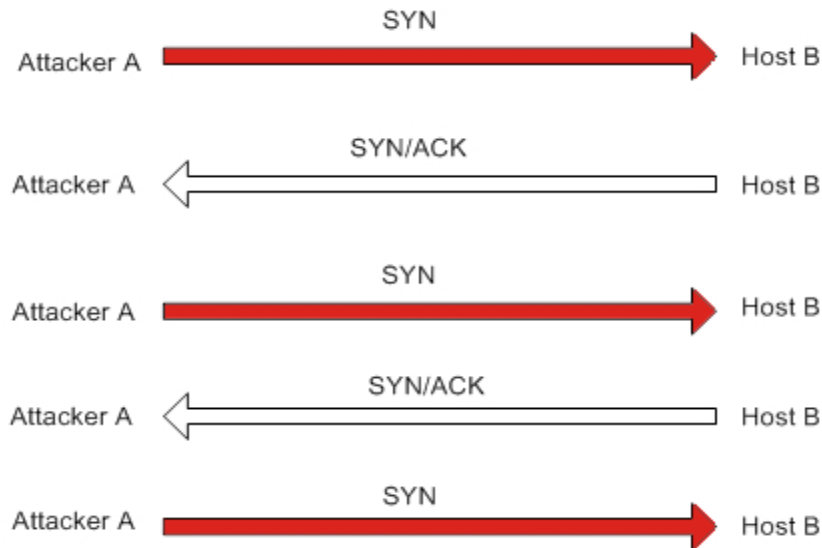
Inbound Network Address Translation (NAT) to DMZ using fixed IP address translation with one-to-one mapping.

3.1.5 DYNAMIC/HIDE NAT (NETWORK ADDRESS TRANSLATION)

Outbound Network Address Translation (NAT) (from Internal to External) where all outbound traffic “hides” behind the IP Address of the External Interface of the Firewall utilizing a pool of high ports to manage multiple connections.

3.1.6 SYN FLOOD PROTECTION

The basis of a SYN Flood attack is to not complete the 3-way handshake necessary to establish communication. Specifically the attacker (client machine A in fig. 6) refusing to send the ACK signal to the host server (B) after receiving the SYN/ACK from Host B. Such a connection is called a half open connection.



Instead of sending an ACK, attacker A sends another SYN signal to the victim server. The server again acknowledges it with a SYN/ACK and B again refuses to send the final ACK signal. By repeating this several times the attacker tries to overflow the data structure of the host server. The data structure is built in the memory of the host server with the purpose of keeping records of connections to be completed (or half open connections). Since the data structure is of a finite size, it is possible to overflow it by establishing a large number of open connections.

Once overflow occurs the host server will not be able to accept new connections thus resulting in a denial of service. There is however a time-out associated with each of the connections (approximately 3 minutes) after which the host server will automatically drop the half open connection and can start accepting new connections. If the attacker can request connections at a rate higher than the victim servers ability to expire the pending connections then it is possible to crash the server.

Thus the objective of SYN flooding is to disable one side of the TCP connection which will result in one or more of the following:

- The server is unable to accept new connections.
- The server crashes or becomes inoperative.
- Authorization between servers is impaired.

The Firewall is expected to protect against SYN Floods.

3.1.7 IP ADDRESS SPOOFING

This test attempts to confuse the Firewall into allowing traffic to pass from one network segment to another. Each IP packet header contains the source and destination address of the packet. The source address is normally the address that the packet was sent from. By forging the header so it contains a different source address, an attacker can make it appear that the packet was sent by a different (trusted) machine. The machine that receives spoofed packets will send response back to the forged source address.

The Firewall is expected to protect against IP Address spoofing.

3.1.8 TCP SPLIT HANDSHAKE SPOOF

This test attempts to confuse the Firewall into allowing traffic to pass from one network segment to another. The TCP Split handshake blends features of both the three way handshake and the simultaneous-open connection. The result is a TCP Spoof that allows an attacker to bypass the firewall by having the attacker instruct the target to "initiate" the session back with the attacker. Popular TCP/IP networking stacks respect this handshaking method, including Microsoft, Apple, and Linux stacks, with no modification.¹

The Firewall is expected to protect against TCP Split Handshake spoofing.

