TEST METHODOLOGY

Breach Prevention Systems (BPS)

MARCH 5, 2018

V2.0

Table of Contents

1 Introduction

1.1 The Need for Breach Prevention

Threat actors are demonstrating the capability to bypass protection offered by conventional endpoint and perimeter security solutions. Enterprises must in turn evolve their defenses to incorporate a different kind of protection, one that NSS Labs defines as a breach prevention system (BPS).

Through constant analysis of suspicious code and identification of communications with malicious hosts, breach prevention solutions are capable of providing enhanced detection of threats ranging from commodity malware to targeted attacks from state-sponsored threat actors that could bypass defenses such as next generation firewalls (NGFWs), intrusion preventions systems (IPS), intrusion detection systems (IDS), antivirus/endpoint protection (including host IPS), and secure web gateways (SWGs).

1.2 About This Test Methodology

NSS test reports are designed to address the challenges faced by IT professionals in selecting and managing security products. The scope of this particular methodology includes:

- Security effectiveness
- Performance
- Stability and reliability
- Total cost of ownership (TCO)

Based on the needs identified in NSS' research, the following capabilities are essential in any breach prevention system:

- Centralized management of multiple devices
- Breach prevention utilizing one or more of the following methods:
  - Malware identification (signatures, heuristics, or both)
  - Network traffic analysis (flow monitoring, content analysis, or both)
  - Sandboxing that allows for modeling of internal systems
  - Emulation
- Response mechanism (alerting, session termination, etc.)
- Robust logging of conviction events
- Reporting

Security Effectiveness: At the heart of the BPS test harness is a patented technology called **BaitNET™**. This cloud-based, fully instrumented, targeted execution environment, is a unique live test harness that is used by NSS for security effectiveness testing on all leading endpoint and network security products.

Using this test harness, NSS is able to determine the protection offered by the BPS as a whole, as well as many of its subcomponents. Some basic principles:

- All products are tested in a way that does not bias results (for example, multiple operating system and application configurations (stacks) are tested at the same time; there is verification that the attacks are delivered; extensive measures are taken to ensure that threat actors do not blacklist the test network).

- Exploits are validated for efficacy against precisely configured stacks.
- Features of the BPS products are tested as they are used in the real world.
- In order to provide the most actionable information, testing utilizes actual, live attacks from genuine cybercriminals and other threat actors.
- Scoring is based upon observed results; Attack success or failure will be determined by the ability of the BPS to perform kill chain interdiction prior to an attacker successfully gaining command and control of a victim.

See the current Security Stack (Network) Test Methodology for more details.

Performance: The aim of this section is to verify that network-based appliances are capable of detecting and logging breaches when subjected to increasing loads of background traffic up to the maximum bandwidth supported.

Tuning: Each vendor will be expected to be capable of monitoring the same range of operating systems and applications as in the NSS Labs Live Testing™ harness. Vendors will be informed of the specifications of all software and operating systems in the test harness and will be permitted to tune their products or create custom virtual machines (VMs) to model those environments. Vendors will be provided with a baseline sample set of malicious software ahead of testing in order to ensure their products are functioning correctly. This baseline sample set will be used to verify basic detection and performance capabilities only at the start of the test and will not count toward final security effectiveness scores.

NSS Labs test methodologies are continually evolving in response to feedback. If you would like to provide input, please contact advisors@nsslabs.com. For a list of changes, please reference the Change Log in the Appendix.

1.3 Inclusion Criteria

NSS invites all BPS vendors claiming breach prevention capabilities to submit their products at no cost. Vendors with significant market share, as well as challengers with new technology, will be included.

1.4 Deployment

The BPS as a concept covers a system-of-system approach. At a minimum, a BPS consists of a single, network-based appliance (inline or routed). Solutions composed of multiple on-premises boxes, cloud-based components, and/or endpoint agents acting together can be considered breach prevention systems. For BPS of a distributed nature, the ability to log and report to a single, centralized location is key to maintaining operational efficiency in the enterprise.

