NEXT GENERATION INTRUSION PREVENTION SYSTEM (NGIPS) TEST REPORT

Check Point Software Technologies, Ltd. 13800 Next Generation Firewall Appliance vR77.20 –Recommended Policy

SEPTEMBER 22, 2016

Author – Tim Otto

Overview

NSS Labs performed an independent test of the Check Point Software Technologies, Ltd. 13800 Next Generation Firewall Appliance vR77.20. The product was subjected to thorough testing at the NSS facility in Austin, Texas, based on the Next Generation Intrusion Prevention System (NGIPS) methodology v2.0 available at www.nsslabs.com. This test was conducted free of charge and NSS did not receive any compensation in return for Check Point's participation.

Recommended Policy

NGIPS products are deployed at the perimeter and within corporate networks to protect employee desktops, laptops, and PCs. NSS research has determined that the majority of enterprises do not tune their NGIPS products, but rather rely on a vendor's default/recommended policies and settings. Since there are enterprises that do tune their devices, NSS has tested NGIPS products using both tuned and vendor-recommended settings.

NSS defines a Vendor-Recommended policy as an out-of-the-box, vendor-pre-defined policy that is available to all customers. NSS defines tuning as the act of changing the device setting from the default/recommended setting to a specific configuration based on the environment it is protecting. In both cases, the signatures/filters/rules that trigger false positives are turned off, since that is what would happen in an enterprise environment.

This Test Report provides results of the product being tested with the Vendor-Recommended policy.

Product	NSS Exploit Library Block Rate[1]	NSS-Tested Throughput	3-Year TCO (List Price)	3-Year TCO (Street Price)
Check Point 13800 Next Generation Firewall Appliance vR77.20	99.9%	12,345 Mbps	$173,725	$95,819
False Positives	Application Control		Evasions	Stability & Reliability
PASS	PASS		PASS	PASS

Figure 1 – Overall Test Results (Recommended Policy)

Using a recommended policy, the 13800 Next Generation Firewall Appliance blocked 100.0% of attacks against server applications, 99.9% of attacks against client applications, and 99.9% of attacks overall. The device proved effective against all evasion techniques tested. The device also passed all of the stability and reliability tests.

The Check Point 13800 Next Generation Firewall Appliance is rated by NSS at 12,345 Mbps, which is higher than the vendor-claimed performance (Check Point rates this device at 6,4 Gbps). NSS-tested throughput is calculated as an average of all the "real-world" protocol mixes and the 21 KB HTTP response-based capacity tests.

[1] Defined as the rate at which the device under test blocked exploits from the *NSS Exploit Library*. This value is a component of the overall block rate, which is reported in the NSS Labs Security Value Map™

Table of Contents

Table of Figures

Security Effectiveness

While the companion Comparative Reports on security, performance, and total cost of ownership (TCO) provide information about all tested products, this Test Report provides detailed information not available elsewhere.

This section verifies that the device under test (DUT) is capable of enforcing the security policy effectively.

Vendor Configuration

- Software Version: R77.20
- Protections Update Version: 634165175
- Profile: Recommended_Protection

False Positive Testing

The Check Point 13800 Next Generation Firewall Appliance correctly identified traffic and did not fire IPS alerts for non-malicious content.

NSS Exploit Library

In order to accurately represent the protection that may be achieved, NSS evaluates the DUT using a recommended policy.

NSS' security effectiveness testing leverages the deep expertise of our engineers who utilize multiple commercial, open-source, and proprietary tools as appropriate. With 1986 exploits, this is the industry's most comprehensive test to date. Most notably, all of the exploits and payloads in this test have been validated such that:

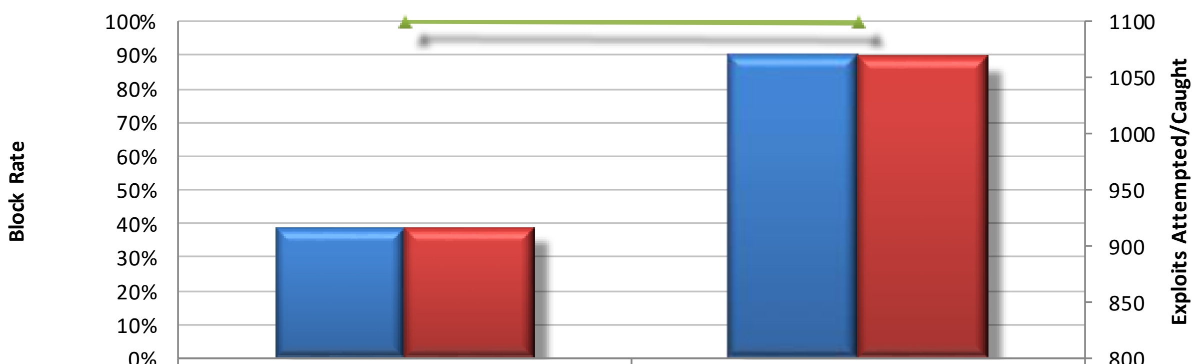
- A reverse shell is returned
- A bind shell is opened on the target, allowing the attacker to execute arbitrary commands
- Arbitrary code is executed
- A malicious payload is installed
- A system is rendered unresponsive
- Etc.

Product	Total Number of Exploits Run	Total Number Blocked	Block Percentage
Check Point 13800 Next Generation Firewall Appliance vR77.20	1986	1985	99.9%

Figure 2 – Number of Exploits Blocked in %

Coverage by Attack Vector

Because a failure to block attacks could result in significant compromise and impact to critical business systems, network intrusion prevention systems should be evaluated against a broad set of exploits. Exploits can be categorized into two groups: *attacker-initiated* and *target-initiated*. Attacker-initiated exploits are threats executed remotely against a vulnerable application and/or operating system by an individual while target-initiated exploits are initiated by the vulnerable target. With target-initated exploits, the most common type of attack experienced by the end user, the attacker has little or no control as to when the threat is executed.



	Attacker Initiated	Target Initiated
Attempted	916	1070
Caught	916	1069
Coverage	100.0%	99.9%

Figure 3 – Coverage by Attack Vector

Coverage by Impact Type

The most serious exploits are those that result in a remote system compromise, providing the attacker with the ability to execute arbitrary system-level commands. Most exploits in this class are "weaponized" and offer the attacker a fully interactive remote shell on the target client or server.

Slightly less serious are attacks that result in an individual service compromise, but not arbitrary system-level command execution.