¹ The TCP Split Handshake: Practical Effects on Modern Network Equipment, Tod Alien Beardsley & Jin Qian, <http://www.macrothink.org/journal/index.php/npa/article/view/285>

4 FIREWALL PERFORMANCE

This section measures the performance of the Firewall using various traffic conditions that provide metrics for real world performance. Individual implementations will vary based on usage, however these quantitative metrics provide a gauge as to the performance boundary conditions and sweet spot of a particular hypervisor and can indicate whether or not it is appropriate for a given environment.

4.1 RAW PACKET PROCESSING PERFORMANCE (UDP TRAFFIC)

This test uses UDP packets of varying sizes generated by both BreakingPoint Systems and Spirent SmartBits traffic generation tools.

A constant stream of the appropriate packet size—with variable source IP addresses and ports transmitting to a single fixed IP address/port—is transmitted bi-directionally through each port pair of the IPS.

Each packet contains dummy data, and is targeted at a valid port on a valid IP address on the target subnet. The percentage load and packets per second (pps) figures across each in-line port pair are verified by the Adtech network monitoring tool before each test begins. Multiple tests are run and averages taken where necessary.

This traffic does not attempt to simulate any form of “real-world” network condition. No TCP sessions are created during this test, and there is very little for the detection engine to do in the way of protocol analysis (although each vendor will be required to write a signature to detect the test packets to ensure that they are being passed through the detection engine and not “fast-tracked” from the inbound to outbound port).

The aim of this test is purely to determine the raw packet processing capability of each in-line port pair of the IPS, and its effectiveness at passing “useless” packets quickly in order to pass potential attack packets to the detection engine.

4.1.1 128 BYTE PACKETS

Maximum 842,000 packets per second per Gigabit of traffic

4.1.2 256 BYTE PACKETS

Maximum 452,000 packets per second per Gigabit of traffic.

4.1.3 512 BYTE PACKETS

Maximum 235,000 packets per second per Gigabit of traffic. This test provides a reasonable indication of the ability of a device to process packets from the wire on an “average” network.

4.1.4 1024 BYTE PACKETS

Maximum 120,000 packets per second per Gigabit of traffic.

4.1.5 1514 BYTE PACKETS

Maximum 82,000 packets per second per Gigabit of traffic. This test has been included mainly to demonstrate how easy it is to achieve good results using large packets. Readers should use caution

when taking into consideration those test results that only quote performance figures using similar packet sizes.

4.2 LATENCY

The aim of the latency and user response time tests is to determine the effect the Firewall has on the traffic passing through it under various load conditions. Should a device impose a high degree of latency on the packets passing through it, a network or security administrator would need to think carefully about how many devices could be installed in a single data path before user response times became unacceptable or the combination of devices caused excessive timeouts.

This test uses UDP packets of varying sizes generated by BreakingPoint appliances to determine raw packet latency. The BreakingPoint software runs through several iterations of the test, varying the traffic load through multiple in-line port pairs bi-directionally from 25% to 100% of the maximum DUT throughput with zero packet loss.

This is repeated for a range of packet sizes (128, 256, 512, 1024 and 1514 bytes) of UDP traffic with variable IP addresses and ports. At each iteration of the test, the test equipment records the number of packets dropped, together with average and maximum latency, measured in microseconds.

This test - while not indicative of real-life network traffic - provides an indication of how much the DUT affects the traffic flow through it. This data is particularly useful for network administrators who need to gauge the effect of any form of in-line device which is likely to be placed at critical points within the corporate network.

4.2.1 128 BYTE PACKETS

Maximum 842,000 packets per second per Gigabit of traffic

4.2.2 256 BYTE PACKETS

Maximum 452,000 packets per second per Gigabit of traffic.

4.2.3 512 BYTE PACKETS

Maximum 235,000 packets per second per Gigabit of traffic. This test provides a reasonable indication of the ability of a device to process packets from the wire on an "average" network.

4.2.4 1024 BYTE PACKETS

Maximum 120,000 packets per second per Gigabit of traffic.

4.2.5 1514 BYTE PACKETS

Maximum 82,000 packets per second per Gigabit of traffic. This test has been included mainly to demonstrate how easy it is to achieve good results using large packets. Readers should use caution when taking into consideration those test results that only quote performance figures using similar packet sizes.