The test harness and test methodology support testing 10 Gigabit fiber interfaces. All BPS included for testing will be tested up to a maximum of a single 10 Gigabit port pair.

Once installed in the test lab, the BPS will be configured for the use case appropriate to the target deployments (corporate network perimeter and internal segmentation). The BPS must also be configured to block all traffic when resources are exhausted or when traffic cannot be analyzed for any reason.

2 Security Effectiveness

The aim of this section is to verify that the BPS is capable of preventing a breach while remaining resistant to false positives. All test cases in this section are completed with background network load.

A BPS must retain all logs and event information, including information about the malicious activities that enabled systems to achieve conviction events. After *conviction* → *prevention* sequences occur, enterprise SOC analysts investigate events according to their security workflows (e.g., vulnerability assessment, security control improvements, incident archival).

This test utilizes real threats and attack methods that exist in the wild and are actually being used by cybercriminals and other threat actors, based on attacks collected from NSS' global threat intelligence network.

For detail on live testing, please refer to the latest Security Stack (Network) Test Methodology.

2.1 False Positive Testing

This test will include a varied sample of legitimate network traffic, file formats, and executables. The BPS will be expected to ignore all non-malicious traffic. It is imperative that a BPS does not suffer from frequent false-positive events as events generated for non-malicious traffic or samples dilute incident response resources, and, since a BPS is able to block traffic and applications, the usability of the network and access to resources is also at stake.

In addition to known-good samples of common file formats and executables, false-positive testing can include non-malicious samples that may exhibit characteristics commonly, but not uniquely, associated with malicious software.

False-positive scores will be based on the propensity of the BPS to generate an event when analyzing non-malicious files or network traffic.

2.2 Detection and Prevention Engine

The ability of the BPS to detect and block on successful breaches in a timely manner is critical to maintaining the security and functionality of the monitored network. Breach convictions should be reported accurately, giving analysts the opportunity to assess incidents and determine root cause to prevent further infections.

The use of standardized logging and reporting formats, which facilitate the fast and accurate consumption of presented data, is imperative to enable administrators to assess conviction accuracy. The BPS should allow easy generation and exportation of reports, logs, and/or alerts into one or more of these formats:

- CSV
- XML
- JSON
- Other machine-parseable formats may also be acceptable. Please coordinate with NSS to ensure compatibility and adequate capabilities.

As response time is critical in halting the damage caused by an infection or breach, the BPS should be able to block known exploits and malware; detect, analyze, and convict previously unknown malicious software as it is seen traversing a network or at execution time; block command and control (C&C), post-exploitation, or other malicious activity. The BPS is expected to block malware or the malware's post-infection malicious activity. Dispositions from

sandboxes or events generated for malicious activity, which do not result in a block action, but which are generated within 15 minutes of an attack will not count towards the security effectiveness score of the solution. They will however be reflected in the solution's total cost of ownership (TCO). Failure to return a disposition or generate a security-relevant event within 15 minutes of an initial infection attempt will constitute a miss.

Any combination of the following tests may be conducted against each stack, and the BPS will be expected to identify and block the same breach as reported by the NSS Labs Live Testing™ harness:

2.2.1 Exploits

Exploits are attacks against the computer that take advantage of an underlying vulnerability in an application to perform unauthorized tasks. Exploit vectors can include, but are not limited to:

- Drive-by exploits against web browsers, plug-ins, extensions, and add-ons (e.g., Java, Flash)
- Exploits delivered via social engineering—exploits that require user interaction, including but not limited to exploits embedded in documents such as PDF, .docx, HTML).

2.2.2 Malware

Traditional malware requires user interaction to download and install.