Finally, there are attacks that result in a system- or service-level fault that crashes the targeted service or application and requires administrative action to restart the service or reboot the system. Clients can contact NSS for more information about these tests.

Coverage by Date

Figure 4 provides insight into whether or not a vendor is aging out protection signatures aggressively enough to preserve performance levels. It also reveals whether a product lags behind in protection for the most current vulnerabilities. NSS reports exploits by individual years for the past ten years. Exploits older than ten years are grouped together.

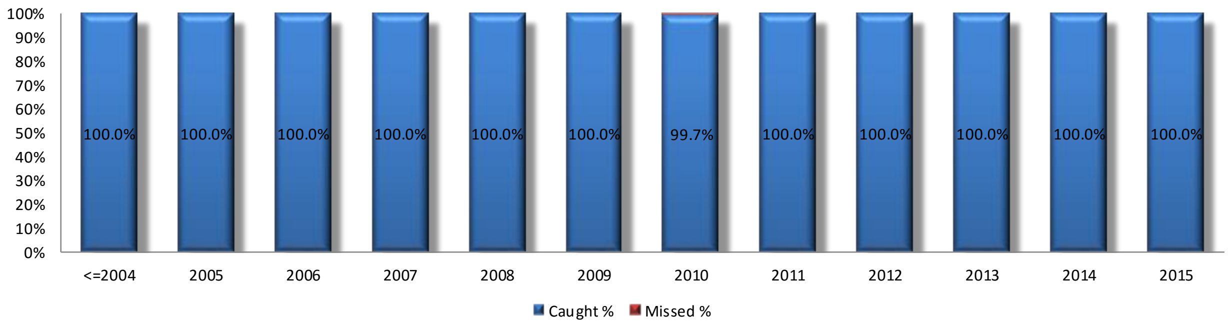


Figure 4 – Product Coverage by Date

Coverage by Target Vendor

Exploits within the *NSS Exploit Library* target a wide range of protocols and applications. Figure 5 highlights the coverage offered by the Check Point 13800 Next Generation Firewall Appliance for some of the top vendor targets (out of more than 70) represented for this round of testing.

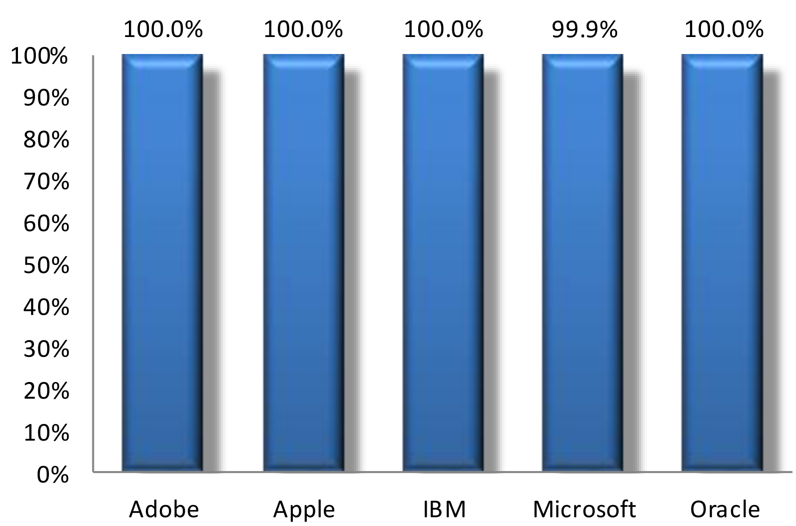


Figure 5 – Product Coverage by Target Vendor

Coverage by Result

These tests determine the protection provided against different types of exploits based on the intended action of those exploits; for example, arbitrary execution, buffer overflow, code injection, cross-site scripting, directory traversal, or privilege escalation. Further details are available to NSS clients via inquiry call.

Coverage by Target Type

These tests determine the protection provided against different types of exploits based on the target environment, for example, web server, web browser, database, ActiveX, Java, browser plugins, etc. Further details are available to NSS clients via inquiry call.

Application Control

An NGIPS should provide granular control based upon applications, not just ports. This capability is needed to re-establish a secure perimeter where unwanted applications are unable to tunnel over HTTP/S. As such, granular application control is a requirement of an NGIPS since it enables the administrator to define security policies based upon applications rather than ports alone. Figure 6 depicts whether the Check Point 13800 Next Generation Firewall Appliance passed or failed the application control test.

Test Procedure	Result
Block Unwanted Applications	PASS

Figure 6 – Application Control

Our testing found that the Check Point 13800 Next Generation Firewall Appliance correctly enforced complex outbound and inbound policies consisting of multiple rules, objects, and applications.

Live (Real-Time) Drive-by Exploits

Protection from web-based exploits targeting client applications, also known as "drive-by" downloads, can be effectively measured using NSS' Cyber Advanced Warning System (CAWS). CAWS provides continuous validation of a product's security effectiveness posture in real time.

For further detail, please see the companion Comparative Reports on Security and the NSS Security Value Map™.

Resistance to Evasion Techniques

Evasion techniques are a means of disguising and modifying attacks at the point of delivery to avoid detection and blocking by security products. Failure of a security device to correctly identify a specific type of evasion potentially allows an attacker to use an entire class of exploits for which the device is assumed to have protection. This renders the device virtually useless. Many of the techniques used in this test have been widely known for years and should be considered minimum requirements for the NGIPS product category.

Providing exploit protection results without fully factoring in evasion can be misleading. The more classes of evasion that are missed (such as IP packet fragmentation, stream segmentation, RPC fragmentation, URL obfuscation, payload encoding, and FTP evasion), the less effective the device. For example, it is better to miss all techniques in one evasion category, such as FTP evasion, than one technique in each category, which would result in a broader attack surface.

Furthermore, evasions operating at the lower layers of the network stack (IP packet fragmentation or stream segmentation) have a greater impact on security effectiveness than those operating at the upper layers (URL or

FTP obfuscation). Lower-level evasions will potentially impact a wider number of exploits; missing TCP segmentation, for example, is a much more serious issue than missing FTP obfuscation.

