RADWARE DEFENSEPRO 1020

PRODUCT CERTIFICATION

ATTACK MITIGATOR (AM)
METHODOLOGY VERSION: 2.00
APRIL 11, 2008

Published by NSS Labs.

CONTACT:

5115 Avenida Encinas
Suite H
Carlsbad, CA 92008

Tel: +1.847.553.4300
E-mail: info@nsslabs.com
Internet: http://www.nsslabs.com

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

1. The information in this Report is subject to change by NSS Labs without notice.

2. The information in this Report is believed by NSS Labs to be accurate and reliable, but is not guaranteed. All use of and reliance on this Report 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 Report 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 Report 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.

EXECUTIVE SUMMARY

In Q1 of 2008, NSS Labs performed comprehensive testing of the Radware DefensePro 1020 against our Attack Mitigator (AM) testing methodology. This report summarizes the results from the tests performed on the Attack Mitigator system installed in our real-world test lab. This NSS Labs report provides readers with empirically validated evidence about a product's features and capabilities.

Radware provided NSS Labs with a DefensePro 1020 appliance.

Although the DefensePro is a full IPS as well as Attack Mitigator, this test was focussed entirely on the mitigation capabilities of the DefensePro. In this light, security effectiveness was impressive. With a range of innovative technologies under the hood, we found the DefensePro's detection and mitigation capabilities to be excellent. We also found it to be very stable and reliable, coping with our extensive reliability tests with ease and without succumbing to most common evasion techniques.

The DefensePro x-20 range covers from 600Mbps up to 3Gbps. The 1020 is the mid-range offering rated at 1Gbps and offers good performance coupled with low latency under all normal traffic conditions. The DefensePro was tested with 512KB of memory and provided a level of performance which was enough to enable the device to support 1Gbps of traffic on a typical network. This would be further improved with the full complement of 1GB of RAM, allowing it to support more stressful loads with small response sizes.

The APSolute Insite management system has been designed to handle management and configuration of large numbers of sensors across the enterprise. The Java-based console can be slow, particularly when first staring up, but alert handling is powerful and flexible, and the custom reporting is extensive. There are also a number of nice visual touches, such as the radar and map views.

Overall we found the DefensePro 1020 to be a robust and capable Attack Mitigator and believe that it should be on any short list as a candidate for a mitigation solution on the network perimeter. As a result, we are pleased to award NSS Approved to the DefensePro 1020.

NSS Labs
APPROVED

CONTENTS

1 INTRODUCTION

In Q1 of 2008, NSS Labs performed a comprehensive test of the Radware DefensePro 1020 against our Attack Mitigator (AM) v2.00 testing methodology.

This report summarizes the results from over 1200 individual tests and over 4GB of test results collected in our real-world test lab while testing the DefensePro 1020. The NSS Labs test reports are designed to address the challenges faced by IT professionals in selecting security products. This NSS Labs report provides readers with empirically validated evidence about a product’s features and capabilities. Testing and analysis covers several aspects of the security product including:

- ✓ Security Effectiveness
- ✓ Performance
- ✓ Management and Usability

As part of its extensive AM test methodology, NSS Labs subjects each product to a brutal battery of tests that verify the stability and performance of each device tested, determine the accuracy of its security coverage, and ensure that the device will not block legitimate traffic. If a particular AM has been designated as NSS Approved customers can be confident that the device will not significantly impact network performance

To assess the complex matrix of NIPS performance and security requirements, NSS Labs has developed a specialized lab environment that is able to exercise every facet of an AM product. The test suite contains a large variety of individual tests that evaluate the performance, reliability, security effectiveness and usability of AM products, providing the most thorough and complete evaluation of AM products available anywhere today.

NSS Labs AM test methodologies have become the de facto standard for testing in-line AM devices. The NSS Approved logo is often an essential item on the list of requirements when purchasing these products.

2 The Product Under Test

DefensePro is an accelerated hardware-based IPS appliance based on multi layer processing architecture that includes a master CPU, Network Processors and String Match Engine (SME) accelerating deep packet inspection.

It is equipped with a 44gbps non-blocking switching fabric providing 12 Gigabit copper ports with fail-open bypass and 8 Gigabit (GBIC) ports for copper / fibre. The appliance includes 512MB master memory and 1GB network processor memory, and is capable of monitoring up to nine Gigabit network segments simultaneously.

DefensePro is offered with standard built in HA features including zero-power, fail-open bypass for copper ports and dual power supplies. External zero-power fail-open switches for fibre ports are also available as an extra cost option.

The Radware systems were installed and patched with the following:

Radware DefensePro

- Model 1020
- Firmware version 2.30
- Software version 4.00.00

Manufacturer Provided Product Description

DefensePro Sensor

Radware DefensePro is an in-line Intrusion Prevention and Denial-of- Service system that detects and prevents network threats in real-time. DefensePro inspects incoming and outgoing traffic for potential attacks, clearing the network from unwanted malicious traffic. DefensePro also manages bandwidth and establishes traffic shaping rules.

DefensePro’s multi-layer security approach combines a set of features for detecting and mitigating a wide range of network attacks:

- Network-wide Protections:
  - *Behavioral DoS*: Protects against zero-day flood attacks, including SYN Floods, TCP Floods, UDP floods, ICMP and IGMP floods.
  - *Network Anti Scan & Worm Propagation*: Zero-day protection against self propagating worms, horizontal and vertical TCP and UDP scanning and ping sweeps.
- Server Protections:
  - *SYN Protection*: Protects against any type of SYN flood attack using advanced SYN Cookies.

  - *Server Cracking Protection*: Zero-day protection against application vulnerability scanning, brute force and dictionary attacks.
  - *HTTP mitigator*: Mitigates zero-day HTTP Page flood attacks.
  - *Connection Limit*: Protects against session based attacks, such as half open SYN attacks, request attacks and connection attacks.
- Signature-based Protections:
  - *Signature Protection*: Protects against known application vulnerabilities, and common malware, such as worms, trojans, spyware, DoS.
- Stateful Inspection: Ensures that transmission and application stateful rules are enforced based on the protocol RFCs.

DefensePro fully supports IPv4 and IPv6 network inspection and blocks IPv6 attacks as well.

In addition, using DefensePro's Bandwidth Management module, it is possible to define policies to restrict or maintain the bandwidth that can be sent or received by each application, user or segment. Bandwidth Management policies can be configured to guarantee bandwidth for each critical application or limit non critical traffic such as P2P. Rules can also be set to block or allow specific traffic types.

APSolute Insite Management System

APSolute Insite is the management interface for DefensePro. APSolute Insite Management Station is a graphic application that enables you to configure, modify, monitor and generate reports centrally for single or multiple DefensePro deployments. Using APSolute Insite it is possible to:

- Configure ***security policies*** using the *Connect & Protect* table.
- Configure ***bandwidth management*** policies and ***access lists*** in the *BWM & Access Control* table.
- Monitor security event logs and generate network wide reports using the *Security Reporting* window.

Radware Security Update Service on Web

Radware's Security Update Service delivers immediate and ongoing signature updates, protecting against the latest network and application security threats including worms, Trojans, BOTs and application vulnerabilities, to safeguard applications, networks and users.

The Security Update Service consists of the following key service elements:

- 24/7 Security Operations Center (SOC) Scanning: Continuous threat monitoring, detection, risk assessment and filter creation for threat mitigation.
- Emergency Filters: Rapid response filter releases for high impact security events through Emergency Filters.
- Weekly Updates: Scheduled periodic updates to the signature files, with automatic distribution through Radware APSolute Insite, or on-demand download from www.radware.com
- Custom Filters: Custom filters for environment specific threats and newly reported attacks reported to the SOC.

Security Modules

Fuzzy Logic Module - Adaptive Multi-Dimension Decision Engine

When decisions about traffic, users and applications' behavior are to be made, the Fuzzy Logic Module is the main decision engine for these cases. This engine collects traffic characteristics parameters and assigns them an anomaly weight according to an adaptive fuzzy membership function. It then correlates these parameter weights and produces real-time decisions represented by a "degree of attack (or anomaly)" value. Based on these degrees of attack figures, the system is then able to introduce counter-measures that actively repel a perceived threat.

The fuzzy logic algorithm overcomes traffic analysis difficulties that Internet communications usually present. The fuzzy logic algorithm provides a remarkably simple way to draw definite conclusions from vague, ambiguous or imprecise information. Difficulties such as incomplete knowledge on the one hand and noisy signals on the other (something that usually happens when dealing with Internet traffic) are smoothly handled by the fuzzy logic algorithm. Radware has chosen fuzzy logic over other traditional analysis and approximation methods due to the large amount of CPU and memory resources that these methods consume.

The fuzzy algorithm can process a large amount of parameters, decide about their degree of anomaly, correlate between them and reach conclusions in real-time. Fuzzy logic provides a methodology that does an excellent job of balancing significance and precision. Using fuzzy logic as a decision engine, Radware's Network IPS can perform more in-depth traffic analysis and come to conclusions quicker than any other traditional method.

The Fuzzy Logic Module includes adaptive capabilities. As such, the sensitivity of the module is being continuously tuned in order to match the characteristics of the protected network. The adaptive algorithms include IIR (Infinite Impulse Response) filters that continually average traffic parameters and shape the fuzzy logic membership functions accordingly. These capabilities allow the Radware IPS to establish normal behavior baselines according to the date and the time of day.

For each required protection type, the Fuzzy Logic decision collects and learns traffic parameters that are needed in order to best characterize the threat that should be identified and mitigated. Typically, the fuzzy logic decision engine uses two categories of traffic behavioral parameters to generate a degree of attack:

- Rate-based behavioral parameters such as packet rate, Mbps, connection rate, application request rate, application response rate etc.
- Rate invariant behavioral parameters such as protocol break-down, TCP flag distributions, ratio between inbound and outbound traffic, application request/response ratio, connections distribution, URL hits probability functions and more.

In order to eliminate false positive decisions and misdetections, the fuzzy logic engine correlates between both rate and rate-invariant parameters. To illustrate this point, consider the frequent legitimate behavior of a mass crowd entering a news website in an unexpected manner. This behavior immediately causes rate-based behavioral parameters to significantly increase, thus making it look like an anomaly. If the detection engine relies only on rate-based behavioral parameters, this completely legitimate behavior will be flagged as an attack, and will be blocked. However, because rate-invariant parameters will remain unchanged (within certain boundaries) during such legitimate mass crowd behavior, an engine that intelligently correlates between both rate-based and rate-invariant parameters, such as Radware's fuzzy logic engine, will not be susceptible to the aforementioned false positive decision.