- Web browser downloads
- Email attachments

The definition of malware includes so-called "grayware" samples where the overt functionality of the software:

- Has no legitimate purpose (for instance, if it is illegal in a major jurisdiction)
- Exposes the organization to undue risk in terms of regulatory compliance, financial exposure, etc.

2.2.3 Offline and Out-of-Band Infections

Mobile assets owned by the enterprise, such as laptops, may become infected when they are outside the corporate network; e.g., if an employee is working from home or from a public location, or through out-of-band vectors such as malware hosted on removable media (e.g., flash drives, CD-ROM, etc.). Additionally, employees, contractors, or other visitors who attach non-corporate assets to the network may introduce infections that were incurred without being exploited while attached to the enterprise network and behind the protection of the BPS.

The ability of the BPS to identify and block out-of-band infections or post-exploitation activity sourced from endpoints, be these assets enterprise-owned or not, will be assessed. The test may include endpoints that have an endpoint protection agent component of the BPS installed, as well as those simulating non-enterprise assets that would not necessarily have the benefit of such an endpoint protection agent.

The following cases may also be considered:

- The infection is sourced over the network but not from a location where the network component of the BPS has visibility and the ability to block (for example, the user is on public Wi-Fi without having connected to a corporate VPN).
- No network connection was available at the time of infection.

2.3 Physical Access and Malicious Insider Attacks

Breaches may occur as a result of a malicious insider or as a result of an outside attacker gaining physical access to a system. Commercial hardware implants are easily available to enable persistence, load malware, offload documents, and backdoor systems.

Keystroke injection attacks, sideloading of malware, main-in-the-middle attacks, and other techniques, based on physical access to the system/premises will be tested using COTS hardware implants and by living off the land with regards to built-in OS functionality to test the BPS's ability to defend against this category of attack.

Attempts to disable or evade security controls put in place by the BPS by an authorized user and/or local administrator will also be assessed.

2.4 Data Exfiltration

During a breach, one of the primary goals of a threat actor is to exfiltrate sensitive or valuable information. The ability of the BPS to detect common exfiltration techniques will be assessed. A non-exhaustive list of potential techniques includes:

- Use of SSH (SFTP or SCP) over nonstandard ports
- Modified SSH banners
- Exfiltration via DNS query
- Abuse of commodity collaboration tools (email, file sharing, remote access)
- Etc.

2.5 Advanced Attacks Against System Hardware

The ability of advanced threat actors, such as nation states, to exploit flaws in the underlying hardware of systems in order to gain persistent access to a network and remain undetected has been known for decades, but has recently become a major concern for commercial entities.

The ability of the BPS to defend assets against the exploitation of hardware features such as the Intel Management Engine will be assessed.

2.6 Evasions

NSS verifies that the BPS is capable of detecting and blocking basic exploits/drops when subjected to varying common evasion techniques. It is a *requirement* of the test that the submitted BPS has all evasion detection options enabled by default in the shipping product. Wherever possible, a component of the BPS solution is expected to successfully decode the evasive traffic to provide an accurate alert relating to the original exploit, rather than alerting purely on anomalous traffic detected as a result of the evasion technique itself.

Evasions testing will be conducted in accordance with the version of the NSS Labs Evasions Test Methodology currently published at the time BPS testing commences.

3 Performance

This section measures the performance of the BPS 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 whether a particular BPS is appropriate for a given environment.

3.1 Raw Packet Processing Performance (UDP Throughput)

This test uses UDP packets of varying sizes generated by traffic generation appliances. A constant stream of the appropriate packet size—with variable source and destination IP addresses transmitting from a fixed source port to a fixed destination port—is transmitted bi-directionally through each port pair of the BPS. 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. Multiple tests are run and averages are 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. However, 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 are not being "fast-tracked" from the inbound port to the outbound port.

The goal of this test is to determine the raw packet processing capability of each inline port pair of the BPS, as well as its effectiveness at forwarding packets quickly in order to provide the highest level of network performance and with the lowest latency.