Figure 7 provides the results of the evasion tests for the Check Point 13800 Next Generation Firewall Appliance

Test Procedure	Result
IP Packet Fragmentation	PASS
Stream Segmentation	PASS
RPC Fragmentation	PASS
SMB & NetBIOS Evasions	PASS
URL Obfuscation	PASS
FTP Evasion	PASS
Payload Encoding	PASS
Layered Evasions	
IP Fragmentation + TCP Segmentation	PASS
IP Fragmentation + MSRPC Fragmentation	PASS
TCP Segmentation + SMB / NetBIOS Evasions	PASS
TCP Split Handshake	PASS

Figure 7 – Resistance to Evasion Results

Performance

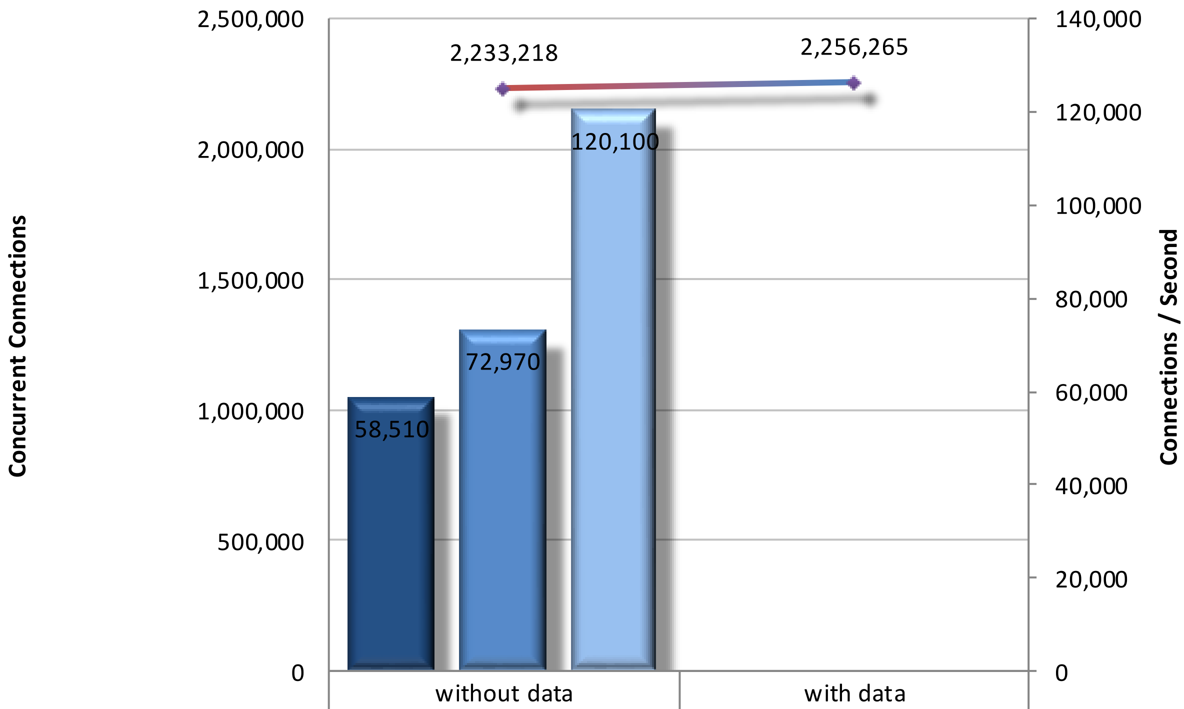
There is frequently a trade-off between security effectiveness and performance. Because of this trade-off, it is important to judge a product's security effectiveness within the context of its performance (and vice versa). This ensures that new security protections do not adversely impact performance and that security shortcuts are not taken to maintain or improve performance.

Maximum Capacity

The use of traffic generation appliances allows NSS engineers to create "real-world" traffic at multi-Gigabit speeds as a background load for the tests. The aim of these tests is to stress the inspection engine and determine how it copes with high volumes 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** – Latency within the NGIPS is causing an unacceptable increase in open connections.
- **Excessive concurrent HTTP connections** – Latency within the NGIPS is causing excessive delays and increased response time.
- **Unsuccessful HTTP transactions** – Normally, there should be zero unsuccessful transactions. Once these appear, it is an indication that excessive latency within the NGIPS is causing connections to time out.



	without data	with data
TCP Connections/Sec	58,510	
HTTP Connections/Sec	72,970	
HTTP Transactions/Sec	120,100	
Concurrent TCP Conns	2,233,218	2,256,265

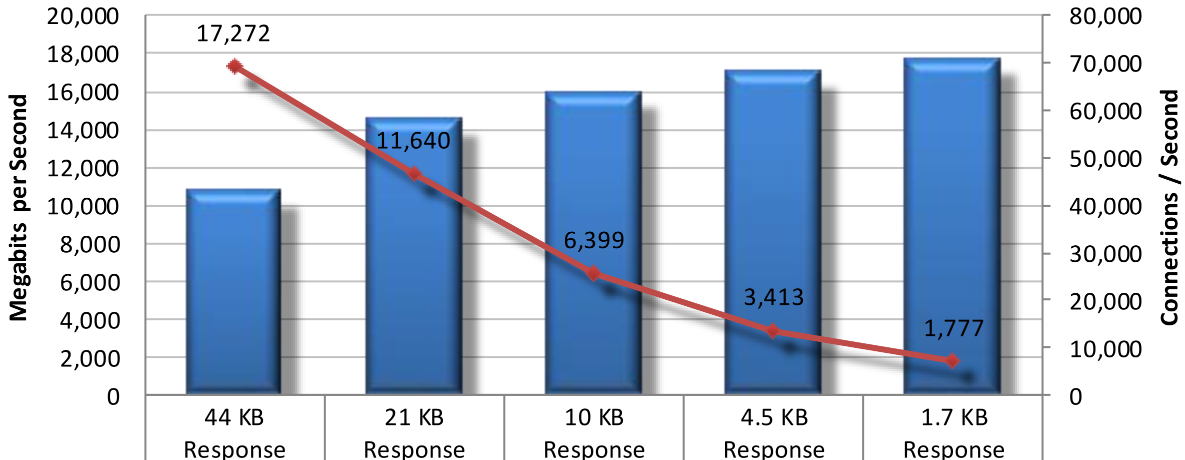
Figure 8 – Concurrency and Connection Rates

HTTP Connections per Second and Capacity

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 DUT 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 conditions as possible, while ensuring absolute accuracy and repeatability.

HTTP Capacity with No Transaction Delays

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. This test provides an excellent representation of a live network (albeit one biased toward HTTP traffic) at various network loads.