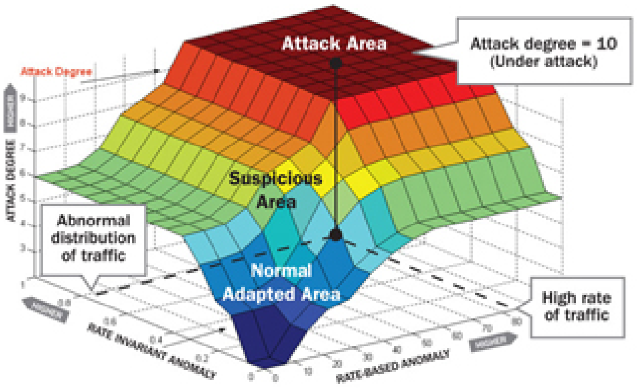


Figure 1: Fuzzy Logic Decision Surface. The XY plane shows the fuzzy input (rate-based input and rate-invariant input). The z-axis represents the degree of attack (or anomaly).

The fuzzy logic decision surface illustrated in the figure 1 above shows a correlation between both rate-based and rate invariant behavioral parameters, before generating a degree of attack. Although in reality the fuzzy logic engine correlates between multiple behavioral parameters, for clarity the figure illustrates a two-dimensional decision surface.

The Fuzzy Logic Module is an adaptive expert system that requires minimal human intervention to configure rules or thresholds. A system that relies upon manually-tuned thresholds and rules produces wildly disparate detection quality, depending mostly on the individual skill level of the system administrator.

Automatic Attack Signature Generation Module

In cases which the attack is unknown or new (the zero-day threat), there is always a great challenge to block the attack without blocking legitimate traffic on the same time. Known attack usually characterized by a well defined content signature that can be used to remove the threat in a surgical manner. However, in the case of zero-day threat, a signature doesn't exist and therefore the technology that detect an anomaly that represent the unknown/new threat should be also capable of characterizing it in a very precise way – in other words, an automatic signature generation technology is needed.

In order to create an attack signature that characterizes the ongoing anomaly without the need for a human research vulnerability group, Radware utilizes probability analysis and closed-feedback loop technology. The following section describes how it works.

For automatic attack signature generation, when the fuzzy logic decision module detects an anomaly, the system activates the attack signature generation mechanism in order to find characteristic parameters of the ongoing anomaly. Working according to a probability theory (a unique patent-pending implementation method that was developed in Radware) that distinguishes between expected and unexpected repetition values of parameters that were studied (statistically) according to the network environment, the signature generation mechanism flags unexpected values as "possible" signatures of attack. Once an attack signature is detected and the anomaly events are confirmed to have a steady behavior, the system transits to a blocking state.

When the system is in the blocking state, it is responsible for optimizing the blocking rules until *the narrowest, but still effective, signature blocking rule is achieved*. This is performed using multiple attack signature parameters including but not limited to the following types:

- Packet checksums
- Packet size
- Packet Identification number
- TTL (Time to Live)
- Fragment offset
- ToS (Type of Service)
- Source/Destination IP address
- Port numbers
- TCP sequence numbers
- HTTP URL's
- SIP URI's (for VoIP anomalies)
- DNS query/Query ID
- DNS Qname (query count)

Each one of the above different signature types can include multiple values, detected by the automatic signature generation mechanism and tailoring them through AND and OR logical relationships.

As far as more AND logical relationships are constructed between different signature type parameters, then the blocking signature is considered to be narrower, thus minimizing the chances of blocking legitimate traffic during attack activities.

Closed Feedback Module

Blocking signatures are dynamically changed according to the nature of the attacks, and in order to achieve the most accurate attack mitigation. When a decision on blocking rules is made, the system checks the affect that these rules will have on traffic behavior. There are three basic feedback cases:

- Positive feedback: If the result is positive, meaning that the traffic anomaly was reduced as a result of the decided blocking signature rules, the system continues to use the same action and tailor more attack characteristic parameters (i.e., signature types) through as many AND logical relationships as possible.
- Negative feedback: If the result is negative, meaning that the degree of traffic anomaly was not changed or was increased, the system stops using the last blocking signature rules and continues to search for more appropriate ones.
- Attack stopped feedback: If the attack stops, then the system will stop all countermeasures immediately.

The main advantage of the system described above is the ability to detect statistical traffic anomalies and create an accurate attack signature-based on heuristic protocol information analysis in less than 20 seconds

Deterministic Security Technology Modules

Still today, many of today's threats simply violating protocols stateful rules, applications rules , or are exploiting application vulnerabilities that are already known and therefore can be precisely removed through a pre-defined attack signature that was the developed by vulnerability research groups or by enforcing deterministic protocol compliancy rules.

Radware's String Match Engine Module

For to the more deterministic types of threats such as known application vulnerabilities exploitation attacks which a signature is already available, a capability to automatically add this attack signature to the system's attack database and to compare it , in real-time, to the network traffic with minimal latency impact should be supported. Radware's hardware accelerated String Match Engine is used for this purpose.

The string match engine is a hardware ASIC-based solution that is capable of multi-gig L7 (application layer) content inspection including inspection of attack signatures that span across multiple packets (i.e., support cross packet inspection) or inspection attack signature that can only be written through regular expressions in order to avoid false sportive or false negative events.

Elimination of False Positive Decisions

Most intelligent systems produce some percentage of false positive decisions. In order to minimize false positives, the system combines deterministic rules with heuristic and adaptive rules. The decision engine correlates between deterministic events, such as those coming from the State Machine Module. Examples of such events include a session's compliance with protocol standards and traffic behavior parameter values being generated by the spectrum analyzer.

In addition, the Closed Feedback Module is also responsible for reducing false positive decisions. When a decision is made to take action against an attack, the Closed Feedback Module checks the results of the action. If the action was successful in reducing the attack, then the system continues to employ the same action. If the action increases the degree of attack or is ineffective in reducing the attack, then the system stops using the action and continues to search for a more appropriate response. When the attack stops, the system stops all countermeasures immediately. The closed feedback operation is done very quickly to minimize the duration that the legitimate address will be blocked unnecessarily (generally less than 1 second).

Another feedback methodology that the system employs is a dynamic blocking period. When the system detects an attack, it initiates a very short blocking period. During this period, the system traces the blocked user and observes their behavior. If subsequent activities represent legitimate network usage, e.g. application recovery from dropped packets, then the system immediately reduces the blocking duration to zero and releases the user. If the user's abnormal activities persist, then the system automatically increases the blocking duration to repel the attack. The dynamic blocking process is performed very quickly, so that normal traffic is not impacted.

3 ATTACK MITIGATOR TEST ENVIRONMENT

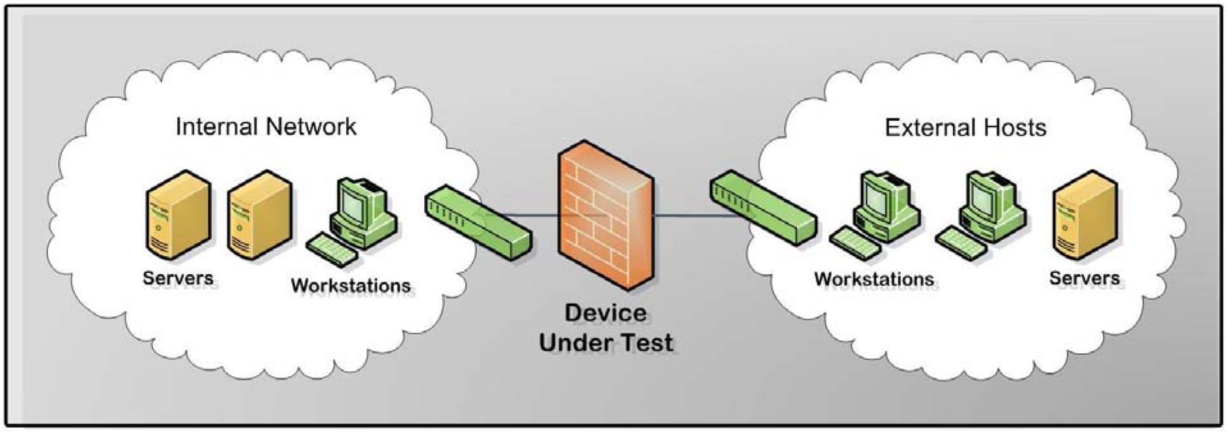
The aim of this procedure is to provide a thorough test of all the main components of an in-line rate-based IPS/Attack Mitigation device in a controlled and repeatable manner and in the most “real world” environment that can be simulated in a test lab.

The Test Environment

The NSS Labs test network is a multi-Gigabit infrastructure based around multiple Cisco Catalyst 6500-series switches (these have a mix of fiber and copper Gigabit interfaces). The NIPS will be configured for the use-case appropriate to the target deployment environment.

Traffic generation equipment - such as the hosts generating exploits, Spirent Avalanche, TestCenter and Smartbits transmit ports - is connected to the “external” network, while the “receiving” equipment - such as the vulnerable hosts for the exploits, Spirent Reflector, TestCenter Spirent Smartbits receive ports - is connected to the internal network. The NIPS 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 will therefore be transmitted through the NIPS, 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 Adtech AX/4000 network monitoring devices are connected. The Adtech AX/4000’s monitor the same mirrored traffic to ensure that the total amount of traffic per in-line port pair never exceeds 1Gbps.



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

4 Results Summary

4.1 Performance

The DefensePro was tested up to 1Gbps, the rated speed of the appliance, over a single port pair. The limit of approximately 22,500 connections per second allowed the device to pass 1Gbps of traffic at all response sizes, and detection and mitigation capabilities were excellent at all loads. We would thus rate the DefensePro 1020 as a true 1Gbps device on any normal network.

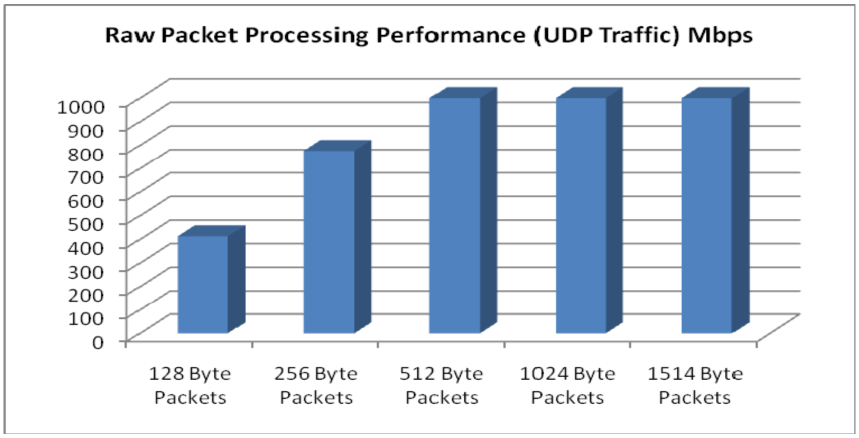


Chart 4.1

Basic latency figures were excellent for a device of this type at almost all traffic loads and packet sizes, ranging from a minimum of 67µs with 250Mbps of 128 byte packets, to a maximum of 128µs with 1Gbps of 1514 byte packets. There was very little variance in latency as load was increased. There was packet loss above 25% load with 128 byte packets and above 75% load with 256 byte packets which prevented us from recording latency figures at these loads. The packet loss with 256 byte packets is disturbing for a Gigabit device.