3.1.1 64 Byte Packets

Maximum 1,488,000 frames per second per Gigabit of traffic. This test determines the ability of a BPS to process packets from the wire under the most challenging packet processing conditions.

3.1.2 128 Byte Packets

Maximum 844,000 frames per second per Gigabit of traffic

3.1.3 256 Byte Packets

Maximum 452,000 frames per second per Gigabit of traffic

3.1.4 512 Byte Packets

Maximum 234,000 frames per second per Gigabit of traffic. This test provides a reasonable indication of the ability of a BPS to process packets from the wire on an "average" network.

3.1.5 1024 Byte Packets

Maximum 119,000 frames per second per Gigabit of traffic

3.1.6 1514 Byte Packets

Maximum 81,000 frames per second per Gigabit of traffic. This test demonstrates how easy it is to achieve good results using large packets. Readers should use caution when reviewing test results that only quote performance figures using similar packet sizes.

3.2 Latency

The goal of the latency and user response time tests is to determine the effect the BPS has on traffic passing through it under various load conditions. Test traffic is passed across the infrastructure switches and through all inline port pairs of the BPS simultaneously (the latency of the basic infrastructure is known and is constant throughout the tests).

Packet loss and average latency (μs) are recorded for each packet size (64, 128, 256, 512, 1024, and 1514 bytes) at a load level of 95% of the maximum throughput with zero packet loss as previously determined in section 3.1

3.2.1 64 Byte Packets

Maximum 1,488,000 packets per second per Gigabit of traffic

3.2.2 128 Byte Packets

Maximum 844,000 packets per second per Gigabit of traffic

3.2.3 256 Byte Packets

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

3.2.4 512 Byte Packets

Maximum 234,000 packets per second per Gigabit of traffic.

3.2.5 1024 Byte Packets

Maximum 119,000 packets per second per Gigabit of traffic.

3.2.6 1514 Byte Packets

Maximum 81,000 packets per second per Gigabit of traffic.

3.3 Maximum Capacity

The use of traffic generation appliances allows NSS engineers to create true "real-world" traffic at multi-Gigabit speeds as a background load for the tests.

The goal of these tests is to stress the inspection engine and determine how it handles 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 BPS is causing an unacceptable increase in open connections.
- **Excessive concurrent HTTP connections** – Latency within the BPS 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 BPS is causing connections to time out.

3.3.1 Theoretical Maximum Concurrent TCP Connections

This test is designed to determine the maximum concurrent TCP connections of the BPS 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.

An increasing number of 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.

3.3.2 Theoretical Maximum Concurrent TCP Connections with Data

This test is identical to section 3.3.1, except that once a connection has been established, data is transmitted (in 1 KB segments). This ensures that the BPS is capable of passing data across the connections once they have been established.

3.3.3 Maximum TCP Connections per Second

This test is designed to determine the maximum TCP connection rate of the BPS 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.

An increasing number of new sessions are established through the BPS, ramped slowly to determine the exact point of failure. Each session is opened normally, one byte of data is 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.

3.3.4 Maximum HTTP Connections per Second

This test is designed to determine the maximum TCP connection rate of the BPS with a one-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 one-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 upon the request being satisfied; and thus any concurrent TCP connections will be caused purely as a result of latency the BPS introduces on the network. Load is increased until one or more of the breaking points defined earlier is reached.