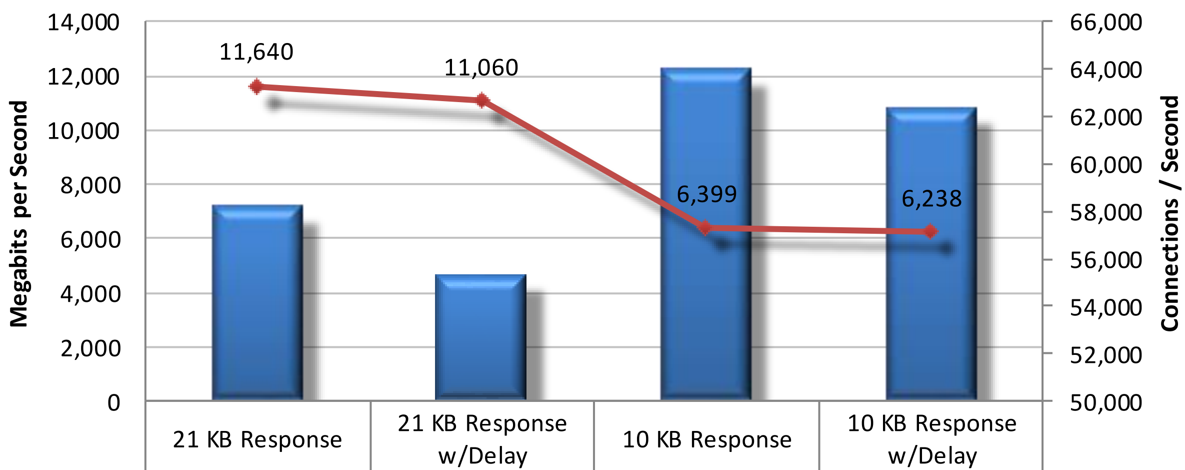


	44 KB Response	21 KB Response	10 KB Response	4.5 KB Response	1.7 KB Response
(bars)	43,180	58,200	63,990	68,260	71,080
Mbps	17,272	11,640	6,399	3,413	1,777

Figure 9 – HTTP Connections per Second and Capacity

HTTP Capacity with Transaction Delays

Typical user behavior introduces delays between requests and reponses (for example, “think time”) as users read web pages and decide which links to click next. This group of tests is identical to the previous group except that these include a five-second delay in the server response for each transaction. This has the effect of maintaining a high number of open connections throughout the test, thus forcing the sensor to utilize additional resources to track those connections.



	21 KB Response	21 KB Response w/Delay	10 KB Response	10 KB Response w/Delay
	58,200	55,300	63,990	62,380
Mbps	11,640	11,060	6,399	6,238

Figure 10 – HTTP Capacity with Transaction Delays

Application Average Response Time – HTTP

Application Average Response Time – HTTP (at 90% Maximum Load)	Milliseconds
2,500 Connections per Second – 44 KB Response	1.71
5,000 Connections per Second – 21 KB Response	1.21
10,000 Connections per Second – 10 KB Response	0.80
20,000 Connections per Second – 4.5 KB Response	0.44
40,000 Connections per Second – 1.7 KB Response	0.09

Figure 11 – Application Average Response Time (Milliseconds)

Real-World Traffic Mixes

This test measures the performance of the device under test in a “real-world” environment by introducing additional protocols and real content, while still maintaining a precisely repeatable and consistent background traffic load. Different protocol mixes are utilized based on the intended location of the device under test (network core or perimeter) to reflect real use cases. For details about “real-world” traffic protocol types and percentages, see the NSS Labs Next Generation Intrusion Prevention System Test Methodology, available at www.nsslabs.com.

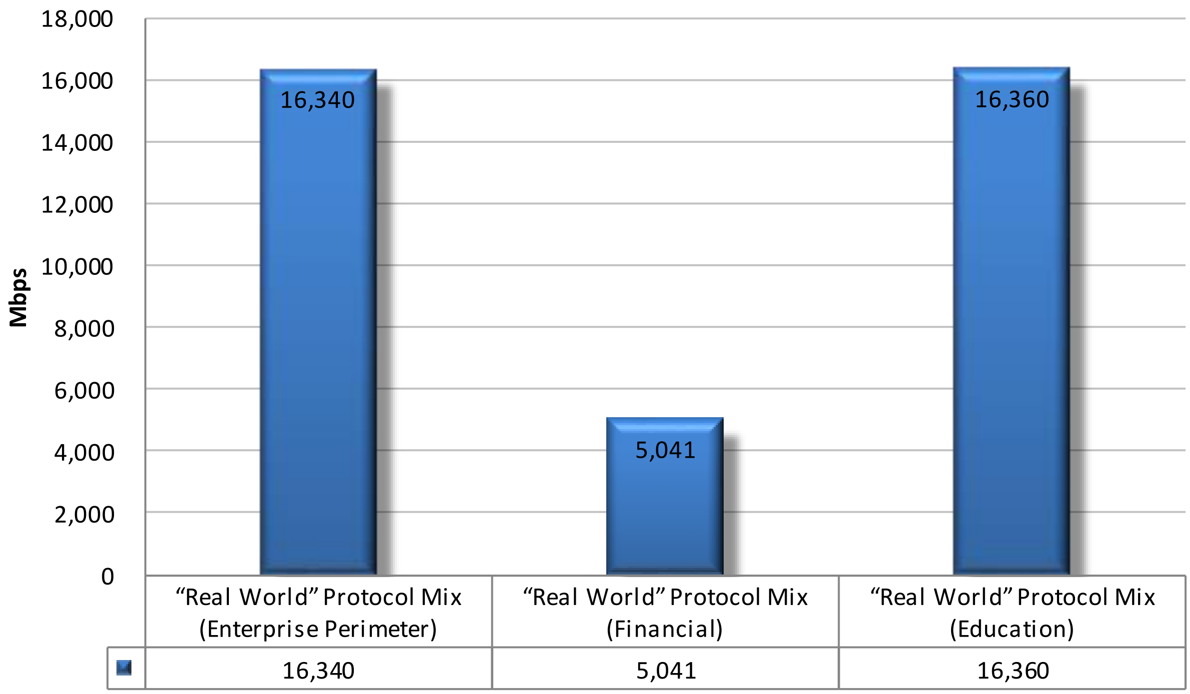


Figure 12 – Real-World Traffic Mixes

The 13800 Next Generation Firewall Appliance was tested by NSS and performed above the throughput claimed by the vendor for all real-world traffic mixes except for the financial traffic mix, where it performed below its vendor-claimed throughput.