4.3 MAXIMUM CAPACITY

The use of BreakingPoint appliances allows us to create true “real world” traffic at multi-Gigabit speeds as a background load for our tests.

The aim of these tests is to stress the detection engine and determine how the hypervisor and its VMs cope with large numbers of TCP connections per second, application layer transactions per second, and concurrent open connections. All packets contain valid payload and address data, and these tests provide an excellent representation of a live network at various connection/transaction rates.

Note that in all tests, the following critical “breaking points” - where the final measurements are taken - are used:

- **Excessive concurrent TCP connections** - unacceptable increase in open connections on the server-side
- **Excessive response time for HTTP transactions** - excessive delays and increased response time to client
- **Unsuccessful HTTP transactions** - normally there should be zero unsuccessful transactions. Once these appear, it is an indication that excessive latency causing connections to time out

Note: All tests in this section are performed multiple times with 1, 4, and 8 VMs per hypervisor to determine the net productivity per VM as well as overall productivity per hypervisor. Each number is recorded, and the net impact on performance is calculated and reported.

4.3.1 THEORETICAL MAXIMUM CONCURRENT TCP CONNECTIONS

This test is designed to determine the maximum concurrent TCP connections of the VM with no data passing across the connections. This type of traffic would not typically be found on a normal network, but it provides the means to determine the maximum possible concurrent connections figure.

A maximum of 7.5 million Layer 4 TCP sessions are opened through the device. Each session is opened normally and then held open for the duration of the test as additional sessions are added up to the maximum possible. Load is increased until no more connections can be established, and this number is recorded.

4.3.2 THEORETICAL MAXIMUM CONCURRENT TCP CONNECTIONS WITH DATA

This test is identical to 4.1.1 except that once the maximum number of concurrent connections have been established, 1GB of data is transmitted 21KB segments. This ensures that the VMs are capable of passing data across the connections once they have been established.

4.3.3 MAXIMUM TCP CONNECTIONS PER SECOND

This test is designed to determine the maximum TCP connection rate of the VM with one byte of data passing across the connections. This type of traffic would not typically be found on a normal network, but it provides the means to determine the maximum possible TCP connection rate.

A maximum of 750,000 connections per second are generated, ramped slowly to determine the exact point of failure. Each session is opened normally, one byte of data passed to the host, and then the session is closed immediately. Load is increased until one or more of the breaking points defined earlier is reached.

4.3.4 MAXIMUM HTTP CONNECTIONS PER SECOND

This test is designed to determine the maximum TCP connection rate of the DUT with a 1 byte HTTP response size. The response size defines the number of bytes contained in the body, excluding any bytes associated with the HTTP header. A 1 byte response size is designed to provide a theoretical maximum HTTP connections per second rate.

Client and server are using HTTP 1.0 without keep alive, and the client will open a TCP connection, send one HTTP request, and close the connection. This ensures that all TCP connections are closed immediately the request is satisfied, thus any concurrent TCP connections will be caused purely as a result of latency of the host on which the HIPS is installed. Load is increased until one or more of the breaking points defined earlier is reached.

4.3.5 MAXIMUM HTTP TRANSACTIONS PER SECOND

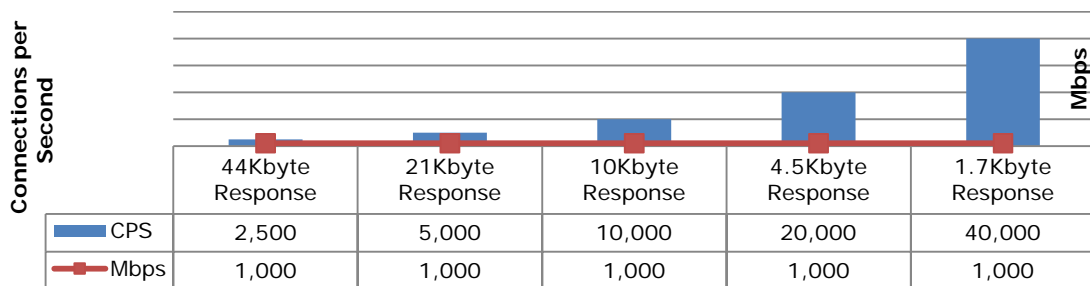
This test is designed to determine the maximum HTTP transaction rate of the DUT with a 1 byte HTTP response size. The object size defines the number of bytes contained in the body, excluding any bytes associated with the HTTP header. A 1 byte response size is designed to provide a theoretical maximum connections per second rate.