3.3.5 Maximum HTTP Transactions per Second

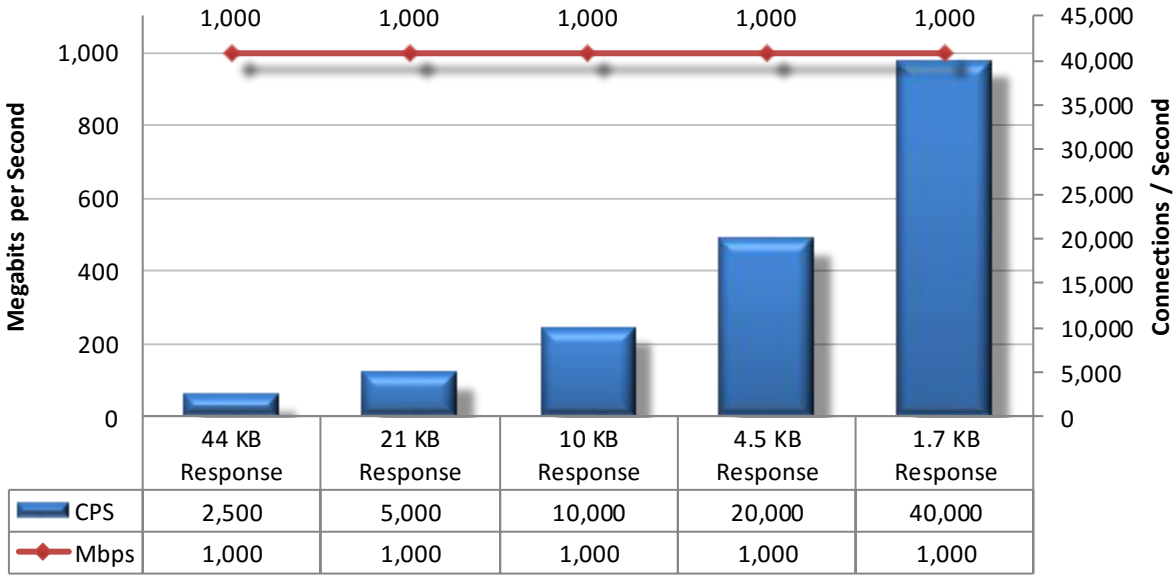
This test is designed to determine the maximum HTTP transaction rate of the BPS with a one-byte HTTP response size. The object size defines the number of bytes contained in the body, excluding any bytes associated with the HTTP header. A one-byte response size is designed to provide a theoretical maximum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

Client and server are using HTTP 1.1 with persistence, and the client will open a TCP connection, send 10 HTTP requests, and close the connection. This ensures that TCP connections remain open until all 10 HTTP transactions are complete, thus eliminating the maximum connection per second rate as a bottleneck (1 TCP connection = 10 HTTP transactions). Load is increased until one or more of the breaking points defined earlier is reached.

3.4 HTTP Capacity with No Transaction Delays

The aim of these tests is to stress the HTTP detection engine and determine how the BPS 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 BPS 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 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.



	44 KB Response	21 KB Response	10 KB Response	4.5 KB Response	1.7 KB Response
CPS	2,500	5,000	10,000	20,000	40,000
Mbps	1,000	1,000	1,000	1,000	1,000

3.4.1 44 KB HTTP Response Size – 2,500 Connections per Second

Maximum 2,500 new connections per second per Gigabit of traffic with a 44 KB HTTP response size—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.

3.4.2 21 KB HTTP Response Size – 5,000 Connections per Second

Maximum 5,000 new connections per second per Gigabit of traffic with a 21 KB HTTP response size—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.

3.4.3 10 KB HTTP Response Size – 10,000 Connections per Second

Maximum 10,000 new connections per second per Gigabit of traffic with a 10 KB HTTP response size—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.

3.4.4 4.5 KB HTTP Response Size – 20,000 Connections per Second

Maximum 20,000 new connections per second per Gigabit of traffic with a 4.5 KB HTTP response size—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.

3.4.5 1.7 KB HTTP Response Size – 40,000 Connections per Second

Maximum 40,000 new connections per second per Gigabit of traffic with a 1.7 KB HTTP response size—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.

3.5 Application Average Response Time: HTTP

Test traffic is passed across the infrastructure switches and through all inline port pairs of the BPS simultaneously (the latency of the basic infrastructure is known and is constant throughout the tests). The results are recorded at each response size (44 KB, 21 KB, 10 KB, 4.5 KB, and 1.7 KB HTTP responses), at a load level of 95% of the maximum throughput with zero packet loss as previously determined in section 3.4.