Raw Packet Processing Performance (UDP Throughput)

This test uses UDP packets of specific sizes, generated by a traffic generation tool, to validate the packet performance of a DUT. A constant stream of packets of a specified size, with varying source and destination IP addresses, transmitting from varying source ports to a fixed destination port, is transmitted bi-directionally through each tested port pair of the DUT.

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 frames per second (fps) figures across each inline port pair are verified by network monitoring tools before each test begins. Vendors will be expected to create an ad hoc signature to detect random payload data prior to running the first test to demonstrate that deep packet inspection is occurring on this traffic. Multiple tests are run and averages taken where necessary.

This traffic does not attempt to simulate any form of real-world network use case. No TCP sessions are created during this test, and there is very little for the detection engine to do. The goal of this test is to determine the raw packet processing capability of each inline port pair of the DUT and its effectiveness at forwarding packets quickly, in order to provide the highest level of network performance and lowest latency.

Raw Packet Processing Performance (UDP Latency)

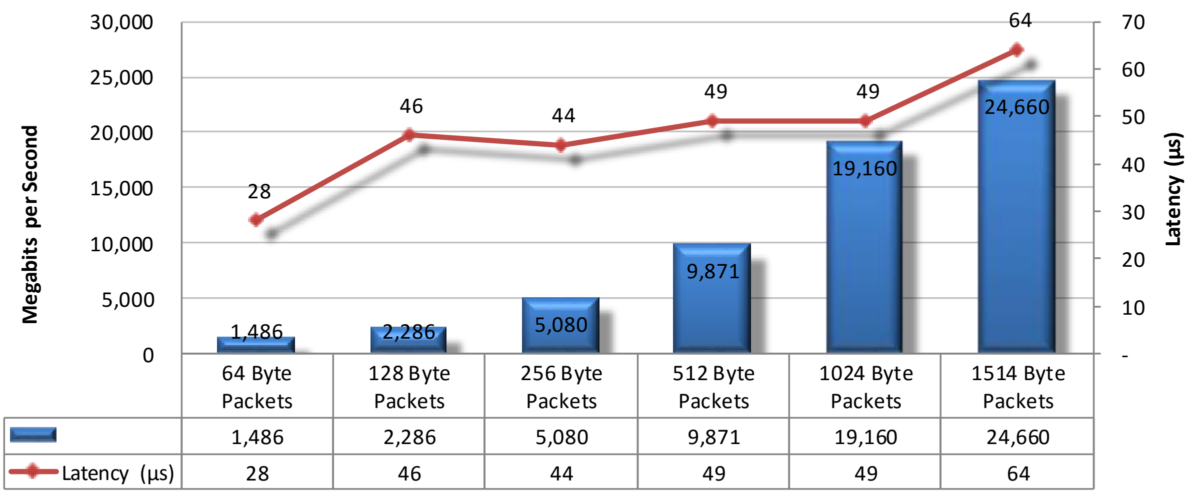


Figure 13 – Raw Packet Processing Performance (UDP Traffic)

NGIPS that introduce high levels of latency lead to unacceptable response times for users, especially where multiple security devices are placed in the data path. Figure 14 depicts UDP latency (in microseconds) as recorded during the UDP throughput tests at 90% of maximum load.

Latency – UDP	Microseconds
64-Byte Packets	28
128-Byte Packets	46
256-Byte Packets	44
512-Byte Packets	49
1024-Byte Packets	49
1514-Byte Packets	64

Figure 14 – UDP Latency in Microseconds

Stability and Reliability

Long-term stability is particularly important for an inline 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 cannot sustain legitimate traffic (or that crash) while under hostile attack will not pass.

The device is required to remain operational and stable throughout these tests, and to block 100% of previously blocked traffic, raising an alert for each. If any non-allowed traffic passes successfully, caused either by the volume of traffic or by the DUT failing open for any reason, the device will fail the test.

Stability and Reliability	Result
Blocking under Extended Attack	PASS
Passing Legitimate Traffic under Extended Attack	PASS
Behavior of the State Engine under Load	
• State Preservation – Normal Load	PASS
• State Preservation – Maximum Exceeded	PASS
Protocol Fuzzing and Mutation	PASS
Power Fail	PASS
Redundancy	YES
Persistence of Data	PASS

Figure 15 – Stability and Reliability Results

These tests also determine the behavior of the state engine under load. All products must choose whether to risk denying legitimate traffic or risk allowing malicious traffic once they run low on resources. A product will drop new connections when resources (such as state table memory) are low, or when traffic loads exceed its capacity. In theory, this means the NGIPS will block legitimate traffic but maintain state on existing connections (and prevent attack leakage).

Management and Configuration

Security devices are complicated to deploy; essential systems such as centralized management console options, log aggregation, and event correlation/management systems further complicate the purchasing decision.

Understanding key comparison points will allow customers to model the overall impact on network service level agreements (SLAs), to estimate operational resource requirements to maintain and manage the systems, and to better evaluate the required skills/competencies of staff.

Enterprises should include management and configuration during their evaluations, focusing on the following at a minimum:

- **General Management and Configuration** – How easy is it to install and configure devices, and how easy is it to deploy multiple devices throughout a large enterprise network?
- **Policy Handling** – How easy is it to create, edit, and deploy complicated security policies across an enterprise?
- **Alert Handling** – How accurate and timely is the alerting, and how easy is it to drill down to locate critical information needed to remediate a security problem?
- **Reporting** – How effective is the reporting capability, and how readily can it be customized?

Total Cost of Ownership (TCO)

Implementation of security solutions can be complex, with several factors affecting the overall cost of deployment, maintenance, and upkeep. All of the following should be considered over the course of the useful life of the solution:

- **Product Purchase** – The cost of acquisition
- **Product Maintenance** – The fees paid to the vendor, including software and hardware support, maintenance, and other updates
- **Installation** – The time required to take the device out of the box, configure it, put it into the network, apply updates and patches, and set up desired logging and reporting
- **Upkeep** – The time required to apply periodic updates and patches from vendors, including hardware, software, and other updates
- **Management** – Day-to-day management tasks, including device configuration, policy updates, policy deployment, alert handling, and so on

For the purposes of this report, capital expenditure (capex) items are included for a single device only (the cost of acquisition and installation).