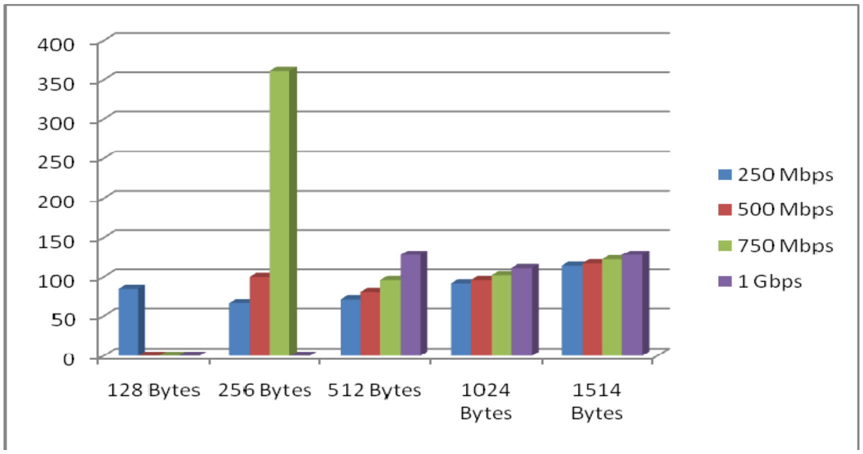


Chart 4.2

Average HTTP response times were also excellent at 195ms, and the device proved capable of a maximum of over 22,000 TCP connections per second, over 92,000 HTTP transactions per second, and over 500,000 concurrent connections. The maximum concurrent connections figure was restricted by the 512KB memory configuration of the DUT. Significantly enhanced performance would be available from the 1GB configuration.

The DefensePro 1020 performed consistently and completely reliably throughout our tests. Under eight hours of extended attack (comprising rate-based attacks mixed with genuine traffic) it continued to block 100 per cent of attack traffic, while passing 100 per cent of legitimate traffic. There almost no increase in user response times as we placed the device under increasing loads of DOS traffic – this is an outstanding feat.

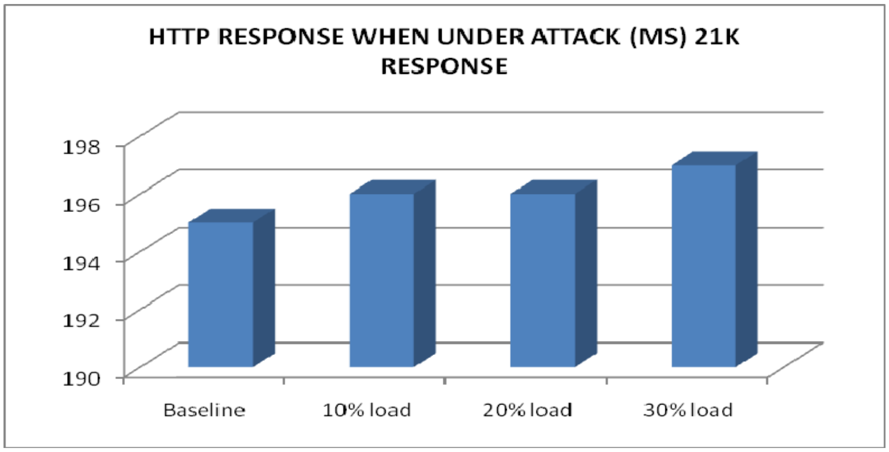


Chart 4.3

Exposing the sensor and management interfaces to traffic subjected to random protocol mutations and fuzzing had no adverse effect, and the device continued to detect and mitigate all other attacks throughout and following the fuzzing process. Attacks were also detected against the management interface.

There was minimal interruption in processing normal traffic during a policy push to the sensor, since the DefensePro is designed to pass traffic without inspection during this process. The amount of time the device fails open is less than 1 millisecond but this is not configurable and, while it is not a serious issue on an Attack Mitigator, it may be a worry to some on an IPS device (the DefensePro performs both functions).

4.2 Security Effectiveness

Attack detection/mitigation was excellent, with the DefensePro detecting and successfully mitigating all of our attacks. Reconnaissance, flooding, Denial of Service (single source and distributed) and protocol fuzzing traffic were all handled effectively by the 1020.

Performance in the high volume detection/mitigation tests was also impeccable across the board, with perfect detection and mitigation at all load levels. Mitigation of the highest levels of DOS and DDOS was almost instantaneous and complete. At the other end of the scale, all of the “low and slow” attacks were detected relatively quickly and also mitigated completely. This level of performance is extremely impressive, and is achieved with virtually no end-user configuration. Radware has done a good job of integrating the old V-Secure technology into its DefensePro device, and increasing the performance in the process, making this one of the best Attack Mitigator devices we have seen in our labs to date.

A major concern in deploying an in-line device is the blocking of legitimate traffic. Once we had configured the appropriate trusted hosts and the device had finished its learning process, the DefensePro completed all our tests without raising a single false positive alert.

Resistance to evasion attempts also proved effective, with the DefensePro successfully detecting all of the fragmented and slow attacks we ran. It would appear to be very difficult to evade this device by simply slowing down port scans and connection floods thanks to the fuzzy logic mechanism employed to compare "*normal*" vs. "*abnormal*" traffic. Employing URL obfuscation techniques during Web vulnerability scans also failed to trick the DefensePro.

4.3 Usability

With multiple methods of managing the device DefensePro offers the administrator a choice of how to approach his management tasks. The system offers a full-featured, text-based command-line interface, a two-tier system using a browser-based interface, and a three-tier system using a Java-based interface.

Our biggest criticism would be that the Java-based Insite management system appears to be extremely resource-hungry, occasionally making it slow. Radware informs us that the UI is undergoing a complete re-write at the time of writing.

DefensePro is straightforward to install and configure and manage on a daily basis. Policy creation is very flexible and powerful, though not always as intuitive as we might like. The biggest issue here is the inability to manage multiple devices simultaneously through native console capabilities. Although all devices can be managed from a central console, it is necessary to connect to each individual device in order to deploy policies. It should be possible to create a single policy for subsequent distribution to multiple DefensePro devices directly within the console. In the current release, this can only be achieved via custom macros. This is a serious omission for an enterprise-class IPS system.

Alert handling is flexible and relatively intuitive, with a wide range of graphing, reporting and analysis functions available to the administrator. Of particular note are the innovative map and "radar" views. We found the dashboard with "radar" display to be both attractive and useful, given that it also supports limited drill-down capabilities.

One thing that is missing is the ability to select rapidly individual elements of the alerts and drill down or up to further analyse the attacks. For example, highlighting the source IP address on one alert, it would be useful to be able to right click, and generate a view of all attacks from that same source IP address. Similar functionality can be achieved via the filtering options, but it is a more lengthy process. One advantage of this approach, however, is that the filters can be saved for later recall.

Reporting is adequate, with high-level graphical views offering the ability to drill down to the detail beneath.

4.4 NSS Test Methodologies

The following chart depicts the PASS/FAIL status of each NSS Labs test. Note that NSS Labs test ID's start with section 5 of this document.

RESULT	Test ID	Description	Comment
	5.1	Detection Engine	
PASS	5.1.1	Ping Sweep	100%
PASS	5.1.2	Port Scanning	100%
PASS	5.1.3	Web Vulnerability Scanning	100%
PASS	5.1.4	Obfuscated Port Scanning	100%
PASS	5.1.5	Denial of Service	100%
PASS	5.1.6	Distributed Denial of Service	100%
PASS	5.1.7	Random Protocol Mutations	100%
	5.2	Resistance to False Positives	
PASS	5.2.1	False Positive Resistance	100%
	5.3	Evasion	
PASS	5.3.1	Baseline attack replay	100%
	5.4	Packet Fragmentation	
PASS	5.4.1	Fragmented UDP Flood (Teardrop)	100%
PASS	5.4.2	Fragmented Stealth Port Scan	100%
PASS	5.4.3	Slow Stealth Port Scan (0.4 secs between packets)	100%
PASS	5.4.4	Very Slow Stealth Port Scan (15 secs between packets)	100%
PASS	5.4.5	Slow Connection Flood (1 second between packets)	100%
PASS	5.4.6	Very Slow Connection Flood (3 seconds between packets)	100%
	5.5	URL Obfuscation	
PASS	5.5.1	URL Encoding - Level 1 (minimal)	100%
PASS	5.5.2	URL Encoding - Level 2	100%
PASS	5.5.3	URL Encoding - Level 3	100%
PASS	5.5.4	URL Encoding - Level 4	100%
PASS	5.5.5	URL Encoding - Level 5	100%
PASS	5.5.6	URL Encoding - Level 6	100%
PASS	5.5.7	URL Encoding - Level 7	100%
PASS	5.5.8	URL Encoding - Level 8 (extreme)	100%
PASS	5.5.9	Premature URL Ending	100%
PASS	5.5.10	Long URL	100%
PASS	5.5.11	Fake Parameter	100%
PASS	5.5.12	TAB Separation	100%
PASS	5.5.13	Case Sensitivity	100%
PASS	5.5.14	Windows \ Delimiter	100%
PASS	5.5.15	Session Splicing	100%
	6	Attack Mitigation Performance	
	6.1	Raw Packet Processing Performance (UDP Traffic)	
PASS	6.1.1	128 Byte Packets	Max 413Mbps
PASS	6.1.2	256 Byte Packets	Max 776Mbps
PASS	6.1.3	512 Byte Packets	Max 1Gbps
PASS	6.1.4	1024 Byte Packets	Max 1Gbps