3.6 HTTP Capacity with HTTP Persistent Connections

The aim of these tests is to determine how the BPS copes with network loads of varying average packet size and varying connections per second while inspecting all traffic. By creating genuine session-based traffic with varying session lengths, the BPS 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 it is possible to achieve in a lab environment, while ensuring absolute accuracy and repeatability.

This test will use HTTP persistent connections, with each TCP connection containing 10 HTTP GETs and associated responses. 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 at various network loads. The stated response size is the total of all HTTP responses within a single TCP session.

3.6.1 250 Connections per Second

This test will simulate HTTP persistent connections, each containing a total of 10 HTTP GET/responses of various sizes. The total HTTP response size for each persistent connection will be equal to four megabits, transmitted over a maximum of 250 connections per second for each gigabit of traffic.

3.6.2 500 Connections per Second

This test will simulate HTTP persistent connections, each containing a total of HTTP 10 GET/responses of various sizes. The total HTTP response size for each persistent connection will be equal to two megabits, transmitted over a maximum of 500 connections per second for each gigabit of traffic.

3.6.3 1,000 Connections per Second

This test will simulate HTTP persistent connections, each containing a total of 10 HTTP GETs/responses of various sizes. The total HTTP response size for each persistent connection will be equal to one megabit, transmitted over a maximum of 1000 connections per second, for each gigabit of traffic.

3.7 "Real-World" Single Application Flows

3.7.1 Single Application SIP flow

3.7.2 Single Application SMTP flow

3.7.3 Single Application SMB flow

3.7.4 Single Application RDP flow

3.7.5 Single Application YouTube flow

3.7.6 Single Application WebEx flow

3.7.7 Single Application BitTorrent flow

3.7.8 Single Application Netflix flow

3.7.9 Single Application SSH flow

4 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 BPS along with its ability to maintain security effectiveness while under normal load and while passing malicious traffic. Systems that are not able to sustain legitimate traffic (or that crash) while under hostile attack will not pass.

The BPS 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 prohibited traffic passes successfully, caused by either the volume of traffic or by the BPS failing open for any reason, this will result in a FAIL.

4.1 Blocking Under Extended Attack

The BPS 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 of 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 BPS for eight hours at a maximum of 100 Mbps, with no additional background traffic. This is not intended as a stress test in terms of traffic load (covered in the previous section); it is merely a reliability test in terms of consistency of blocking performance.

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

4.2 Passing Legitimate Traffic under Extended Attack

This test is identical to test 4.1 where the external interface of the BPS is exposed to a constant stream of exploits over an extended period of time.

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

4.3 Behavior of the State Engine Under Load

This test determines whether the BPS is capable of preserving state across a large number of open connections over an extended period of time.

At various points throughout the test (including after the maximum has been reached), it is confirmed that the BPS is still capable of inspecting and blocking traffic that is in violation of the currently applied security policy, while confirming that legitimate traffic is not blocked (perhaps as a result of exhaustion of the resources allocated to state tables). The BPS must be able to apply policy decisions effectively based on inspected traffic at all load levels.

4.3.1 Attack Detection/Blocking – Normal Load

This test determines if the BPS is able to detect and block policy violations as the number of open sessions reaches 75% of the maximum determined in Test 3.3.1.

4.3.2 State Preservation – Normal Load

This test determines if the sensor maintains the state of pre-existing sessions as the number of open sessions reaches 75% of the maximum determined in Test 3.3.1.