Installation Hours

This table depicts the number of hours of labor required to install each device using only local device management options. The table accurately reflects the amount of time that NSS engineers, with the help of vendor engineers, needed to install and configure the DUT to the point where it operated successfully in the test harness, passed legitimate traffic, and blocked and detected prohibited or malicious traffic. This closely mimics a typical enterprise deployment scenario for a single device.

The installation cost is based on the time that an experienced security engineer would require to perform the installation tasks described above. This approach allows NSS to hold constant the talent cost and measure only the difference in time required for installation. Readers should substitute their own costs to obtain accurate TCO figures.

Product	Installation (Hours)
Check Point 13800 Next Generation Firewall Appliance vR77.20	8

Figure 16 – Sensor Installation Time (Hours)

List Price and Total Cost of Ownership

Calculations are based on vendor-provided pricing information. Where possible, the 24/7 maintenance and support option with 24-hour replacement is utilized, since this is the option typically selected by enterprise customers. Prices are for single device management and maintenance only; costs for central management solutions (CMS) may be extra.

Product	Purchase	Maintenance /Year	Year 1 Cost	Year 2 Cost	Year 3 Cost	3-Year TCO
Check Point 13800 Next Generation Firewall Appliance vR77.20	$107,500	$23,625	$126,375	$23,675	$23,675	$173,725

Figure 17 – List Price 3-Year TCO (US$)

- **Year 1 Cost** is calculated by adding installation costs (US$75 per hour fully loaded labor x installation time) + purchase price + first-year maintenance/support fees.

For the Check Point 13800 Next Generation Firewall Appliance, updates for the first year are included in the initial purchase price and are not counted again in Year 1 Cost.

- **Year 2 Cost** consists only of maintenance/support fees.
- **Year 3 Cost** consists only of maintenance/support fees.

Street Price and Total Cost of Ownership

Calculations are based on vendor-provided pricing information. Where possible, the 24/7 maintenance and support option with 24-hour replacement is utilized, since this is the option typically selected by enterprise customers. Prices are for single device management and maintenance only; costs for CMS may be extra.

Product	Purchase	Maintenance /Year	Year 1 Cost	Year 2 Cost	Year 3 Cost	3-Year TCO
Check Point 13800 Next Generation Firewall Appliance vR77.20	$59,125	$13,021	$69,776	$13,021	$13,021	$95,819

Figure 18 – Street Price 3-Year TCO

- **Year 1 Cost** is calculated by adding installation costs (US$75 per hour fully loaded labor x installation time) + purchase price + first-year maintenance/support fees.

For the Check Point 13800 Next Generation Firewall Appliance, updates for the first year are included in the initial purchase price and are not counted again in Year 1 Cost.

- **Year 2 Cost** consists only of maintenance/support fees.
- **Year 3 Cost** consists only of maintenance/support fees.

For additional TCO analysis, including for the CMS, refer to the TCO Comparative Report.

Detailed Product Scorecard

The following chart depicts the status of each test with quantitative results where applicable.

Description	Result
Security Effectiveness	
Exploit Library	
NSS Exploit Library Block Rate	99.9%
False Positive Testing	PASS
Coverage by Impact Type	
Attacker Initiated	100%
Target Initiated	99.9%
System Exposure	Contact NSS
Service Exposure	Contact NSS
System or Service Fault	Contact NSS
Coverage by Date	Contact NSS
Coverage by Target Vendor	Contact NSS
Coverage by Result	Contact NSS
Coverage by Target Type	Contact NSS
Evasions and Attack Leakage	
Resistance to Evasion	PASS
IP Packet Fragmentation	100%
Ordered 8 byte fragments	100%
Ordered 16 byte fragments	100%
Ordered 24 byte fragments	100%
Ordered 32 byte fragments	100%
Out of order 8 byte fragments	100%
Ordered 8 byte fragments, duplicate last packet	100%
Out of order 8 byte fragments, duplicate last packet	100%
Ordered 8 byte fragments, reorder fragments in reverse	100%
Ordered 16 byte fragments, fragment overlap (favor new)	100%
Ordered 16 byte fragments, fragment overlap (favor old)	100%
Out of order 8 byte fragments, interleaved duplicate packets scheduled for later delivery	100%
Ordered 8 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload.	100%
Ordered 16 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload.	100%
Ordered 24 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload.	100%
Ordered 32 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload.	100%
TCP Stream Segmentation	100%
Ordered 1 byte segments, interleaved duplicate segments with invalid TCP checksums	100%
Ordered 1 byte segments, interleaved duplicate segments with null TCP control flags	100%
Ordered 1 byte segments, interleaved duplicate segments with requests to resync sequence numbers mid-stream	100%
Ordered 1 byte segments, duplicate last packet	100%
Ordered 2 byte segments, segment overlap (favor new)	100%
Ordered 1 byte segments, interleaved duplicate segments with out-of-window sequence numbers	100%
Out of order 1 byte segments	100%
Out of order 1 byte segments, interleaved duplicate segments with faked retransmits	100%
Ordered 1 byte segments, segment overlap (favor new)	100%
Out of order 1 byte segments, PAWS elimination (interleaved duplicate segments with older TCP timestamp options)	100%
Ordered 16 byte segments, segment overlap (favor new (Unix))	100%