Client and server are using HTTP 1.1 with persistence, and the client will open a TCP connection, send ten HTTP requests, and close the connection. This ensures that TCP connections remain open until all ten HTTP transactions are complete, thus eliminating the maximum connection per second rate as a bottleneck (one TCP connection = 10 HTTP transactions). Load is increased until one or more of the breaking points defined earlier is reached.

4.4 HTTP CAPACITY WITH NO TRANSACTION DELAYS

The aim of these tests is to stress the HTTP detection engine and determine how the DUT copes with network loads of varying average packet size and varying connections per second. By creating genuine session-based traffic with varying session lengths, the VM is forced to track valid TCP sessions, thus ensuring a higher workload than for simple packet-based background traffic. This provides a test environment that is as close to “real world” as it is possible to achieve in a lab environment, while ensuring absolute accuracy and repeatability.

Each transaction consists of a single HTTP GET request and there are no transaction delays (i.e. the Web server responds immediately to all requests). All packets contain valid payload (a mix of binary and ASCII objects) and address data, and this test provides an excellent representation of a live network (albeit one biased towards HTTP traffic) at various network loads.



4.4.1 44KB HTTP RESPONSE SIZE – 2,500 CONNECTIONS PER SECOND

Max 2,500 new connections per second per Gigabit of traffic with a 44KB HTTP response size - average packet size 900 bytes - maximum 140,000 packets per second per Gigabit of traffic. With

relatively low connection rates and large packet sizes, all hosts should be capable of performing well throughout this test.

4.4.2 21KB HTTP RESPONSE SIZE – 5,000 CONNECTIONS PER SECOND

Max 5,000 new connections per second per Gigabit of traffic with a 21KB HTTP response size - average packet size 670 bytes - maximum 185,000 packets per second per Gigabit of traffic. With average connection rates and average packet sizes, this is a good approximation of a real-world production network, and all hosts should be capable of performing well throughout this test.

4.4.3 10KB HTTP RESPONSE SIZE – 10,000 CONNECTIONS PER SECOND

Max 10,000 new connections per second per Gigabit of traffic with a 10KB HTTP response size - average packet size 550 bytes - maximum 225,000 packets per second per Gigabit of traffic. With smaller packet sizes coupled with high connection rates this represents a very heavily used production network.

4.4.4 4.5KB HTTP RESPONSE SIZE – 20,000 CONNECTIONS PER SECOND

Max 20,000 new connections per second per Gigabit of traffic with a 4.5KB HTTP response size - average packet size 420 bytes - maximum 300,000 packets per second per Gigabit of traffic. With small packet sizes and extremely high connection rates this is an extreme test for any host.

4.4.5 1.7KB HTTP RESPONSE SIZE – 40,000 CONNECTIONS PER SECOND

Max 40,000 new connections per second per Gigabit of traffic with a 1.7KB HTTP response size - average packet size 270 bytes - maximum 445,000 packets per second per Gigabit of traffic. With small packet sizes and extremely high connection rates this is an extreme test for any host.

4.5 “REAL-WORLD” TRAFFIC

Where previous tests provide a pure HTTP environment with varying connection rates and average packet sizes, the aim of this test is to simulate a “real-world” environment by introducing additional protocols and real content while still maintaining a precisely repeatable and consistent background traffic load.

The result is a background traffic load that is closer to what may be found on a heavily-utilized “normal” production network.