A legitimate HTTP session is opened, and the first packet of a two-packet exploit is transmitted. As the number of open connections approaches the maximum, the initial HTTP session is completed with the second half of the exploit, and the session is closed. If the sensor is still maintaining state on the original session, the exploit will be recorded. If the state tables have been exhausted, the exploit string will be seen as a non-stateful attack and will thus be ignored. Both halves of the exploit are required to trigger an alert. A product will fail the test if it fails to generate an alert after the second packet is transmitted, or if it raises an alert on either half of the exploit on its own.

4.3.3 Pass Legitimate Traffic – Normal Load

This test ensures that the BPS continues to pass legitimate traffic as the number of open sessions reaches 75% of the maximum determined in Test 3.3.1.

4.3.4 State Preservation – Maximum Exceeded

This test determines whether the BPS maintains the state of pre-existing sessions as the number of open sessions exceeds the maximum determined in Test 3.3.1. The method of execution is identical to Test 4.3.2.

4.3.5 Drop Legitimate Traffic – Maximum Exceeded

This test ensures that the BPS continues to drop all traffic as the number of open sessions exceeds the maximum determined in Test 3.3.1.

Note: If a BPS allows traffic to “leak” due to the way it expires old connections, this will result in an automatic fail for the entire test.

4.4 Power Failure and Persistence of Data

Power to the BPS is cut while passing a mixture of legitimate and disallowed traffic. The BPS should retain all configuration data, policy data, and locally logged data once restored to operation following power failure. During power loss, the device should not allow any traffic through, including disallowed traffic.

5 Total Cost of Ownership and Value

Implementation of security solutions can be complex, with several factors affecting the overall cost of deployment, maintenance, and upkeep. All of these 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 updates)
- **Installation –** The time required to take the device out of the box, configure it, deploy it in the network, apply updates and patches, perform initial tuning, and set up desired logging and reporting
- **Upkeep –** The time required to apply periodic updates and patches from vendors, including hardware, software, and firmware updates

Appendix A: Change Log

Version 2.0 – January 2018

- Renamed section 2.2.2 to “Malware”
- Modified section 2.3
- Added sections:
  - 2.3: Physical Access and Malicious Insider Attacks
  - 2.4: Data Exfiltration
  - 2.5: Advanced attacks against hardware systems
- Removed 3.6: HTTP Capacity with Transaction Delays
- Removed 3.8.1: “Real World” Protocol Mix (Enterprise Perimeter)
- Removed 3.8.2: “Real World” Protocol Mix (Education)
- Added sections:
  - 3.7.1: Single Application SIP Flow
  - 3.7.2: Single Application SMTP Flow
  - 3.7.3: Single Application SMB Flow
  - 3.7.4: Single Application RDP Flow
  - 3.7.5: Single Application YouTube Flow
  - 3.7.6: Single Application WebEx Flow
  - 3.7.7: Single Application BitTorrent Flow
  - 3.7.8: Single Application NetFlix Flow
  - 3.7.9: Single Application SSH Flow
- RPC Fragmentation removed from evasions section
- Removed 4.4: Protocol Fuzzing and Mutation
- Changes to wording in the following sections:
  - 1.1: The Need for Breach Prevention
  - 1.4: Deployment
  - 2.1: False Positive Testing
  - 2.2: Detection and Prevention Engine
  - 2.2.2: Malware
- Updated contact information with office address

Version 1.1 – April 2017

- Section 2.3.1: Removed polymorphism and metamorphism from Binary Obfuscation
- Section 2.3.7: HTML Obfuscation: Removed HTTP Evader
- Removed Section 3.3.6: Maximum SSL Handshakes per Second
- Removed Section 3.8: HTTPS Capacity with No Transaction Delay (HTTP Persistent Connections)
- Section 3.9: Replaced Financial Traffic Mix with Education Traffic Mix
- Combined sections 4.5 (Power Failure) and 4.6 (Persistence of Data)

Contact Information

NSS Labs, Inc.
3711 South Mopac Expressway
Building 1, Suite 400
Austin, TX 78746-8022 USA
info@nsslabs.com
www.nsslabs.com