Ordered 32 byte segments	100%
Ordered 64 byte segments	100%
Ordered 128 byte segments	100%
Ordered 256 byte segments	100%
Ordered 512 byte segments	100%
Ordered 1024 byte segments	100%
Ordered 2048 byte segments (sending MSRPC request with exploit)	100%
Reverse Ordered 256 byte segments, segment overlap (favor new) with random data	100%
Reverse Ordered 512 byte segments, segment overlap (favor new) with random data	100%
Reverse Ordered 1024 byte segments, segment overlap (favor new) with random data	100%
Reverse Ordered 2048 byte segments, segment overlap (favor new) with random data	100%
Out of order 1024 byte segments, segment overlap (favor new) with random data, Initial TCP sequence number is set to 0xffffffff - 4294967295	100%
Out of order 2048 byte segments, segment overlap (favor new) with random data, Initial TCP sequence number is set to 0xffffffff - 4294967295	100%
RPC Fragmentation	100%
One-byte fragmentation (ONC)	100%
Two-byte fragmentation (ONC)	100%
All fragments, including Last Fragment (LF) will be sent in one TCP segment (ONC)	100%
All frags except Last Fragment (LF) will be sent in one TCP segment. LF will be sent in separate TCP seg (ONC)	100%
One RPC fragment will be sent per TCP segment (ONC)	100%
One LF split over more than one TCP segment. In this case no RPC fragmentation is performed (ONC)	100%
Canvas Reference Implementation Level 1 (MS)	100%
Canvas Reference Implementation Level 2 (MS)	100%
Canvas Reference Implementation Level 3 (MS)	100%
Canvas Reference Implementation Level 4 (MS)	100%
Canvas Reference Implementation Level 5 (MS)	100%
Canvas Reference Implementation Level 6 (MS)	100%
Canvas Reference Implementation Level 7 (MS)	100%
Canvas Reference Implementation Level 8 (MS)	100%
Canvas Reference Implementation Level 9 (MS)	100%
Canvas Reference Implementation Level 10 (MS)	100%
MSRPC messages are sent in the big endian byte order, 16 MSRPC fragments are sent in the same lower layer message, MSRPC requests are fragmented to contain at most 2048 bytes of payload	100%
MSRPC messages are sent in the big endian byte order, 32 MSRPC fragments are sent in the same lower layer message, MSRPC requests are fragmented to contain at most 2048 bytes of payload	100%
MSRPC messages are sent in the big endian byte order, 64 MSRPC fragments are sent in the same lower layer message, MSRPC requests are fragmented to contain at most 2048 bytes of payload	100%
MSRPC messages are sent in the big endian byte order, 128 MSRPC fragments are sent in the same lower layer message, MSRPC requests are fragmented to contain at most 2048 bytes of payload	100%
MSRPC messages are sent in the big endian byte order, 256 MSRPC fragments are sent in the same lower layer message, MSRPC requests are fragmented to contain at most 2048 bytes of payload	100%
MSRPC messages are sent in the big endian byte order, 512 MSRPC fragments are sent in the same lower layer message, MSRPC requests are fragmented to contain at most 2048 bytes of payload	100%
MSRPC messages are sent in the big endian byte order, 1024 MSRPC fragments are sent in the same lower layer message, MSRPC requests are fragmented to contain at most 2048 bytes of payload	100%
SMB & NetBIOS Evasions	100%
A chaffed NetBIOS message is sent before the first actual NetBIOS message. The chaff message is an unspecified NetBIOS message with HTTP GET request like payload	100%
A chaffed NetBIOS message is sent before the first actual NetBIOS message. The chaff message is an unspecified NetBIOS message with HTTP POST request like payload	100%
A chaffed NetBIOS message is sent before the first actual NetBIOS message. The chaff message is an unspecified NetBIOS message with MSRPC request like payload	100%
URL Obfuscation	100%
URL encoding – Level 1 (minimal)	100%

URL encoding – Level 2	100%
URL encoding – Level 3	100%
URL encoding – Level 4	100%
URL encoding – Level 5	100%
URL encoding – Level 6	100%
URL encoding – Level 7	100%
URL encoding – Level 8 (extreme)	100%
Directory Insertion	100%
Premature URL ending	100%
Long URL	100%
Fake parameter	100%
TAB separation	100%
Case sensitivity	100%
Windows \ delimiter	100%
Session splicing	100%
FTP Evasion	100%
Inserting spaces in FTP command lines	100%
Inserting non-text Telnet opcodes – Level 1 (minimal)	100%
Inserting non-text Telnet opcodes – Level 2	100%
Inserting non-text Telnet opcodes – Level 3	100%
Inserting non-text Telnet opcodes – Level 4	100%
Inserting non-text Telnet opcodes – Level 5	100%
Inserting non-text Telnet opcodes – Level 6	100%
Inserting non-text Telnet opcodes – Level 7	100%
Inserting non-text Telnet opcodes – Level 8 (extreme)	100%
Payload Encoding	100%
x86/call4_dword_xor	100%
x86/fnstenv_mov	100%
x86/jmp_call_additive	100%
x86/shikata_ga_nai	100%
Layered Evasions	100%
IP Fragmentation + TCP Segmentation	100%
Ordered 8 byte fragments + Ordered TCP segments except that the last segment comes first	100%
Ordered 24 byte fragments + Ordered TCP segments except that the last segment comes first	100%
Ordered 32 byte fragments + Ordered TCP segments except that the last segment comes first	100%
Ordered 8 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload + Reverse order TCP segments, segment overlap (favor new), Overlapping data is set to zero bytes	100%
Ordered 16 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload + Out of order TCP segments, segment overlap (favor new), Overlapping data is set to zero bytes	100%
Ordered 24 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload + Out of order TCP segments, segment overlap (favor new), Overlapping data is set to zero bytes	100%
Ordered 32 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload + Out of order TCP segments, segment overlap (favor new), Overlapping data is set to zero bytes	100%
Ordered 8 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload + Out of order TCP segments, segment overlap (favor new), Overlapping data is set to random alphanumeric	100%
Ordered 16 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload + Out of order TCP segments, segment overlap (favor new), Overlapping data is set to random alphanumeric	100%
Ordered 32 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload + Out of order TCP segments, segment overlap (favor new), Overlapping data is set to random alphanumeric	100%

Ordered 8 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload + Out of order TCP segments, segment overlap (favor new), Overlapping data is set to random bytes	100%
Ordered 16 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload + Out of order TCP segments, segment overlap (favor new), Overlapping data is set to random bytes	100%
Ordered 24 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload + Out of order TCP segments, segment overlap (favor new), Overlapping data is set to random bytes	100%
Ordered 32 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has random payload + Out of order TCP segments, segment overlap (favor new), Overlapping data is set to random bytes	100%
IP Fragmentation + MSRPC Fragmentation	100%
Ordered 8 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has a shuffled payload + MSRPC messages are sent in the big endian byte order with 8 MSRPC fragments sent in the same lower layer message. MSRPC requests are fragmented to contain at most 2048 bytes of payload.	100%
Ordered 16 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has a shuffled payload + MSRPC messages are sent in the big endian byte order with 16 MSRPC fragments sent in the same lower layer message. MSRPC requests are fragmented to contain at most 2048 bytes of payload.	100%
Ordered 32 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has a shuffled payload + MSRPC messages are sent in the big endian byte order with 32 MSRPC fragments sent in the same lower layer message. MSRPC requests are fragmented to contain at most 64 bytes of payload.	100%
Ordered 64 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has a shuffled payload + MSRPC messages are sent in the big endian byte order with 64 MSRPC fragments sent in the same lower layer message. MSRPC requests are fragmented to contain at most 64 bytes of payload.	100%
Ordered 128 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has a random payload + MSRPC messages are sent in the big endian byte order with 1024 MSRPC fragments sent in the same lower layer message. MSRPC requests are fragmented to contain at most 128 bytes of payload.	100%
Ordered 256 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has a random payload + MSRPC messages are sent in the big endian byte order with 1024 MSRPC fragments sent in the same lower layer message. MSRPC requests are fragmented to contain at most 256 bytes of payload.	100%
Ordered 512 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has a random payload + MSRPC messages are sent in the big endian byte order with 1024 MSRPC fragments sent in the same lower layer message. MSRPC requests are fragmented to contain at most 512 bytes of payload.	100%
Ordered 1024 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has a random payload + MSRPC messages are sent in the big endian byte order with 1024 MSRPC fragments sent in the same lower layer message. MSRPC requests are fragmented to contain at most 1024 bytes of payload.	100%
IP Fragmentation + SMB Evasions	100%
Ordered 1024 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has a random payload + SMB chaff message before real messages. The chaff is a WriteAndX message with a broken write mode flag, and has random MSRPC request-like payload	100%
Ordered 8 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has a random payload + A chaffed NetBIOS message is sent before the first actual NetBIOS message. The chaff message is an unspecified NetBIOS message with MSRPC request like payload	100%
Ordered 8 byte fragments, duplicate packet with an incrementing DWORD in the options field. The duplicate packet has a random payload + A chaffed NetBIOS message is sent before the first actual	100%

NetBIOS message. The chaff message is an unspecified NetBIOS message with HTTP GET request like payload	
TCP Segmentation + SMB / NetBIOS Evasions	100%
Reverse Ordered 2048 byte TCP segments, segment overlap (favor new) with random data + A chaffed NetBIOS message is sent before the first actual NetBIOS message. The chaff message is an unspecified NetBIOS message with MSRPC request like payload	100%
TCP Split Handshake	100%
Performance	
Vendor-Claimed Performance	
Raw Packet Processing Performance (UDP Traffic)	Mbps
64 Byte Packets	92
128 Byte Packets	275
256 Byte Packets	321
512 Byte Packets	871
1024 Byte Packets	1,616
1514 Byte Packets	2,511
Latency – UDP	Microseconds
64 Byte Packets	28
128 Byte Packets	46
256 Byte Packets	44
512 Byte Packets	49
1024 Byte Packets	49
1514 Byte Packets	64
Maximum Capacity	
Theoretical Max. Concurrent TCP Connections	2,233,218
Theoretical Max. Concurrent TCP Connections w/Data	2,256,265
Theoretical Max. Throughput – Single Connection	788
Maximum TCP Connections Per Second	58,510
Maximum HTTP Connections Per Second	72,970
Maximum HTTP Transactions Per Second	120,100
HTTP Capacity With No Transaction Delays	
25,000 Connections Per Second – 44 KB Response	43,180
50,000 Connections Per Second – 21 KB Response	58,200
100,000 Connections Per Second – 10 KB Response	63,990
200,000 Connections Per Second – 4.5 KB Response	68,260
400,000 Connections Per Second – 1.7 KB Response	71,080
Application Average Response Time - HTTP (at 90% Max Load)	Milliseconds
25,000 Connections Per Second – 44 KB Response	1.71
50,000 Connections Per Second – 21 KB Response	1.21
100,000 Connections Per Second – 10 KB Response	0.80
200,000 Connections Per Second – 4.5 KB Response	0.44
400,000 Connections Per Second – 1.7 KB Response	0.09
HTTP CPS & Capacity With Transaction Delays	
21 KB Response With Delay	55,300
10 KB Response With Delay	62,380
Real-World Traffic	Mbps
Real World Protocol Mix (Enterprise Perimeter)	16,340
Real World Protocol Mix (Financial)	5,041
Real World Protocol Mix (Education)	16,360
Stability & Reliability	
Blocking Under Extended Attack	PASS
Passing Legitimate Traffic Under Extended Attack	PASS
Behavior of The State Engine under Load	

State Preservation - Normal Load	PASS
State Preservation - Maximum Exceeded	PASS
Protocol Fuzzing & Mutation	PASS
Power Fail	PASS
Redundancy	YES
Persistence of Data	PASS
Total Cost of Ownership (List Price)	
Ease of Use	
Initial Setup (Hours)	8
Time Required for Upkeep (Hours per Year)	Contact NSS Labs
Time Required to Tune (Hours per Year)	Contact NSS Labs
Expected Costs	
Initial Purchase (hardware as tested)	$107,500
Installation Labor Cost (@$75/hr)	$600
Annual Cost of Maintenance & Support (hardware/software)	$18,275
Annual Cost of Updates (IPS/AV/etc.)	$5,400
Initial Purchase (enterprise management system)	See Comparative
Annual Cost of Maintenance & Support (enterprise management system)	See Comparative
Total Cost of Ownership	
Year 1	$126,375
Year 2	$23,675
Year 3	$23,675
3-Year Total Cost of Ownership	$173,725
Total Cost of Ownership (Street Price)	
Ease of Use	
Initial Setup (Hours)	8
Time Required for Upkeep (Hours per Year)	Contact NSS Labs
Time Required to Tune (Hours per Year)	Contact NSS Labs
Expected Costs	
Initial Purchase (hardware as tested)	$59,125
Installation Labor Cost (@$75/hr)	$600
Annual Cost of Maintenance & Support (hardware/software)	$10,051
Annual Cost of Updates (IPS/AV/etc.)	$2,970
Initial Purchase (enterprise management system)	See Comparative
Annual Cost of Maintenance & Support (enterprise management system)	See Comparative
Total Cost of Ownership	
Year 1	$69,776
Year 2	$13,021
Year 3	$13,021
3-Year Total Cost of Ownership	$95,819

Figure 19 – Detailed Scorecard

Test Methodology

Next Generation Intrusion Prevention System Test Methodology v.2.0

A copy of the test methodology is available at www.nsslabs.com

Contact Information

NSS Labs, Inc.
206 Wild Basin Road
Building A, Suite 200
Austin, TX 78746
info@nsslabs.com
www.nsslabs.com

This and other related documents are available at **www.nsslabs.com**. To receive a licensed copy or report misuse, please contact NSS Labs.