4.5.1 “REAL-WORLD” PROTOCOL MIX (PERIMETER)

Traffic is generated across the firewall comprising the following protocol mix typically seen by a perimeter security device:

HTTP text	33%
HTTP Images (<50k)	14%
SMTP	18%
FTP	8%
DNS	6%
HTTP Video	4%
HTTP Audio	4%
HTTP Images (>300 kb)	4%
SSH	4%
AOL IM	3%
SIP/RTP	1%
BitTorrent	1%

For this test and the one described in Section 6.6.2, HTTP traffic comprises genuine transactions and web pages from real websites such as Google, Yahoo, MSN, and NSS Labs, including small (< 50KB) and large (>300 KB) JPEG images. Also included as part of the HTTP traffic is genuine QuickTime movie content and MP3 files, taking the total HTTP traffic of all types to approximately 65% of the overall load. SMTP traffic comprises real e-mail messages of varying lengths (with and without attachments) from the NSS Labs mail server. Maximum 6000 connections per second per Gigabit of traffic - 220,000 packets per second per Gigabit of traffic - average packet size of 550 bytes.

With lower connection rates, average packets sizes, and a common protocol mix comprising protocols which all require inspection by the firewall engine, this test is a good approximation of a heavily-used production network. All sensors should be capable of performing well throughout this test (and the one described below in Section 4.5.2).

4.5.2 “REAL-WORLD” PROTOCOL MIX (CORE)

Traffic is generated across the firewall comprising the following protocol mix typical of that seen by a network core security device:

HTTP text	24%
SMB File transfer	14%
HTTP Images (<50 kb)	12%
SMTP	12%
PostgreSQL	10%
DNS	6%
DCERPC	4%
FTP	3%
SMB NULL	3%
HTTP Video	2%
HTTP Audio	2%
HTTP Images (>300 kb)	2%
AIM	2%
SIP/RTP	1%
NFS	1%
SSH	1%
RTSP	1%

Maximum 5000 connections per second per Gigabit of traffic - 270,000 packets per second per Gigabit of traffic - average packet size of 440 bytes.

5 STABILITY & RELIABILITY

Long term stability is particularly important for an in-line device, where failure can produce network outages. These tests verify the stability of the DUT along with its ability to maintain security effectiveness while under normal load and while passing malicious traffic. Products that are not able to sustain legitimate traffic (or crash) while under hostile attack will not pass.

The DUT is required to remain operational and stable throughout these tests, and to block 100 per cent of previously blocked traffic, raising an alert for each. If any non-allowed traffic passes successfully - caused by either the volume of traffic or the DUT failing open for any reason - this will result in a FAIL.

5.1.1 BLOCKING UNDER EXTENDED ATTACK

The DUT is exposed to a constant stream of security policy violations over an extended period of time. The device is configured to block and alert, and thus this test provides an indication the effectiveness of both the blocking and alert handling mechanisms.

A continuous stream of security policy violations mixed with legitimate traffic is transmitted through the device at a maximum of 100Mbps (max 50,000 packets per second, average packet sizes in the range of 120-350 bytes) for 8 hours with no additional background traffic. This is not intended as a stress test in terms of traffic load (covered in the previous section) - merely a reliability test in terms of consistency of blocking performance.

The device is expected to remain operational and stable throughout this test, and to block 100 per cent of recognisable violations, raising an alert for each. If any recognisable policy violations are passed - caused by either the volume of traffic or the sensor failing open for any reason - this will result in a FAIL.

5.1.2 PASSING LEGITIMATE TRAFFIC UNDER EXTENDED ATTACK

This test is identical to 5.1.1, where the external interface of the device is exposed to a constant stream of exploits over an extended period of time.

The device is expected to remain operational and stable throughout this test, and to pass most/all of the legitimate traffic. If an excessive amount of legitimate traffic is blocked throughout this test - caused by either the volume of traffic or the DUT failing for any reason - this will result in a FAIL.

5.1.3 PROTOCOL FUZZING & MUTATION

This test stresses the protocol stacks of the DUT by exposing it to traffic from various protocol randomizer and mutation tools. Several of the tools in this category are based on the ISIC test suite and the BreakingPoint *Stack Scrambler* component.

Traffic load is a maximum of 350Mbps and 60,000 packets per second (average packet size is 690 bytes). Results are presented as a simple PASS/FAIL - the device is expected to remain operational and capable of detecting and blocking exploits throughout the test.

6 MANAGEMENT AND CONFIGURATION COSTS

Organizations should be concerned with the ongoing, amortized cost of operating security products. This section evaluates the costs associated with the purchase, installation, and ongoing management of the IPS.