RESULT	Test ID	Description	Comment
PASS	6.1.5	1514 Byte Packets	Max 1Gbps
	6.2	Maximum Capacity	
PASS	6.2.1	Theoretical Maximum Concurrent TCP Connections	513,026
PASS	6.2.2	Max. Concurrent TCP Connections with 5K Response	512,351
PASS	6.2.3	Max. Concurrent TCP Connections with 21K Response	241,684
PASS	6.2.4	Maximum Concurrent Stateful TCP Connections	513,026
PASS	6.2.5	Max. TCP Connections Per Second (8 byte Response)	22,565
PASS	6.2.6	Max. TCP Connections Per Second (5Kbyte Response)	13,191
PASS	6.2.7	Max. TCP Connections Per Second (21Kbyte Response)	5,000
PASS	6.2.8	Max. HTTP Transactions Per Second (8 Byte Response)	92,071
PASS	6.2.9	Max. HTTP Transactions Per Second (5Kbyte Response)	20,629
PASS	6.2.10	Max. HTTP Transactions Per Second (21Kbyte Response)	5,000
	6.3	HTTP Capacity With No Transaction Delays	
PASS	6.3.1	44Kbyte Response	Max 1Gbps
PASS	6.3.2	21Kbyte Response	Max 1Gbps
PASS	6.3.3	11Kbyte Response	Max 1Gbps
PASS	6.3.4	5Kbyte Response	Max 800Mbps
	6.4	HTTP Capacity With Transaction Delays	
PASS	6.4.1	21Kbyte Response With Delay	Max 1Gbps
PASS	6.4.2	11Kbyte Response With Delay	Max 1Gbps
	6.5	“Real World” Traffic	
PASS	6.5.1	“Real World” Protocol Mix	Max 1Gbps
	6.6	Latency	
PASS	6.6.1	Latency	67-128μs
	6.7		
PASS	6.7.1	Web Response With No Background Traffic (21Kbyte Response)	195ms
PASS	6.7.2	Web Response When Under Attack (10% Load)	196ms
PASS	6.7.3	Web Response When Under Attack (20% Load)	196ms
PASS	6.7.4	Web Response When Under Attack (30% Load)	197ms
	7	Stability & Reliability	
PASS	7.1.1	Blocking Under Extended Attack	
PASS	7.1.2	Passing Legitimate Traffic Under Extended Attack	
PASS	7.1.3	Protocol Fuzzing	
PASS	7.1.4	Protocol Mutation	
PASS	7.1.5	Policy Push	
PASS	7.1.6	Power Fail	
YES	7.1.7	Redundancy	
YES	7.1.8	Fail Open (Power Fail/Reboot)	
PASS	7.1.9	Fail Open (Resource Issues)	
YES	7.1.10	Fail Closed (Power Fail/Reboot)	
PASS	7.1.11	Fail Closed (Resource Issues)	
YES	7.1.12	High Availability (HA) Option (Stateful)	
YES	7.1.13	High Availability (HA) Option (Non-stateful)	

RESULT	Test ID	Description	Comment
PASS	7.1.14	Persistence Of Data	
PASS	7.1.15	IPV6	
	8	Management and Configuration	
	8.1	Management Port	
PASS	8.1.1	Open Ports Detected	
PASS	8.1.2	Open Ports Required	
PASS	8.1.3	Protocol Fuzzing	
YES	8.1.4	Protocol Fuzzing Detection	
	8.2	Management & Configuration - General	
PASS	8.2.1	Transparent Mode	
PASS	8.2.2	Management Port	
PASS	8.2.3	Management Protocol	
PASS	8.2.4	Authentication	
PASS	8.2.5	Enterprise Authentication	
PASS	8.2.6	Direct NIPS Management (Optional)	
PASS	8.2.7	Centralized NIPS Management	
PASS	8.2.8	Pass-Through Mode (Optional)	
PASS	8.2.9	Secure Registration	
PASS	8.2.10	Documentation	
	8.3	Management & Configuration – Policy	
PASS	8.3.1	Sensor Configuration	
PASS	8.3.2	Policy Definition	
PASS	8.3.3	Recommended Settings	
PASS	8.3.4	Bulk Operations	
PASS	8.3.5	Granularity	
PARTIAL	8.3.6	Policy Association	
FAIL	8.3.7	Inheritance	
PASS	8.3.8	Virtualization	
PASS	8.3.9	Policy Deployment	
PASS	8.3.10	Policy Auditing	
FAIL	8.3.11	Policy Version Control	
	8.4	Management & Configuration - Alert Handling	
PASS	8.4.1	Required Log Events	
PASS	8.4.2	Log Location (Optional)	
PASS	8.4.3	Communication Interruption	
PASS	8.4.4	Log Flooding	
PASS	8.4.5	Alerts	
PASS	8.4.6	Alert Accuracy	
PASS	8.4.7	Centralized Alerts	
PASS	8.4.8	Alert Delivery Mechanism	
PASS	8.4.9	Alert Actions (Mandatory)	
FAIL	8.4.10	Alert Actions (Optional)	
PASS	8.4.11	Summarize Alerts	
PASS	8.4.12	View Alert Detail	
PASS	8.4.13	Alert Suppression	
	8.5	Management & Configuration – Reporting	

RESULT	Test ID	Description	Comment
PASS	8.5.1	Centralized Reports	
PASS	8.5.2	Top Attacks	
PASS	8.5.3	Top Sources	
PASS	8.5.4	Top Targets	
PASS	8.5.5	Top Services	
PASS	8.5.6	Top Protocols	
PASS	8.5.7	Custom Reports	
PASS	8.5.8	Saved Reports	
PASS	8.5.9	Scheduled Reports	
PASS	8.5.10	Log File Maintenance	

Table 4.1

5 Security Effectiveness

The aim of this section is to verify that the Attack Mitigator is capable of detecting and blocking a wide range of common rate-based exploits accurately, while remaining resistant to false positives. During the attacks, the victim is expected to remain available and responsive.

All tests in this section are completed with no background network load, and only live exploits/attack tools are used (no replay traffic).

5.1 Detection Engine

While it is not possible to validate completely the entire detection / prevention range of any AM, this test attempts to demonstrate how accurately the AM detects and blocks a wide range of common rate-based attacks, port scans, and Denial of Service attempts.

The AM is installed and all possible detection modes are activated. The vendor is permitted to tune the device (or to configure the device to learn automatically) in order to match the expected loads of attack and background traffic - just as they would for a normal customer. All attacks are run with no load on the network and no IP fragmentation.

The target hosts are placed inline behind the Attack Mitigator and the following rate-based attacks are launched:

5.1.1 Ping Sweep

Sequential and pseudorandom ICMP/Ping scanning at varying rates from a single source targeting multiple protected hosts.

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.1.2 Port Scanning

TCP port scanning based on a variety of SYN rates from a single host.

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.1.3 Web Vulnerability Scanning

Attempts to scan protected Web servers to enumerate potential vulnerabilities.

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.1.4 Obfuscated Port Scanning

TCP port scanning based on non-SYN mechanisms such as ACK and FIN from a limited subset of hosts. Scans are also initiated from pseudorandom sources within the same relative addressing space as if an attacker was attacking from several hosts on the same IP subnet.

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.1.5 Denial of Service

Both stateless and stateful DoS are generated from a single attacker. Attack types include SYN floods, UDP floods, IGMP floods and connection floods, amongst others. Stateless attacks contain TCP and UDP packet blasting. Stateful attacks comprise of protocol specific DoS attacks that target specific services.

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.1.6 Distributed Denial of Service

DDOS is the expansion of the DoS testing suite to use large quantities of attackers. Typical DDoS attacks are sourced from tens of thousands of attackers. Large DDoS attacks are sourced from millions of attackers.

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.1.7 Random Protocol Mutations

Random protocol fuzzing and mutation are generated from a small subset of attackers. Many products and services exhibit instabilities when exposed to unexpected / random protocol content. An Attack Mitigator should block these random protocols from reaching the protected hosts and services while remaining immune to their effects.

PASS	100% of attacks mitigated. All target hosts remain alive and responsive. No instability noted

All the attacks are expected to be reported in as straightforward and clear a manner as possible, and alerts raised in a timely manner.

It is necessary to recognize that different devices detect and mitigate rate-based attacks in different ways. For example, where SYN proxies are utilized, a flood attack could be mitigated instantly with no SYNs reaching the victim, whereas if thresholds are used, some attack packets will inevitably reach the victim before the attack can be mitigated.

Thus, our criteria for determining whether or not an attack has been **successfully** mitigated is as follows:

- The victim remains alive and responsive (i.e. returning Web requests in a timely manner) throughout the attack.
- It is possible to make valid requests to the victim from external hosts **and** (in certain circumstances) from the *apparent* attacking host, and receive responses in a timely manner.

- The attack is detected and mitigated within a reasonable time frame (i.e. it is not allowed to have a detrimental effect on the victim before it is mitigated).
- Once the attack has been detected, no further attack traffic from the attacking host is allowed through for the duration of the test.

5.2 Resistance To False Positives

This test demonstrates the likelihood that a sensor will raise a false positive alert - particularly critical for in-line devices. It is important to note that it is impossible to state definitively whether or not a particular device is susceptible to false positives, since this depends almost entirely on the type of traffic seen in the live deployment.

5.2.1 False Positive Resistance

The network is loaded with a wide range of "normal" network traffic. It is noted how many - if any - false alarms are raised on this traffic once the device has been tuned/configured, and the actions necessary to reduce or eliminate such false positive scenarios are recorded. The product attains a "PASS" for this section if it does not raise an alert and does not block any normal traffic once the initial tuning/learning process has been completed. Raising an alert on any normal traffic once the device has been completely configured is considered a "FAIL", which would indicate the chance that the sensor could block legitimate traffic inadvertently.

PASS	No significant false positive alerts following profiling.

5.3 Evasion

This section verifies that the Attack Mitigator is capable of detecting and mitigating basic rate-based attacks when subjected to varying common evasion techniques.

5.3.1 Baseline Attack Replay

A number of common attacks are executed across the AM to ensure that they are detected in their unmodified state.

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.4 Fragmentation and Timing

These tests determine the ability of the AM to continue to detect common rate-based attacks as the traffic is fragmented and subjected to timing variations.

5.4.1 Fragmented UDP Flood (Teardrop)

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.4.2 Fragmented Stealth Port Scan

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.4.3 Slow Stealth Port Scan (0.4 secs between packets)

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.4.4 Very Slow Stealth Port Scan (15 secs between packets)

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.4.5 Slow Connection Flood (1 second between packets)

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.4.6 Very Slow Connection Flood (3 seconds between packets)

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5 URL Obfuscation

Random URL encoding techniques are employed during the Web vulnerability scans to transform simple URLs to apparently meaningless strings of escape sequences and expanded path characters using a combination of the following techniques:

- Escape encoding (% encoding)
- Microsoft %u encoding
- Path character transformations and expansions (/./ , //, \ )

These techniques are combined in various ways for each URL tested, ranging from minimal transformation, to extreme (every character transformed). All transformed URLs are verified to ensure they still function as expected after transformation.

5.5.1 URL Encoding - Level 1 (minimal)

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.2 URL Encoding - Level 2

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.3 URL Encoding – Level3

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.4 URL Encoding – Level 4

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.5 URL Encoding – Level 5

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.6 URL Encoding – Level 6

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.7 URL Encoding – Level 7

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.8 URL Encoding – Level 8 (Extreme)

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.9 Premature URL Ending

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.10 Long URL

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.11 Fake Parameter

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.12 TAB Separation

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.13 Case Sensitivity

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.14 Windows \ Delimiter

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

5.5.15 Session Splicing

PASS	100% of attacks mitigated. All target hosts remain alive and responsive.

6 Attack Mitigation Performance

Sensors are deployed with all available detection modes enabled. Each sensor is tuned or configured to learn automatically to handle the levels of traffic involved. The “attacker” hosts launch a number of attacks at target hosts on the subnet being protected by the AM. The Adtech network monitors are configured to monitor the switch SPAN ports consisting of normal, exploit and background traffic, and are capable of reporting the total level of attack and/or normal traffic seen across each in-line port pair as verification.

Multiple separate 1Gbps connections will be made from the external to internal switches via the AM. The minimum number of port pairs will be connected to support the claimed maximum bandwidth of the AM (thus an 8 Gbps AM with ten port pairs will have only eight 1Gbps connections tested).

Attacks are launched through the AM against protected hosts with zero background traffic to ensure the AM is capable of detecting the baseline attacks. Once that has been established, increasing levels of varying types of background traffic are generated through the AM in order to determine the point at which the AM begins to fail to mitigate attacks.

All tests are repeated with background traffic levels of 25%, 50%, 75% and 100% of the maximum throughput of the device. At each level, rate-based attacks are launched from the external network and it is verified that they are successfully detected and mitigated. It is also verified that the victim servers remain alive and responsive to legitimate requests from the external network throughout the tests.

At all stages, the Adtech network monitors verify both the overall traffic loading and the level of malicious traffic seen on the target subnet. For each type of background traffic, the maximum load the sensor can sustain before it begins to drop packets/fail to mitigate is also determined.

6.1 Raw Packet Processing Performance (UDP Traffic)

This test uses UDP packets of varying sizes generated by 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 NIPS.

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. Each test is repeated with traffic loads of 25%, 50%, 75% and 100% of the maximum throughput of the NIPS, ans the percentage of attacks detected and blocked is recorded at each load level. Maximum throughput with zero packet loss is also recorded.

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 NIPS, and its effectiveness at passing "useless" packets quickly in order to pass potential attack packets to the detection engine.

6.1.1 128 Byte Packets

Maximum 842,000 Packets Per Second per Gigabit of traffic. Repeated with traffic loads of 25%, 50%, 75% and 100% of maximum throughput of NIPS.

PASS	Load	25%	50%	75%	100%	Max
	Blocked	100%	100%	N/A[1]	N/A[1]	413bps

[1] Device was capable of passing a maximum of 41% of 128 byte packets, preventing us from running the 75/100% load tests

6.1.2 256 Byte Packets

Maximum 452,000 Packets Per Second per Gigabit of traffic. Repeated with traffic loads of 25%, 50%, 75% and 100% of maximum throughput of NIPS.

PASS	Load	25%	50%	75%	100%	Max
	Blocked	100%	100%	100%	N/A[1]	776Mbps

[1] Device was capable of passing a maximum of 78% of 256 byte packets, preventing us from running the 100% load test

6.1.3 512 Byte Packets

Maximum 235,000 Packets Per Second per Gigabit of traffic. Repeated with traffic loads of 25%, 50%, 75% and 100% of maximum throughput of NIPS. This test provides a reasonable indication of the ability of a device to process packets from the wire on an "average" network.

PASS	Load	25%	50%	75%	100%	Max
	Blocked	100%	100%	100%	100%	1Gbps

6.1.4 1024 Byte Packets

Maximum 120,000 Packets Per Second per Gigabit of traffic. Repeated with traffic loads of 25%, 50%, 75% and 100% of maximum throughput of NIPS.

PASS	Load	25%	50%	75%	100%	Max
	Blocked	100%	100%	100%	100%	1Gbps

6.1.5 1514 Byte Packets

Maximum 82,000 Packets Per Second per Gigabit of traffic. Repeated with traffic loads of 25%, 50%, 75% and 100% of maximum throughput of NIPS. This test has been included mainly to demonstrate how easy it is to achieve good results using large packets – beware of test results that only quote performance figures using similar packet sizes.

PASS	Load	25%	50%	75%	100%	Max
	Blocked	100%	100%	100%	100%	1Gbps

6.2 Maximum Capacity

The use of multiple Spirent Communications Avalanche appliances and TestCenter chassis 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 sensor copes 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 - latency within the NIPS is causing unacceptable increase in open connections on the server-side
- Excessive response time for HTTP transactions/SMTP sessions - latency within the NIPS is causing excessive delays and increased response time to client
- Unsuccessful HTTP transactions/SMTP sessions - normally there should be zero unsuccessful transactions. Once these appear, it is an indication that excessive latency within the NIPS is causing connections to time out

6.2.1 Theoretical Maximum Concurrent TCP Connections

This test is designed to determine the maximum concurrent TCP connections of the NIPS with a 8 byte object size. This response size would not typically be found on a normal network, but it provides the means to determine the maximum possible concurrent connections figure.

Client and server are using HTTP 1.0 with keep-alive, 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 ten HTTP transactions are complete, and a "user think time" of 30 seconds between every HTTP Transaction ensures a high number of concurrent connections. Load is increased until one or more of the defined breaking points is reached (the concurrent TCP connections breaking point does not apply to this test). Avalanche load specification is Connections Per Second.

PASS	513,026[1]

[1] State table limit was increased to maximum possible of 700,000 before testing. This is maximum setting with 512KB memory configuration. Enhanced performance would be available from the 1GB memory configuration.

6.2.2 Maximum Concurrent TCP Connections With 5K Response

This test is designed to determine the maximum concurrent TCP connections of the NIPS with a 5Kbyte object size. Test parameters as above.

PASS	512,351

6.2.3 Maximum Concurrent TCP Connections With 21K Response

This test is designed to determine the maximum concurrent TCP connections of the NIPS with a 5Kbyte object size. This is more typical of a “normal” network. Test parameters as above.

PASS	241,684

6.2.4 Maximum Concurrent Stateful TCP Connections

This test is identical to 7.2.1, but is designed to verify the maximum concurrent TCP connections on which the vendor claims the NIPS can maintain state.

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 claimed by the vendor, the initial HTTP session is then 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.

PASS	513,026

6.2.5 Maximum TCP Connections Per Second (8 byte Response)

This test is designed to determine the maximum TCP connection rate of the NIPS with an 8 byte HTTP response size. The response size defines the number of bytes contained in the body, excluding any bytes associated with the HTTP header. An 8 byte response size is designed to provide a theoretical maximum 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 within the NIPS. Load is increased until one or more of the breaking points defined earlier is reached. Avalanche load specification is Connections Per Second.

PASS	22,565

6.2.6 Maximum TCP Connections Per Second (5Kbyte Response)

This test is designed to determine the maximum TCP connection rate of the NIPS with a 4700 byte HTTP response size. The object size defines the number of bytes contained in the body, excluding any bytes associated with the HTTP header. Maximum connections per second is approximately 20,000 per Gigabit of traffic. Test parameters as above.

PASS	13,191

6.2.7 Maximum TCP Connections Per Second (21Kbyte Response)

This test is designed to determine the maximum TCP connection rate of the NIPS with a 21000 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 21Kbyte response size is designed to provide an indication of connections per second rate on a typical network (maximum 5,000 per Gigabit of traffic). Test parameters as above.

PASS	5,000

[1]This is the maximum possible with a Gigabit connection

6.2.8 Maximum HTTP Transactions Per Second (8 Byte Response)

This test is designed to determine the maximum HTTP transaction rate of the NIPS with an 8 byte HTTP response size. The object size defines the number of bytes contained in the body, excluding any bytes associated with the HTTP header. An 8 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. Avalanche load specification is Transactions Per Second.

PASS	92,071

6.2.9 Maximum HTTP Transactions Per Second (5Kbyte Response)

This test is designed to determine the maximum HTTP transaction rate of the NIPS with a 4700 byte HTTP response size. The object size defines the number of bytes contained in the body, excluding any bytes associated with the HTTP header. Maximum connections per second is approximately 20,000 per Gigabit of traffic. Test parameters as above.

PASS	20,629

6.2.10 Maximum HTTP Transactions Per Second (21Kbyte Response)

This test is designed to determine the maximum HTTP transaction rate of the NIPS with a 21000 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 21Kbyte response size is designed to provide an indication of connections per second rate on a typical network (maximum 5,000 per Gigabit of traffic). Test parameters as above.

PASS	5,000

6.3 HTTP Capacity With No Transaction Delays

The aim of these tests is to stress the HTTP detection engine and determine how the sensor copes with detecting and blocking exploits under network loads of varying average packet size and varying connections per second. By creating genuine session-based traffic with varying session lengths, the sensor 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.

6.3.1 44Kbyte Response

Max 2,500 new connections per second per Gigabit of traffic - average packet size 1000 bytes - maximum 120,000 packets per second per Gigabit of traffic. Repeated with background traffic loads of 25%, 50%, 75% and 100% of maximum throughput of NIPS. With relatively low connection rates and large packet sizes, all sensors should be capable of performing well throughout this test.

PASS	Load	25%	50%	75%	100%	Max
	Blocked	100%	100%	100%	100%	1Gbps (2500cps)

6.3.2 21Kbyte Response

Max 5,000 new connections per second per Gigabit of traffic - average packet size 540 bytes - maximum 225,000 packets per second per Gigabit of traffic. Repeated with background traffic loads of 25%, 50%, 75% and 100% of maximum throughput of NIPS. With average connection rates and average packet sizes, this is a good approximation of a real-world production network, and all sensors should be capable of performing well throughout this test.

PASS	Load	25%	50%	75%	100%	Max
	Blocked	100%	100%	100%	100%	1Gbps (5000cps)

6.3.3 11Kbyte Response

Max 10,000 new connections per second per Gigabit of traffic - average packet size 440 bytes - maximum 275,000 packets per second per Gigabit of traffic. Repeated with background traffic loads of 25%, 50%, 75% and 100% of maximum throughput of NIPS. With average packet sizes coupled with very high connection rates this represents a very heavily used production network and is a strenuous test for any sensor.

PASS	Load	25%	50%	75%	100%	Max
	Blocked	100%	100%	100%	100%	1Gbps (10000cps)

6.3.4 5Kbyte Response

Max 20,000 new connections per second per Gigabit of traffic - average packet size 360 bytes - maximum 320,000 packets per second per Gigabit of traffic. Repeated with background traffic loads of 25%, 50%, 75% and 100% of maximum throughput of NIPS. With small packet sizes and extremely high connection rates this is an extreme test for any sensor.

PASS	Load	25%	50%	75%	100%	Max
	Blocked	100%	100%	100%	N/A[1]	800Mbps

						(16000cps)

[1] Unable to run this test above 75% of the maximum load of 1Gbps without connection failures

6.4 HTTP Capacity With Transaction Delays

This is identical to the previous test except that it includes a 10 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.

6.4.1 21Kbyte Response With Delay

Max 5,000 new connections per second per Gigabit of traffic - average packet size 540 bytes - maximum 225,000 packets per second per Gigabit of traffic. Repeated with background traffic loads of 25%, 50%, 75% and 100% of maximum throughput of NIPS. 10 second transaction delay resulting in a maximum of 50,000 open connections during the test. With average connection rates and average packet sizes, this is a good approximation of a real-world production network, and all sensors should be capable of performing well throughout this test.

PASS	Load	25%	50%	75%	100%	Max
	Blocked	100%	100%	100%	100%	1Gbps (5000cps)

6.4.2 11Kbyte Response With Delay

Max 10,000 new connections per second per Gigabit of traffic - average packet size 440 bytes - maximum 275,000 packets per second per Gigabit of traffic. Repeated with background traffic loads of 25%, 50%, 75% and 100% of maximum throughput of NIPS. 10 second transaction delay resulting in a maximum of 100,000 open connections during the test. With average packet sizes coupled with very high connection rates represents a very heavily used production network and is a strenuous test for any sensor.

PASS	Load	25%	50%	75%	100%	Max
	Blocked	100%	100%	100%	100%	1Gbps (10000cps)

6.5 "Real World" Traffic

Whereas 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 (something rarely seen in a real world environment).

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

6.5.1 "Real World" Protocol Mix

Traffic is played across the NIPS comprising the following protocol mix:

79% HTTP
10% SMTP

3% IMAP
5% FTP
3% DNS

HTTP traffic comprises genuine transactions and Web pages from real Web sites such as Google, Yahoo, MSN, NSS Labs, etc. SMTP and IMAP traffic comprises real e-mail messages of varying lengths from the NSS Labs mail server. Maximum 30 simulated users per Gigabit of traffic - 300 connections per second per Gigabit of traffic – 10,000 transactions per second per Gigabit of traffic - 120,000 packets per second per Gigabit of traffic - average packet size of 580 bytes. Repeated with background traffic loads of 25%, 50%, 75% and 100% of maximum throughput of NIPS. Maximum of 5,000 open connections during the test.

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

PASS	Load	25%	50%	75%	100%	Max
	Blocked	100%	100%	100%	100%	1Gbps (10000tps)

6.6 Latency

The aim of the latency and user response time tests is to determine the effect the sensor 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 a SmartBits SMB6000 and TestCenter chassis. The Spirent SmartFlow 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 AM throughput.

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, SmartFlow records the number of packets dropped, together with average and maximum latency.

This test - while not indicative of real-life network traffic - provides an indication of how much the sensor 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.

6.6.1 Latency (RFC 2544)

SmartFlow traffic is passed across the infrastructure switches and through all in-line port pair of the NIPS simultaneously (the latency of the basic infrastructure is known and is constant throughout the tests). The packet loss and average latency (µs) are recorded at each packet size (128, 256, 512, 1024 and 1514 bytes) and each load level from 250Mbps to 1Gbps (in 250Mbps steps).

	Packet size	UDP Traffic Load			
		25%	50%	75%	100%
PASS	128	85	N/A[1]	N/A[1]	N/A[1]
	256	67	100	361	N/A[1]
	512	72	81	96	128
	1024	92	96	102	111
	1514	114	117	122	128

[1] Packet loss prevented accurate latency figures above 250Mbps of 128 byte packets and at 1Gbps of 256 byte Packets

6.7 User Response Times

Spirent Avalanche appliances and TestCenter chassis are used to generate HTTP sessions through the device in order to determine the user experience in terms of failed connections and Web response times.

6.7.1 HTTP Response Time (21Kbyte Response)

Max 5,000 new connections per second per Gigabit of traffic - average packet size 540 bytes - maximum 225,000 packets per second per Gigabit of traffic. Repeated with background traffic loads of 25%, 50%, 75% and 100% of maximum throughput of NIPS.

The average page/URL response time is recorded by Avalanche to provide an indication of the expected response times under normal traffic conditions.

PASS	195ms

6.7.2 HTTP Response When Under Attack (10% load)

HTTP traffic is generated through the sensor as for Test 7.7.1. The Spirent TestCenter is then used to generate SYN flood traffic (from a single source IP) through the sensor at a rate of 10 per cent of the maximum bandwidth of the device under test. Note that with the background traffic, this test will result in a maximum load of 60 per cent of the rated bandwidth of the device under test.

The average page/URL response time is recorded by Avalanche to provide an indication of the expected response times when the device is under attack.

PASS	196ms

6.7.3 HTTP Response When Under Attack (20% load)

HTTP traffic is generated through the sensor as for Test 7.7.1. The Spirent TestCenter is then used to generate SYN flood traffic (from a single source IP) through the sensor at a rate of 20 per cent of the maximum bandwidth of the device under test. Note that with the background traffic, this test will result in a maximum load of 70 per cent of the rated bandwidth of the device under test.

The average page/URL response time is recorded by Avalanche to provide an indication of the expected response times when the device is under attack.

PASS	196ms

6.7.4 HTTP Response When Under Attack (30% load)

HTTP traffic is generated through the sensor as for Test 7.7.1. The Spirent TestCenter is then used to generate SYN flood traffic (from a single source IP) through the sensor at a rate of 30 per cent of the maximum bandwidth of the device under test. Note that with the background traffic, this test will result in a maximum load of 90 per cent of the rated bandwidth of the device under test.

The average page/URL response time is recorded by Avalanche to provide an indication of the expected response times when the device is under attack.

PASS	197ms

7 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 Attack Mitigator along with its ability to maintain security effectiveness while under normal load and While passing malicious traffic. Attack Mitigator products that are not able to sustain legitimate traffic (or crash) while under hostile attack will not receive NSS certification.

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

7.1.1 Blocking Under Extended Attack

The AM is exposed to a constant stream of genuine traffic interspersed with random rate-based attacks over an extended period of time. The device is configured to mitigate and alert.

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 mitigate 100 per cent of the malicious traffic, raising an alert for each type of attack detected. If any recognisable attacks are allowed through the AM - caused by either the volume of traffic or the sensor failing open for any reason - this will result in a FAIL.

PASS	100% of malicious traffic was mitigated

7.1.2 Passing Legitimate Traffic Under Extended Attack:

This test is identical to 8.1.1, where the external interface of the device is exposed to a constant stream of genuine traffic interspersed with random rate-based attacks 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 sensor failing closed for any reason - this will result in a FAIL.

PASS	100% of malicious traffic was mitigated

7.1.3 Protocol Fuzzing

This test stresses the protocol stacks of the AM by exposing it to traffic from various protocol randomizer tools. Several of the tools in this category are based on the ISIC test suite.

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.

PASS	100% of malicious traffic was mitigated. No instability was noted

7.1.4 Protocol Mutation

This test stresses the protocol stacks of the AM by exposing it to traffic from various protocol mutation tools. Several of the tools in this category are based on the Mu Security Analyzer.

PASS	100% of malicious traffic was mitigated. No instability was noted

7.1.5 Policy Push

HTTP traffic is generated through each in-line port pair of the AM up to 50 per cent of the maximum rated bandwidth of the AM - maximum 2,500 new connections per second per Gigabit of traffic - average packet size 540 bytes - maximum 112,500 packets per second per Gigabit of traffic.

A new policy is pushed to the AM during the test, and the minimum, maximum and average page response times and number of failed connections are recorded by Avalanche. These results can be compared with Test 7.3.2 to provide an indication of the effect of pushing policies on legitimate traffic.

PASS	Policy push caused minimal effect on the traffic passing through the sensor

7.1.6 Power Fail

HTTP traffic is generated through each in-line port pair of the AM up to 50 per cent of the maximum rated bandwidth of the AM - maximum 2,500 new connections per second per Gigabit of traffic - average packet size 540 bytes - maximum 112,500 packets per second per Gigabit of traffic.

Power to the AM is cut during the test, and the minimum, maximum and average page response times and number of failed connections are recorded by the Spirent test equipment. If the device is configured to fail open, there should be minimal loss of legitimate sessions throughout the test (over and above the baseline loss expected through switch renegotiation). If the device is configured to fail closed, no traffic should be passed once power has been cut.

PASS	Device fails open on copper ports by default. Optional external hardware bypass unit required for fail-open operation on fiber ports.

7.1.7 Redundancy

Does the AM include multiple redundant critical components (fans, power supplies, hard drive, etc.) (YES/NO/OPTION).

YES	Dual power supplies and fans. Flash memory used instead of hard drives for added resilience.

7.1.8 Fail Open (Power Fail/Reboot)

Does the AM provide the ability to fail open with minimal/zero loss of legitimate traffic (either via built-in, or optional hardware bypass) during power fail and reboot (YES/NO/OPTION).

YES	Device fails open on copper ports by default. Optional external hardware bypass unit required for fail-open operation on fiber ports.

7.1.9 Fail Open (Resource Issues)

Does the AM provide the ability to pass all traffic when resources are exhausted or it is no longer possible to analyse traffic for any reason (i.e. packet rate exceeds device capabilities) .

PASS	DefensePro includes advances traffic overload capabilities for all h/w and s/w modules. The DefensePro internal overload mechanism identifies overload conditions, notifies about them and automatically takes actions that aim to reduce the relevant operations that consume resources. If the overload mechanism cannot prevent the overload conditions (e.g., the system reached its traffic forwarding processing limits) then it automatically triggers the internal by-pass mechanism to pass traffic without inspection. In general the Overload Mechanism reduces the number of new sessions that are sent to overloaded software or hardware components (reduction of the number of sessions is done gradually until overload conditions cease to exist). As a last resort there is also a System Wide Overload where, if all offload operations have failed to prevent the overload condition, then a full-bypass is implemented. It should be noted that in any case, an overload mechanism will compromise some of the product's capabilities (in this case, security will be compromised). Nevertheless, the mechanism is able to tune itself in order to compromise as few security capabilities as possible. The Overload Mechanism is enabled by default.

7.1.10 Fail Closed (Power Fail/Reboot)

Does the AM provide the ability to fail closed during power fail and reboot (YES/NO/OPTION) .

YES	Can be configured to pass zero traffic through the GBIC ports during power fail/reboot.

7.1.11 Fail Closed (Resource Issues)

Does the AM provide the ability to block all traffic when resources are exhausted or it is no longer possible to analyse traffic for any reason (i.e. packet rate exceeds device capabilities) .

PASS	Disable Overload Mechanism. See 8.1.9

7.1.12 High Availability (HA) Option (Stateful)

Is an HA option available for this device, providing **fully** stateful active-active or active-passive failover between devices (YES/NO) .

YES	Active-Passive HA configuration is available as extra cost option

7.1.13 High Availability (HA) Option (Non-stateful)

Is an HA option available for this device, providing any form of failover between devices where existing connections may be lost during failover (YES/NO).

YES	Non-stateful option available

7.1.14 Persistence Of Data

The AM should retain all configuration data, policy data and locally logged data once restored to operation following power failure.

PASS	All configuration data is retained across power cycles

7.1.15 IPV6

The AM should be capable of detecting exploits over both IPV6 and IPV4.

PASS	Both IPV6 and IPV4 traffic can be inspected.

8 Management & Configuration

This section evaluates the features and usability of the Attack Mitigator and associated management infrastructure.

8.1 Management Port

Clearly the ability to manage the alert data collected by the sensor is a critical part of any IPS/Attack Mitigation system. For this reason, an attacker could decide that it is more effective to attack the management interface of the device than the detection interface.

Given access to the management network, this interface is often more visible and more easily subverted than the detection interface, and with the management interface disabled, the administrator has no means of knowing his network is under attack.

8.1.1 Open Ports Required

The vendor will list the open ports and active services on the management interface along with their use.

PASS	Port 161 used for SNMP between ManagePro and the device Port 162 used for SNMP Traps sent from the device to ManagePro Port 2088 Used for additional data collection from the device to ManagePro including attack info and packet reporting. This port is user configurable. Port 2093 used for collecting SRP (Statistics Report Protocol) between the device and ManagePro Port 1167 used for passing traffic between ManagePro and the clients. Port 3306 used for MySQL activities between ManagePro and clients. Port 1306 used for traffic sent from ManagePro clients to server. Port 69 Used for Signature Database upload via TFTP.

8.1.2 Open Ports Detected

The management port will be scanned to determine ports/services visible on the management interface. If any ports additional to those listed in Test 9.1.1 are discovered, this will result in an automatic FAIL.

PASS	Only ports details in 9.1.1 were found to be open

8.1.3 Protocol Fuzzing

This test stresses the protocol stacks of the AM management interface by exposing it to traffic from various protocol randomizer tools. Several of the tools in this category are based on the ISIC test suite

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 (a) operational and capable of detecting and blocking exploits, and (b) capable of communicating in both directions with the management server/console throughout the test.

PASS	NIPS remained stable and continued to block attacks and record alerts throughout the test

8.1.4 Protocol Fuzzing Detection

Are fuzzing attempts attacks detected by the AM even though targeted at the management port (YES/NO).

YES	Some of the protocol mutation/fuzzing attacks are mitigated

8.2 Management & Configuration - General

In addition to the specific tests noted below, NSS will execute an in-depth technical evaluation covering all the main features and benefits of the AM. The accompanying report will fully evaluate each product in terms of ease of use, management and configuration, and alerting and reporting capabilities.

8.2.1 Transparent Mode

The AM must be capable of running in transparent bridge mode, with no IP address assigned to detection ports. Detection ports should ignore all direct connection attempts.

PASS	Each Radware 1020 operates in true transparent bridge mode, meaning the mission interfaces have no IP address visible on the network.

8.2.2 Management Port

The AM should feature a dedicated management port, separate from detection ports. Although this is the preferred configuration, lack of a management port (requiring AM to be managed via one of the detection ports) will not be cause for failure providing management connection and communication is securely encrypted.

PASS	One copper 10/100/1000Mbps port is available on the front panel for management functions.

8.2.3 Management Protocol

Connection from management console to AM should be protected by a minimum of a user name/password combination or multi-factor authentication system, and all communications between should be securely encrypted.

Where a three-tier management architecture is employed, all communication between console and management server(s), and between management server(s) and sensor(s) should be securely encrypted.

PASS	All communication between sensor and management console is encrypted securely. Accessing APSolute Insite ManagePro is only possible using HTTPS. The connection between the management client and ManagePro is secure

	and is established using a TCP connection on port 1306. The connection between ManagePro and Radware devices is established through SNMP. The SNMP connection to the DefensePro device depends on the SNMP (v1 or v3) version supported by the Radware device.

8.2.4 Authentication

Access to management console should be protected by a granular user authentication system which allows for separation of read only and read-write access, preventing users who require reporting access only from modifying device parameters, etc. No access to administrative functions should be permitted (using either direct or centralized administration capabilities) without proper authentication.

PASS	Global read-only or read-write access only per user.

8.2.5 Enterprise Authentication

Access to management console should be protected by a granular user authentication system which allows for restriction of individual users to specific devices, ports, reports, alerts and security policies. Authenticated users should be unable to access devices/ports/policies/alerts/reports/etc. restricted to other users of the system.

PASS	ManagePro supports user access control down to device level with read/write permissions.

8.2.6 Direct Sensor Management (Optional)

Direct access to the AM should be provided (either via command line or Web interface) for single-device management.

PASS	Web interface and CLI available

8.2.7 Centralized Management

A centralized management system should be provided to manage one or more sensors from a single point, including centralized device configuration, policy definition, alert handling and reporting for all sensors under the control of the management system. This should be scalable to large numbers of sensors.

PASS	DefensePro is offered with a 3-tier management solution including: • *DefensePro IPS sensors* • *APSolute Insite ManagePro appliance* • *User clients accessing the ManagePro appliance* Insite ManagePro is Radware's enterprise-grade, security management solution. This appliance acts as a single point of access for IT management and operations staff permitting central management and monitoring of the real-time status of security events of the secured infrastructure.

	A 2-Tier management solution is also available for single or few IPS install base via a stand-alone APSolute Insite management software module.

8.2.8 Pass-Through Mode (Optional)

It should be possible to place the AM into a mode whereby all traffic is allowed to pass through the device, but data will be logged according to the policy in place at the time (thus, the AM will log alerts and state whether the packets would have been dropped, session terminated, etc., but without enforcing those actions on the traffic processed). This should be via a single system-wide operation via the management console or sensor command line (i.e. it must be achieved without affecting the current policy in force).

PASS	Can be activated for the whole system or per protection module

8.2.9 Secure Registration

Initial registration of AM to central management console should be in a fully secure manner (it is permitted to offer a less secure/rapid option, but this should not be the default).

PASS	SNMPV3 is used and can be mandated.

8.2.10 Documentation

Adequate documentation should be provided for both installation, and day-to-day management.

PASS	User guide, Solution Guide (deployment strategy guide) and Technical notes are provided in electronic format. Documentation is thorough and accurate

8.3 Management & Configuration – Policy

8.3.1 Sensor Configuration

The management system should provide the means to configure one or more sensors from a central location, assigning signatures, sensor settings, etc.

PASS	Insite ManagePro provides extensive multi-device management and reporting capabilities via SNMP. Rather than focusing on a single device, ManagePro presents the entire network configuration in a graphical format (the network diagram can be created on-screen), with settings and configuration options organized in a logically related manner. On first entering Insite ManagePro the administrator is presented with a graphical display of the site, which can be populated with icons of switches, routers, and other network elements as well as DefensePro sensors. These can be linked together to highlight physical or logical network links, and any of the DefensePro devices can be managed from here providing the administrator is authorised to do so.

	However, it is still necessary to connect to individual devices, or to use macros, in order to manage them. Nor is it possible to define a single policy and push to all devices simultaneously or in groups without using macros. Site layouts can be saved for later recall.

8.3.2 Policy Definition

The management system should provide the means to define and save multiple security policies, consisting of:

- *General sensor configuration*
- *System-wide parameters*
- *Signatures enabled/disabled*
- *Actions to take when malicious traffic discovered*

PASS	Protection policies are defined in the *Connect & Protect* table. This has a number of rows, giving it the appearance of a typical firewall rules table, and a set of global configuration parameters that apply across all policies. Every row in the *Connect & Protect Table* represents a policy. A security policy contains security profiles that are activated within predefined ranges of ports/VLANs, or within a predefined network, and the scope of each policy can be defined in terms of IP address range, VLAN tag, inbound or outbound traffic, and so on. This gives rise to a very powerful feature of the DefensePro system, since it is possible to define many different policies and have each one apply to only a subset of the protected network (right down to individual hosts, if required).

8.3.3 Recommended Settings

The vendor should provide a default policy or suite of recommended settings which comprises the optimum configuration for a typical network, or the device should be capable of auto-learning its own optimum configuration based on normal traffic patterns

PASS	On top of the existing default settings that are included, the solution guide provides further assistance in the configuration for different network environments

8.3.4 Bulk Operations

Where applicable, it should be possible to search quickly and easily for individual rules and subsequently to apply one or more operations to an entire group in a single operation (for example, to enable or disable a group of rules, or to switch a group from mitigation mode to log mode, etc.)

PASS	Although bulk changes to signature rules are not available in an IPS policy, this does not affect the Attack Mitigator functionality where global policy changes

	are simple to effect.

8.3.5 Granularity

The AM should be capable of blocking or creating exceptions based on IP address, application, protocol, VLAN tag, etc. (i.e. never block HTTP traffic between two specific IP addresses, always block FTP traffic to one specific IP address, etc.).

PASS	The ability to apply multiple protection rules to specific VLANs, IP addresses ranges and even individual hosts provides a high level of granularity.

8.3.6 Policy Association

Once policies have been defined, it should be possible to associate them with specific sensor or groups of sensors.

PARTIAL	There is no way to assign a single policy to multiple devices in the current release without using macros.

8.3.7 Inheritance

It should be possible to create groups and sub-groups of devices such that sub-groups can inherit certain aspects of configuration and policy definition from parent groups.

FAIL	There is no way for one device to inherit policy settings from another in the current release (planned for next release)

8.3.8 Virtualization

Once policies have been defined, it should be possible to associate them with specific "virtual" devices or groups of devices, comprising an entire AM, individual ports, port groups, IP address range, subnet or VLAN.

PASS	Protection rules can be assigned to specific VLANs, IP addresses ranges and individual hosts to provide virtualization capabilities.

8.3.9 Policy Deployment

Once policies have been defined, it should be possible to distribute them to the appropriate device(s), virtual device(s), or groups of devices in a single operation.

PASS	Updating the policy for a device cause it to be written to the device immediately. However, this operation needs to be performed for each device in the system since policies exist only within the devices themselves and not within the management console – thus it is not possible to assign a policy to multiple devices in a single operation. This is not scalable.

8.3.10 Policy Auditing

All changes to policies should be logged centrally. Log data should include at a minimum the date/time the changes were made, and the identity of the user who made them. If possible (OPTIONAL) the system should record the actual changes.

PASS	The following actions are logged (including user name and source IP address): • Any SNMP SET command performed on a device via ManagePro. • Client login date and time. • Date and time of failed client login attempts. • Client disconnection date and time. • Device download/upload configuration. • Device reboot. • Device software upgrades. • ManagePro configuration download/upload.

8.3.11 Policy Version Control

All changes to policies should be recorded by saving a version of the policy before each change. It should be possible to roll-back to a previous version of any policy via a single operation.

FAIL	There is no version control capability in the current release (planned for next release).

8.4 Management & Configuration - Alert Handling

8.4.1 Required Log Events

The AM should record log entries for the following events:

- *Detection of malicious traffic*
- *Termination of a session*
- *Successful authentication by administrator*
- *Unsuccessful authentication by administrator*
- *Policy changed*

- *Policy deployed*
- *Hardware failure*
- *Power cycle*

PASS	Full system-level auditing is available.

8.4.2 Log Location (Optional)

The log events should be logged on the AM initially, in a secure manner, and subsequently transmitted to a central repository for permanent storage.

PASS	Up 1000 event logs can be stored in a local secured database on the sensor before being transmitted to the APSolute Insite station.

8.4.3 Communication Interruption

Where communications between sensor and console/management server are interrupted, storage capacity on the AM should be sufficient to hold one week's worth of log data on a typical network. If it is not possible to restore communication in a timely manner, once the local logs are full, the AM should either (1) continue passing traffic and overwrite oldest log entries, or (2) stop passing traffic. This option should be configurable by the administrator.

PASS	When communications between DefensePro and APSolute Insite are interrupted security events are not logged into Insite station. The device has the capacity to store up 1000 event logs in a local secured database that is managed in cyclic LIFO mode

8.4.4 Log Flooding

Mechanisms should be in place (aggregation) to prevent the AM from flooding the management server/console with too many events of the same type in a short interval. (It should be possible to disable aggregation/flood protection completely for testing purposes to ensure NSS engineers can see every individual alert.)

PASS	The management architecture is based on security events reporting at fixed intervals (default every 5 seconds) and an aggregation threshold limiting the number of events reported per each report cycle (default is 30). Events exceeding the aggregation threshold are reported within one event with a counter of the aggregated events

8.4.5 Alerts

The AM should record log entries each time it detects malicious traffic. At a minimum (depending on protocol), these log entries should contain:

- *Unique event ID*
- *Date and time of event*

- *NIPS ID (includes sensor ID, port ID, etc.)*
- *Direction of traffic (physical/logical source and destination interfaces)*
- *Detection engine which raised the alert (OPTIONAL)*
- *Source IP address*
- *Source port/service (where applicable)*
- *Destination IP address*
- *Destination port/service (where applicable)*
- *ICMP message type and code (where applicable)*
- *Protocol*
- *Unique signature ID*
- *Human-readable description of the event/exploit*
- *CVE reference, Bugtraq ID, or other non-vendor-specific identifier*
- *Action taken by the NIPS (block, log, etc.)*

PASS	Buttons along the top of the Security Reporting screen select views for logs, graphs, split screen (multi-pane), top scans, traffic monitoring, mitigation, map (alerts by geographical location of source) and HTTP traffic. Alerts can be grouped via a hierarchical tree menu in the left hand pane and viewed in a list on the right. Double clicking a list entry calls up individual alert details, including alert type, policy name, date, time, source IP/port, destination IP/port, packet count, bandwidth, action taken, protocol and so on.

8.4.6 Alert Accuracy

The AM should record log entries which are accurate and human readable without having to use additional reference material. The AM should attempt to minimize the number of alerts raised for a single event wherever possible.

PASS	Alert descriptions are generally accurate and easy to read and understand.

8.4.7 Centralized Alerts

Regardless of how many sensors are installed, all alerts should be delivered to, and handled by, a single, central, management console. From that console, it should be possible to view all alerts globally, or select alerts from individual devices (logical or physical).

PASS	All alerts are transmitted directly from each sensor to the APSolute Insite manager console. Alerts can be viewed for all sensor devices from a single console. One or more devices can be selected at a time, and alerts can be viewed in real time or on a historical basis.

8.4.8 Alert Delivery Mechanism

At a minimum, the AM should be able to deliver alerts in a timely manner to a central database for permanent storage, central console for a real-time display, and SMTP server for e-mail alerts.

PASS	Security events are logged to an all-purpose cyclic log file on the sensor. This log file can be obtained at any time, but is of limited size. When the number of entries is beyond the permitted limit, the oldest entries are overwritten. Notifications are raised when the file is 80 per cent utilised, and 100 per cent utilised. Alerts are transmitted from the sensor to the Insite station via SNMP traps (Syslog is also supported). Trap notification is set up through the device's *Target Address* table where the administrator specifies SNMP parameters and selects which type of notification the target server will receive. In the *Community Table*, the administrator can designate that specific users have access to the traps.

8.4.9 Alert Actions (Mandatory)

On detecting malicious traffic, the AM should be able to perform the following actions at a minimum:

- *Ignore*
- *Log only*
- *Mitigate*
- *E-mail administrator*

PASS	For each event there are two possible responses available, including: ▪ *Block and report* ▪ *Report only* DefensePro also supports "dynamic blocking filters" that are generated based on behavioral protections. These filter parameters include : ▪ *Source IP* ▪ *Destination IP* ▪ *Source Port* ▪ *Destination Port* ▪ *Packet ID* ▪ *Packet Size* ▪ *Fragment offset* ▪ *TTL (Time to Live)* ▪ *ToS (Type of Service)* ▪ *TCP Sequence Number* ▪ *TCP Checksum* ▪ *TCP Flags* ▪ *ICMP Checksum*

	▪ *UDP Checksum* ▪ *ICMP Message Type* ▪ *DNS Query* ▪ *DNS Query ID* ▪ *HTTP URL*

8.4.10 Alert Actions (Optional)

On detecting malicious traffic, the AM may optionally be able to perform the following actions:

- *Reconfigure external firewall*
- *Reconfigure switch to isolate/quarantine offending port*
- *Page administrator*

FAIL	No direct communication with 3rd party devices

8.4.11 Summarize Alerts

The central console should provide the ability to select a particular piece of data from an alert and summarize on that data field (i.e. select a source IP address and view all alerts for that source IP). Alternatively, it should be possible to construct data filters manually in a search form and summarize on the specified search criteria. The preferred scenario is to offer both of these options.

PASS	Custom filters can be created to group or select alerts by a specific field. This is a complex process and cannot be accomplished via right clicking on a specific field. However, the resulting filters can be saved for re-use.

8.4.12 View Alert Detail

The central console should provide the ability to select an individual alert and view the following information at a minimum:

- *Detailed alert data (including all data mentioned in Test 9.4.5)*
- *Detailed attack data (i.e. description of the exploit research)*
- *Signature/rule*
- *Remediation data/preventative action*

PASS	Double-clicking any alert entry brings up detailed event information. Right-clicking brings up options to view detailed attack information, research data and so on. The ***Dashboard*** provides a graphical and highly visual real-time and short-term history tool for examining activity in the network. The Dashboard enables the administrator to analyze security events in the network, identify security trends and analyze risk. This view automatically refreshes every 30 seconds providing ongoing real-time analysis of the system. The Security Dashboard also provides a live, moving radar, for monitoring attacks as they occur based

	on their frequency.

8.4.13 Alert Suppression

The central console should provide the ability to create exception filters based on alert data to eliminate further alerts which match the specified criteria (i.e. same alert ID from same source IP). This does not disable detection, logging or blocking, but merely excludes alerts from the console display.

PASS	Can be achieved manually via the use of filters, but is not as intuitive as some other systems we have seen.

8.5 Management & Configuration – Reporting

8.5.1 Centralized Reports

No matter how many sensors are installed, the system should be capable of reporting on all alerts from a single, central, management console. From that console, it should be possible to report all alerts globally, or to report on alerts from individual devices (logical or physical).

PASS	All alerts are reported centrally to APSolute Insite ManagePro server. The user may create user defined reports, executive reports and user defined views, and filtering options are available for every field of the security event logs, along with grouping capability and easy drill-down to the specific event log.

8.5.2 Top Attacks

The system should provide a report listing the top N attacks in the previous hour, day, week, month, year, or custom date range.

PASS	Pre-defined report views are available for attacks over time, top 10 attacks and top 100 attacks.

8.5.3 Top Sources

The system should provide a report listing the top N source IPs from which attacks have been detected in the previous hour, day, week, month, year, or custom date range.

PASS	Custom report

8.5.4 Top Targets

The system should provide a report listing the top N target IPs at which attacks have been launched in the previous hour, day, week, month, year, or custom date range.

PASS	Custom report

8.5.5 Top Services

The system should provide a report listing the top N target ports/services at which attacks have been launched in the previous hour, day, week, month, year, or custom date range.

PASS	Custom report

8.5.6 Top Protocols

The system should provide a report listing the top N protocols over which attacks have been launched in the previous hour, day, week, month, year, or custom date range.

PASS	Custom report

8.5.7 Custom Reports

The report generator should provide the ability to construct complex data filters in a search form and summarize alerts on the specified search criteria.

PASS	Along with predefined reports that provide pre-configured types of network analysis, it is possible to set filtering parameters to create custom reports for viewing attack activity. It is possible to create graphs for high-level views or more detailed drill-down views of network attacks.

8.5.8 Saved Reports

Having defined a custom report filter, it should be possible to save it for subsequent recall.

PASS	Complex filter combinations can be saved as complete custom reports

8.5.9 Scheduled Reports

It should be possible to schedule saved reports for regular unattended runs. The output should be saved as HTML or PDF at a minimum. It should optionally be possible to publish to a central FTP/Web server, and/or e-mail reports to specified recipients.

PASS	Saved reports can be scheduled for one-off or repeated runs. HTML, Excel and PDF output formats are supported, and results can be e-mailed to administrator(s) once reports have run.

8.5.10 Log File Maintenance

The system should provide for automatic rotation of log files, archiving, restoring from archive, and reporting from archived logs.

PASS	The security events are logged into a MySQL database. The administrator can backup the database, export it to other management facilities or save copies for forensics. It is also possible to import old log files to the system for analysis

9 Appendix A: Test Infrastructure

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

SPIRENT® Communications

Mu Security™

assurent secure technologies | a TELUS Company

CORE SECURITY TECHNOLOGIES

karalon

DELL

metasploit

<<back|track 2