6.1 EASE-OF-USE

6.1.1 INITIAL SETUP (HOURS)

The initial setup costs include those associated with time required to install and configure the IPS.

6.1.2 TIME REQUIRED FOR UPKEEP (HOURS PER YEAR)

The time required to keep the device in good health. This includes applying patches, OS updates, exporting and archiving logs, generating reports, and backing up the system.

6.2 EXPECTED COSTS

6.2.1 INITIAL PURCHASE

6.2.2 ONGOING MAINTENANCE & SUPPORT (ANNUAL)

6.2.3 INSTALLATION LABOR COST (@\$75/HR)

6.2.4 MANAGEMENT LABOR COST (PER YEAR @\$75/HR)

6.3 TOTAL COST OF OWNERSHIP

6.3.1 YEAR 1

6.3.2 YEAR 2

6.3.3 YEAR 3

6.3.4 THREE-YEAR TOTAL COST OF OWNERSHIP

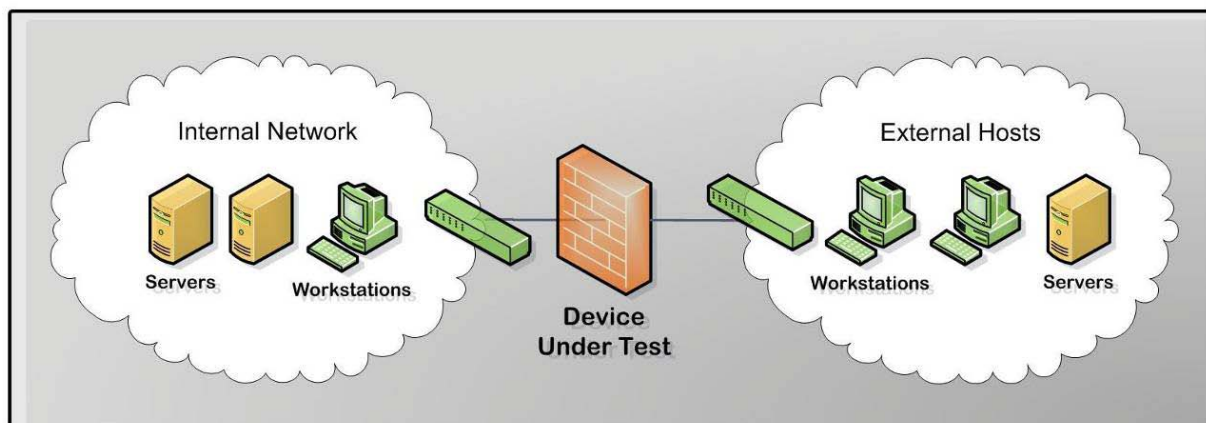
APPENDIX A: TEST ENVIRONMENT

The aim of this procedure is to provide a thorough test of all the main components of a routed firewall device in a controlled and repeatable manner and in the most “real-world” environment that can be simulated in a test lab.

The Test Environment

The NSS Labs test network is a multi-Gigabit infrastructure that can accommodate both Gigabit copper and 10 Gigabit fiber interfaces. The firewall is configured for the use-case according to the test methodology.

Traffic generation equipment—such as the hosts generating exploits, BreakingPoint transmit ports—is connected to the “external” network, while the “receiving” equipment—such as the vulnerable hosts for the exploits, BreakingPoint receive ports—is connected to the internal network. The firewall is connected between two “gateway” switches, one at the edge of the external network and one at the edge of the external network.



All “normal” network traffic, background load traffic, and exploit traffic is transmitted through the firewall, from external to internal (responses will flow in the opposite direction). The same traffic is mirrored to multiple SPAN ports of the external gateway switch, to which network monitoring devices are connected. The network monitoring devices ensure that the total amount of traffic per port pair reflects the amount being sent and received by the BreakingPoint.

The management interface is used to connect the appliance to the management console on a private subnet. This ensures that the firewall and console can communicate even when the target subnet is subjected to heavy loads, in addition to preventing attacks on the console itself.

APPENDIX B: SPECIAL THANKS

Special thanks go to our test infrastructure partners who provide much of the equipment, software, and support that make this testing possible:

