Top Layer IPS 5500 V3.3

Technical Evaluation

An NSS Group Report

First published January 2005 (Version 1.0)

Published by The NSS Group
Security Testing Laboratories
Mas la Carrière, Route de Ganges
30440 Sumène, France

Tel : +33 (0)4 67 81 49 11
E-mail : info@nss.co.uk
Internet : http://www.nss.co.uk

TABLE OF CONTENTS

INTRODUCTION 1

Intrusion Prevention Systems (IPS) 1
Host IPS (HIPS) 2
Network IPS (NIPS) 2
Rate-Based IPS (Attack Mitigator) 3
Implementation Challenges 3
Requirements for effective prevention 5
The NSS Intrusion Prevention Group Test 6
Performance 7
Security Effectiveness 10
Usability 12

TOP LAYER IPS 5500 V3.3 13

Executive Summary 13
Architecture 13
Management Application 14
Central Management System (CMS) 15
IPS 5500 Appliance 15
High Availability 17
Performance 18
Content-Based 18
Rate-Based (Attack Mitigation) 19
Security Effectiveness 19
Content-Based 19
Rate-Based (Attack Mitigation) 20
Usability 21
Installation 21
Configuration 23
Policy Management 25
Alert Handling 32
Reporting and Analysis 34
Verdict 37
Contact Details 40

APPENDIX A – TEST RESULTS (CONTENT-BASED) 41

The Test Environment 41
Section 1 – Detection Engine 41
Section 2 – Evasion 43
Section 3 – Stateful Operation 45
Section 4 – Detection/Blocking Performance Under Load 47
Section 5 – Latency & User Response Times 51
Section 6 – Stability & Reliability 53
Section 7 – Management and Configuration 53
Top Layer IPS 5500 V3.3 Test Results (Content-based) 55
Section 1 - Detection Engine 55
Section 2 - IPS Evasion 55
Section 3 - Stateful Operation 57
Section 4 - Detection/Blocking Performance Under Load 57
Section 5 - Latency & User Response Times 58
Section 6 - Stability & Reliability 58
Section 7 - Management Interface 58

APPENDIX B – TEST RESULTS (RATE-BASED) 62

The Test Environment 62
Section 1 – Detection Engine 62
Section 2 – Evasion 65
Section 3 – Attack Mitigation Performance Under Load 66
Section 4 – Latency & User Response Times 70
Section 5 – Stability & Reliability 72
Section 6 – Management and Configuration 73
Top Layer IPS 5500 V3.3 Test Results (Rate-based) 75
Section 1 - Detection Engine 75
Section 2 - Evasion Techniques 75
Section 3 - Detection/Mitigation Performance Under Load 76
Section 4 - Latency & User Response Times 77
Section 5 - Stability & Reliability 77
Section 7 - Management Interface 77

TABLE OF FIGURES

Figure 1 - Top Layer: The IPS 5500 ports 16
Figure 2 - Top Layer: IPS 5500 architecture 17
Figure 3 - Top Layer: The IPS 5500 appliance 22
Figure 4 - Top Layer: Getting Started Wizard 22
Figure 5 - Top Layer: Management Application Navigation Tree 24
Figure 6 - Top Layer: Policy configuration 26
Figure 7 - Top Layer: Defining services 27
Figure 8 - Top Layer: Filter configuration 28
Figure 9 - Top Layer: Defining Rules 30
Figure 10 - Top Layer: Defining Rule Set Actions 31
Figure 11 - Top Layer: Applying Rule Sets 32
Figure 12 - Top Layer: Monitoring Blocked and Detected Attacks 33
Figure 13 - Top Layer: Monitoring dropped packets 34
Figure 14 - Top Layer: Security Report 35
Figure 15 - Top Layer: Monitoring current connections 36

The NSS Group

The NSS Group is the world's foremost independent security testing facility.

With British headquarters, and security and network infrastructure testing facilities in the South of France, The NSS Group offers a range of specialist IT, networking and security-related services to vendors and end-user organisations world-wide.

The NSS Group's Security Testing Laboratories are available to vendors and end-users for fully independent testing of networking, communications and security hardware and software.

The NSS Group also operates certification schemes for vendors and certification bodies, and currently provides evaluation and certification of a wide range of security products, including IDS/IPS appliances, firewalls, VPNs, Web Application firewalls, multi-function security appliances, cryptographic devices and PKI products.

Output from the labs, including detailed research reports, articles and white papers on the latest network and security technologies, are made available on the NSS web site at http://www.nss.co.uk.

The NSS Group awards are recognised world-wide as being the most desirable and essential when it comes to security products. Vendors consider the awards to be a crucial step in any security-related marketing campaign, whilst feedback from readers of the reports indicates that participation in an NSS Group test and/or one of the **NSS Approved** awards is a prerequisite for any security product in order to be considered for purchase.

Foreword

Following the huge success the first comprehensive *Intrusion Prevention System* (IPS) test of its kind, The NSS Group is pleased to present the results of its second IPS Group Test which includes a number of new products not included in the first report.

As with Edition 1, this exhaustive review will give readers a complete perspective of the capabilities, maturity and suitability for immediate deployment of each of the products tested. The NSS Group established this test as IPS products are being actively deployed as a new layer in defence-in-depth security architectures.

It is interesting to note that between publishing Edition 1 and Edition 2 the analyst groups who were previously so sure that IDS was dead and IPS stillborn have now come around to our way of thinking - while the so-called "*deep inspection firewalls*" are not ready for prime-time deployments, security administrators need to make the best use of the technology that **is** available, and for now that means a combination of firewalls, in-line intrusion prevention devices, and intrusion detection systems. They are likely to be in use for quite some time to come, too!

The NSS IPS Group Test evaluates the performance, reliability, security effectiveness, and usability of Network IPS products. The test consists of seven sections within three primary areas: *performance and reliability, security accuracy,* and *usability*.

Overall, the brand new test suite contains over **800 individual tests**, many of which are run multiple times, to provide the most thorough and complete evaluation of IPS products available anywhere today. This edition also sees the introduction of a new *Rate-Based IPS* methodology to complement our exiting *Content-Based IPS* methodology used in Edition 1. This has allowed us to more accurately test Rate-Based/Attack Mitigation products, and two devices were tested against this new methodology in the latest report (one of them actually tested against **both** methodologies – a first).

It is worth pointing out that not every product submitted for testing receives an *NSS Approved* award. Standards are very high, and out of nine products signed up for this group test initially, only the five included in the final Edition 2 report have received ***NSS Approved*** awards.

We believe that our IPS test methodologies - which have been updated for this test - will become the *de facto* standard for testing in-line Intrusion Prevention/Attack Mitigation devices, and the *NSS Approved* logo an essential item on the list of requirements when purchasing these products.

We also believe that this report is essential reading for anyone considering deploying Intrusion Prevention Systems in their networks, either in a test or live situation, and we hope that you find it both informative and useful in making your purchasing decisions. The **IPS Group Test (Edition 2)** report can be viewed on-line at *www.nss.co.uk/ips*.

Bob Walder

INTRODUCTION

In a survey commissioned by VanDyke Software, some 66 per cent of the companies who responded said that they perceive system penetration to be the largest threat to their enterprises.

The survey revealed that the top eight threats experienced by those surveyed were *viruses* (78 per cent of respondents), *system penetration* (50 per cent), *DoS* (40 per cent), *insider abuse* (29 per cent), *spoofing* (28 per cent), *data/network sabotage* (20 per cent), and *unauthorised insider access* (16 per cent).

Although 86 per cent of respondents use firewalls (a disturbingly **low** figure in this day and age, to be honest!), it is apparent that firewalls are not always effective against many intrusion attempts. The average firewall is designed to deny clearly suspicious traffic - such as an attempt to telnet to a device when corporate security policy forbids telnet access completely - but is also designed to allow some traffic through - Web traffic to an internal Web server, for example.

The problem is, that many exploits attempt to take advantage of weaknesses in the very protocols that **are** allowed through our perimeter firewalls, and once the Web server has been compromised, this can often be used as a springboard to launch additional attacks on other internal servers. Once a “rootkit” or “back door” has been installed on a server, the hacker has ensured that he will have unfettered access to that machine at any point in the future.

Firewalls are also typically employed only at the network perimeter. However, many attacks, intentional or otherwise, are launched from within an organisation. Virtual private networks, laptops, and wireless networks all provide access to the internal network that often bypasses the firewall. Intrusion detection systems may be effective at detecting suspicious activity, but do not provide *protection* against attacks. Recent worms such as Slammer and Blaster have such fast propagation speeds that by the time an alert is generated, the damage is done and spreading fast.

Intrusion Prevention Systems (IPS)

The inadequacies inherent in current defences has driven the development of a new breed of security products known as *Intrusion Prevention Systems* (IPS). This is a term which has provoked some controversy in the industry since some firewall and IDS vendors think it has been “hijacked” and used as a marketing term rather than as a description for any kind of new technology.

Whilst it is true that firewalls, routers, IDS devices and even AV gateways all have intrusion prevention technology included in some form, we believe that there are sufficient grounds to create a new market sector for true *Intrusion Prevention Systems*.

These systems are proactive defence mechanisms designed to detect malicious packets within normal network traffic (something that the current breed of firewalls do not actually do, for example) and stop intrusions dead, blocking the offending traffic automatically before it does any damage rather than simply raising an alert as, or after, the malicious payload has been delivered.

Within the IPS market place, there are two main categories of product: *Host IPS* and *Network IPS*, with the latter being further sub-divided into *Content-Based* and *Rate-Based* (or *Attack Mitigation*) systems.

Host IPS (HIPS)

As with Host IDS systems, the Host IPS relies on agents installed directly on the system being protected. It binds closely with the operating system kernel and services, monitoring and intercepting system calls to the kernel or APIs in order to prevent attacks as well as log them.

It may also monitor data streams and the environment specific to a particular application (file locations and Registry settings for a Web server, for example) in order to protect that application from generic attacks for which no “signature” yet exists.

One potential disadvantage with this approach is that, given the necessarily tight integration with the host operating system, future OS upgrades could cause problems.

Since a Host IPS agent intercepts all requests to the system it protects, it has certain prerequisites - it must be very reliable, must not negatively impact performance, and must not block legitimate traffic. Any HIPS that does not meet these minimum requirements should never be installed in a host, no matter how effectively it blocks attacks.

Network IPS (NIPS)

The Network IPS combines features of a standard IDS, an IPS and a firewall, and is sometimes known as an *In-line IDS* or *Gateway IDS (GIDS)*. The next-generation firewall - the *deep inspection firewall* - also exhibits a similar feature set, though we do not believe that the deep inspection firewall is ready for mainstream deployment just yet.

As with a typical firewall, the NIPS has at least two network interfaces, one designated as *internal* and one as *external*. As packets appear at the either interface they are passed to the detection engine, at which point the IPS device functions much as any IDS would in determining whether or not the packet being examined poses a threat.

However, if it should detect malicious traffic, in addition to raising an alert, it will discard the packet(s) and mark that flow as bad. As the remaining packets that make up that particular TCP session arrive at the IPS device, they are discarded immediately.

Legitimate packets are passed through to the second interface and on to their intended destination. A useful side effect of some NIPS products is that as a matter of course - in fact as part of the initial detection process - they will provide “*packet scrubbing*” functionality to remove protocol inconsistencies resulting from varying interpretations of the TCP/IP specification (or intentional packet manipulation).

Thus any fragmented packets, out-of-order packets, or packets with overlapping IP fragments will be re-ordered and “cleaned up” before being passed to the destination host, and illegal packets can be dropped completely.

One thing to watch out for - don't let the "reactive" IDS vendors kid you into believing that they have *intrusion prevention* capabilities just because they can send TCP reset commands or re-configure a firewall when they detect an attack (a worrying piece of FUD that we have noticed in some IDS marketing literature recently).

The problem here is that unless the attacker is operating on a 2400 baud modem, the likelihood is that by the time the IDS has detected the offending packet, raised an alert, and transmitted the TCP Resets - and especially by the time the two ends of the connection have received the Reset packets and acted on them (or the firewall or router has had time to activate new rules to block the remainder of the flow) - the payload of the exploit has long since been delivered….. *game over*! Our guess is that there are not many crackers using 2400 baud modems these days….

A true IPS device, however, is sitting in-line - **all** the packets have to pass through it. Therefore, as soon as a suspicious packet has been detected - and **before** it is passed to the internal interface and on to the protected network, it can be dropped. Not only that, but now that flow has been flagged as suspicious, **all** subsequent packets that are part of that session can also be dropped with very little additional processing. Oh, and for good measure, some products are also capable of sending *TCP Resets* or *ICMP Unreachable* messages to the attacking host.

Rate-Based IPS (Attack Mitigator)

Most NIPS products are basically IDS engines that operate in-line, and are thus dependent on protocol analysis or signature matching to recognise malicious content within individual packets (or across groups of packets). These can be classed as *Content-Based IPS* systems.

There is, however, a second breed of Network IPS that ignores packet content almost completely, instead monitoring for anomalies in network traffic that might characterise a flood attempt, scan attempt, and so on. These devices are capable of monitoring traffic flows in order to determine what is considered "normal", and applying various techniques to determine when that traffic deviates from normal. This is not always as simple as watching for high-volumes of a specific type of traffic in a short space of time, since they must also be capable of detecting "stealth "attacks, such as low-rate connection floods and slow port scan attempts.

Since these devices are concerned more with anomalies in traffic flow than packet contents, they are classed as *Rate-Based IPS* systems - and are also known as *Attack Mitigators*, as they are so effective against DOS and DDOS attacks.

Implementation Challenges

There are a number of challenges to the implementation of an IPS device that do not have to be faced when deploying passive-mode IDS products. These challenges all stem from the fact that the IPS device is designed to work in-line, presenting a potential choke point and single point of failure.

If a passive IDS fails, the worst that can happen is that some attempted attacks may go undetected. If an in-line device fails, however, it can seriously impact the performance of the network.

Perhaps latency rises to unacceptable values, or perhaps the device fails closed, in which case you have a self-inflicted Denial of Service condition on your hands. On the bright side, there will be no attacks getting through! But that is of little consolation if none of your customers can reach your e-commerce site.

Even if the IPS device does not fail altogether, it still has the potential to act as a bottleneck, increasing latency and reducing throughput as it struggles to keep up with up to a Gigabit or more of network traffic. Devices using off-the-shelf hardware will certainly struggle to keep up with a heavily loaded Gigabit network, especially if there is a substantial signature set loaded, and this could be a major concern for both the network administrator - who could see his carefully crafted network response times go through the roof when a poorly designed IPS device is placed in-line - as well as the security administrator, who will have to fight tooth and nail to have the network administrator allow him to place this unknown quantity amongst his high performance routers and switches.

As an integral element of the network fabric, the Network IPS device must perform much like a network switch. It must meet stringent network performance and reliability requirements as a prerequisite to deployment, since very few customers are willing to sacrifice network performance and reliability for security. A NIPS that slows down traffic, stops good traffic, or crashes the network is of little use.

Dropped packets are also an issue, since if even one of those dropped packets is one of those used in the exploit data stream it is possible that the entire exploit could be missed. Most high-end IPS vendors will get around this problem by using custom hardware, populated with advanced FPGAs and ASICs - indeed, it is necessary to design the product to operate as much as a switch as an intrusion detection and prevention device.

It is very difficult for any security administrator to be able to characterise the traffic on his network with a high degree of accuracy. What is the average bandwidth? What are the peaks? Is the traffic mainly one protocol or a mix? What is the average packet size and level of new connections established every second - both critical parameters that can have detrimental effects on some IDS/IPS engines? If your IPS hardware is operating “on the edge”, all of these are questions that need to be answered as accurately as possible in order to prevent performance degradation.

Another potential problem is the good old *false positive*. The bane of the security administrator’s life (apart from the script kiddie, of course!), the false positive rears its ugly head when an exploit signature is not crafted carefully enough, such that legitimate traffic can cause it to fire accidentally. Whilst merely annoying in a passive IDS device, consuming time and effort on the part of the security administrator, the results can be far more serious and far reaching in an in-line IPS appliance.

Once again, the result is a self-inflicted Denial of Service condition, as the IPS device first drops the “offending” packet, and then potentially blocks the entire data flow from the suspected hacker. If the traffic that triggered the false positive alert was part of a customer order, you can bet that the customer will not wait around for long as his entire session is torn down and all subsequent attempts to reconnect to your e-commerce site (if he decides to bother retrying at all, that is) are blocked by the well-meaning IPS.

Another potential problem with any Gigabit IPS/IDS product is, by its very nature and capabilities, the amount of alert data it is likely to generate. On such a busy network, how many alerts will be generated in one working day? Or even one hour? Even with relatively low alert rates of ten per second, you are talking about 36,000 alerts every hour. That is 864,000 alerts each and every day. The ability to tune the signature set accurately is essential in order to keep the number of alerts to an absolute minimum. Once the alerts have been raised, however, it then becomes essential to be able to process them effectively. Advanced alert handling and forensic analysis capabilities - including detailed exploit information and the ability to examine packet contents and data streams - can make or break a Gigabit IDS/IPS product.

Of course, one point in favour of IPS when compared with IDS is that because it is designed to prevent the attacks rather than just detect and log them, the burden of examining and investigating the alerts - and especially the problem of rectifying damage done by successful exploits - is reduced considerably.

Requirements for effective prevention

Having pointed out the potential pitfalls facing anyone deploying these devices, what features are we looking for that will help us to avoid such problems?

- **In-line operation -** only by operating in-line can an IPS device perform true protection, discarding all suspect packets immediately and blocking the remainder of that flow
- **Reliability and availability -** should an in-line device fail, it has the potential to close a vital network path and thus, once again, cause a DoS condition. An extremely low failure rate is thus very important in order to maximise up-time, and if the worst should happen, the device should provide the option to fail open or support fail-over to another sensor operating in a fail-over group (see below). In addition, to reduce downtime for signature and protocol coverage updates, an IPS must support the ability to receive these updates without requiring a device re-boot. When operating inline, sensors rebooting across the enterprise effectively translate into network downtime for the duration of the reboot
- **Resilience -** as mentioned above, the very minimum that an IPS device should offer in the way of High Availability is to fail open in the case of system failure or power loss (some environments may prefer this default condition to be “fail closed” as with a typical firewall, however - the most flexible products will allow this to be user-configurable). Active-Active stateful fail-over with cooperating in-line sensors in a fail-over group will ensure that the IPS device does not become a single point of failure in a critical network deployment
- **Low latency -** when a device is placed in-line, it is essential that its impact on overall network performance is minimal. Packets should be processed quickly enough such that the overall latency of the device is as close as possible to that offered by a layer 2/3 device such as a switch, and no more than a typical layer 4 device such as a firewall or load-balancer.
- **High performance -** packet processing rates must be at the rated speed of the device under real-life traffic conditions, and the device must meet the stated performance with all signatures enabled.

Headroom should be built into the performance capabilities to enable the device to handle any increases in size of signature packs that may occur over the next three years. Ideally, the detection engine should be designed in such a way that the number “signatures” (or “checks”) loaded does not affect the overall performance of the device.

- **Unquestionable detection accuracy -** it is imperative that the quality of the signatures is beyond question, since false positives can lead to a Denial of Service condition. The user MUST be able to trust that the IDS is blocking only the user selected malicious traffic. New signatures should be made available on a regular basis, and applying them should be quick (applied to all sensors in one operation via a central console) and seamless (no sensor reboot required)
- **Fine-grained granularity and control -** fine grained granularity is required in terms of deciding exactly which malicious traffic is blocked. The ability to specify traffic to be blocked by attack, by policy, or right down to individual host level is vital. In addition, it may be necessary to only alert on suspicious traffic for further analysis and investigation
- **Advanced alert handling and forensic analysis capabilities -** once the alerts have been raised at the sensor and passed to a central console, someone has to examine them, correlate them where necessary, investigate them, and eventually decide on an action. The capabilities offered by the console in terms of alert viewing (real time and historic) and reporting are key in determining the effectiveness of the IPS product.

The NSS Intrusion Prevention Group Test

The NSS Group conducted the first comprehensive IPS test of its kind, now updated in this Edition. This exhaustive review will give readers a complete perspective of the capabilities, maturity and suitability of the products tested for their particular needs.

As part of its extensive IPS/Attack Mitigator test methodologies (see section on *Testing Methodology* later in this report for detailed methodologies, updated for this latest test) The NSS Group subjects each product to a brutal battery of tests that verify the stability and performance of each IPS tested, determine the accuracy of its security coverage, and ensure that the device will not block legitimate traffic.

If a particular IPS has been designated as *NSS Approved*, customers can be confident that the device will not significantly impact network/host performance, cause network/host crashes, or otherwise block legitimate traffic.

To assess the complex matrix of IPS/Attack Mitigator performance and security requirements, the NSS Group has developed a specialised lab environment that is able to exercise every facet of an IPS product. The test suite contains over 800 individual tests that evaluate IPS products in three main areas: *performance and reliability, security accuracy,* and *usability*.

This thorough review should give readers a complete perspective of the capabilities, maturity and suitability of the products tested for their particular needs.

Performance

Any IPS is expected to be reliable (not crash), to never block legitimate traffic, and to not unduly affect network or host system performance.

The latency and throughput of a Network IPS (NIPS) or Attack Mitigation device must be on a par with other equipment in the network on which it is deployed, and in this respect, an in-line NIPS must strive to perform much more like a switch than a typical passive security device, especially when it is necessary to install more than one NIPS in the same data path.

Detection/Blocking Performance Under Load

This group of tests verifies that the IPS does not adversely impact legitimate traffic, even when new TCP connections are being created rapidly. We also verify that the sensor is capable of detecting and blocking exploits when subjected to increasing loads of background traffic up to the maximum bandwidth supported as claimed by the vendor. An IPS that misses attacks under load can be evaded. An IPS that adversely affects legitimate background traffic will not stay in-line for long.

A fixed number of exploits are launched with zero background traffic to ensure the sensor is capable of detecting our baseline attacks. Once that has been established, increasing levels of varying types of background traffic are generated **through** the IPS device in order to determine the point at which the sensor begins to miss attacks.

All tests are repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic (or up to the maximum rated throughput of the device in 25 per cent increments should this be less than 1Gbps). The test is conducted with UDP, HTTP, and mixed-protocol traffic and includes packet rates up to 453,000 packets per second and connection rates up to 20,000 connections per second.

Latency & User Response Times

In any network environment latency is important. Latency may impose an upper bound on throughput and it also has an impact on interactive applications, thus affecting user response time. As such, it is important to understand the impact of latency introduced by a NIPS and to determine the maximum acceptable delay, which will be different for each network.

There is a direct relationship between latency introduced by a networking device and the maximum throughput allowed by that device on a single TCP connection. There is a critical value for the *round trip time* (RTT) of a packet in each network, and if the latency is below this critical value, TCP throughput will be unaffected - instead, it is the line speed of the underlying network which becomes the bottleneck. Above this critical value, however, TCP throughput is negatively impacted. To be specific, the maximum throughput achievable for any given TCP connection in a zero loss network is expressed as:

throughput = window / RTT

where *window* is the maximum TCP window size (64 Kbytes by default) and RTT is the round trip time in the network.

This equation tells us that the throughput of a TCP connection is inversely proportional to network latency (note that this is TCP throughput for *one* connection - the aggregate bandwidth is not affected by latency). In other words, if you double latency, you halve throughput.

Consider adding a NIPS in an internal Gigabit network where the RTT is 200 microseconds. The critical value for RTT in a Gigabit network is 500 microseconds (below which it may no longer be possible to achieve 1Gbps of throughput), which means the NIPS can add a maximum of 300 microseconds to the RTT without affecting the network. In this particular case, therefore, for an internal, high speed deployment, the administrator may determine that his chosen IPS device needs to be capable of sub-300 microsecond latency under normal traffic loads.

Of course, the latency of an IPS device may vary significantly based on packet size, complexity of the protocol, presence of attack traffic, or simply the makeup of the normal traffic passing through it. For example, Gigabit segments, will rarely carry only a single TCP connection. Rather, a saturated Gigabit segment could be supporting hundreds, if not thousands of TCP connections, and this multiplexing eases the impact of latency on the overall throughput on the segment.

Although each of these connections carries only a fraction of the total throughput, a few connections tend to dominate. The maximum latency for a NIPS is then determined by the utilisation of the fastest connection. For example, in a Gigabit Ethernet segment carrying 10,000 TCP connections the fastest connection might have a throughput of 250Mbps. In this case, the critical value for round trip latency is as high as 2 milliseconds.

Assuming the latency without the NIPS is 300 microseconds, an administrator may therefore determine that his chosen NIPS device must be capable of 1700 microsecond round trip latency (850 microseconds in each direction).

Such critical value calculations are important when TCP connections achieve maximum throughput, which is true for large data transfers. For smaller data transfers, and non-TCP applications like NFS, latency has a more direct impact on user experience - response time is directly proportional to latency. That is, *doubling latency doubles response time.* In these situations, the latency of the network in which a NIPS is deployed determines the acceptable latency of the NIPS.

Consider deploying a hypothetical NIPS with 1 millisecond one-way latency in the following scenarios:

- In internal corporate LANs, the round trip latency could be in the 200-300 microsecond range. Deploying our hypothetical NIPS would increase the maximum round trip latency to 2.3 milliseconds, an increase of just over 700 per cent. The time to copy a large group of files, for example, would increase by a factor of seven.
- In inter-campus corporate networks connected over a MAN, the latency could be in the 500-1000 microsecond range (or less). Deploying our hypothetical NIPS would increase the maximum round trip latency to 3 milliseconds, a minimum increase of 300 per cent. The time to copy a large group of files, for example, would increase by at least factor of three.

- Internet facing connections experience round-trip latency from 10-100 milliseconds. Deploying our hypothetical NIPS would increase the round trip latency by 1-10 per cent, which would have only a minor impact on the user experience.

The latency of the NIPS must therefore be evaluated in the context of the network in which it is deployed. For example, to protect networks that are accessed over the public Internet, one-way NIPS latencies in the 1-2 millisecond range would be acceptable. Whereas for NIPS deployments on MAN/WAN links, NIPS latencies of well under 1 millisecond would be essential. And as we have already mentioned, for deployments on internal networks where latencies are a few hundred microseconds, NIPS latencies of less than 300 microseconds would be more appropriate.

Network administrators have laboured long and hard to reduce latency within the corporate network to an absolute minimum. Core network devices such as switches are frequently chosen as much on their performance - packet loss and latency under all load conditions - as any other feature. Given that Network IPS devices are operating in-line, it is not surprising that they will be evaluated in a similar way.

For this reason, part of The NSS Group methodology uses very similar testing techniques to those we would normally employ when testing switches (in order to determine *packet latency*), in **addition** to measuring *application latency*. This group of tests determine the effect the IPS sensor has on the traffic passing through it under various load conditions. High packet latency will lower TCP throughput. High application latency will create a negative user experience.

Bi-directional network latency of a range of differently-sized UDP packets is measured under three test conditions: with no load, with 500 Mbps of HTTP traffic (or half the rated load of the device if this is less than 1Gbps), and while the device is under a heavy SYN flood attack (up to 10 per cent of the rated throughput of the sensor).

Spirent Avalanche and Reflector devices are also used to generate HTTP sessions through the device in order to gauge how any increases in latency will impact the user experience in terms of failed connections and increased Web response times. This "*application latency*" is measured both with no background load and while the device is under attack.

Stability & Reliability

These tests verify the stability of the IPS device under various extreme conditions. Long-term stability is critical for an in-line IPS device, where failure can produce network outages.

In the first part of this test, we expose the external interface of the sensor to a constant stream of attacks over an extended period of time. The device is configured to block and alert, and thus this test provides an indication the effectiveness of both the blocking and alert handling mechanisms. A continuous stream of exploits mixed with some legitimate sessions is transmitted through the sensor at a maximum rate of 90 per cent of the claimed throughput of the device for eight hours with no additional background traffic.

The device is expected to remain operational and stable throughout this test, blocking 100 per cent of recognisable exploits, raising an alert for each, and passing 100 per cent of legitimate traffic. If any recognisable exploits are passed - caused by either the volume of traffic or the IPS device failing open for any reason - this will result in a FAIL. If any legitimate traffic is blocked - caused by either the volume of traffic or the IPS device failing closed for any reason - this will also result in a FAIL.

In the second part of the test we stress the protocol stack of the device under test by exposing it to malformed traffic from the ISIC test tool for eight hours. The device is expected to remain operational and capable of detecting and blocking exploits throughout the test to attain a PASS.

We scan the management interface for open ports and active services and report on known vulnerabilities. We also stress the protocol stack of the management interface of the NIPS by exposing it to malformed traffic from the ISIC test tool. 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 to attain a PASS. We also note whether the sensor detects the ISIC attacks even though targeted at the management port.

Security Effectiveness

Detection Accuracy & Breadth

This group of tests verifies that the NIPS will not block legitimate traffic (*Accuracy*) and is capable of detecting and blocking a wide range of common exploits (*Breadth*). Although *breadth* is extremely important, *accuracy* is critical because a NIPS that blocks legitimate traffic will not remain in-line for long.

We have a number of trace files of normal traffic with “suspicious” content, together with several “neutered” exploits that have been rendered completely ineffective. The IPS attains a “PASS” for each test case if it does **not** raise an alert and does **not** block the traffic. Whilst it is not possible to validate completely the entire signature set of any IPS, this test demonstrates how accurately the IPS detects and blocks a wide range of common exploits, port scans, and Denial of Service attempts.

This test is repeated twice: the first run with blocking disabled on the IPS in order to determine which attacks are detected and how accurately they are detected (*Attack Recognition Rating*); the second run with blocking enabled in order to determine which attacks are blocked successfully regardless of how they are detected or what alerts are raised (*Attack Blocking Rating*).

Following the initial test run, each vendor is provided with a list of CVE references of the attacks missed and is allowed 48 hours to produce an updated signature set. This updated signature set must be released to the general public as a standard signature/product update before the report is published - this ensures that vendors do not attempt to code signatures just for this test.

Naturally, Rate-Based IPS devices will not respond to the same attack traffic as Content-Based devices, and so for those the Detection Accuracy tests involve detecting and mitigating a wide range of rate-based attacks such as port scans, SYN floods, connection floods, and so on.

We note which of these are mitigated completely, which are mitigated partially, and which require the use of built-in firewall capabilities.

Resistance To Evasion Techniques

These tests verify that the IPS is capable of detecting and blocking basic exploits when subjected to varying common evasion techniques. An IPS that cannot detect attacks subjected to these "script kiddie" evasion techniques is easily bypassed.

The tests consist of four parts (only the third is applicable to Rate-Based devices):

- ***Baselines -*** *This establishes that the IPS is capable of detecting and blocking a number of common basic attacks (our baseline suite) in their normal state, with no evasion techniques applied.*
- ***Packet Fragmentation and Stream Segmentation -*** *The baseline HTTP attacks are repeated, running them through fragroute using 19 evasion techniques.*
- ***URL Obfuscation -*** *The baseline HTTP attacks are repeated, this time applying 9 URL obfuscation techniques made popular by the Whisker Web server vulnerability scanner.*
- ***Miscellaneous Evasion Techniques -*** *Certain baseline attacks are repeated, and are subjected to 7 protocol- or exploit-specific evasion techniques, including altering default ports, inserting spaces in FTP command lines, inserting non-text Telnet opcodes in FTP data streams, and RPC record fragging.*

For each of the evasion techniques, we note if (i) the attempted attack is blocked successfully (the primary aim of any IPS device), (ii) the attempted attack is detected and an alert raised in **any** form, and (iii) if the exploit is successfully "decoded" to provide an accurate alert relating to the original exploit, rather than alerting purely on anomalous traffic detected as a result of the evasion technique itself.

Stateful Operation

If the IPS is tracking TCP session state, then it has the potential to introduce denial of service when the session table becomes full (too many connections) or if it can't keep up with the creation of new sessions (too many connections per second). As with latency and bandwidth, the number of connections supported by the IPS and its connection per second rate should be matched to the network.

For example, a fully saturated Gigabit Ethernet link can handle 22,000 5KByte transfers per second. Assuming each connection lasts 20 seconds, the IPS should be able to handle 448,000 simultaneous connections. These numbers scale proportionately for slower networks. Any IPS that doesn't offer these capabilities will impact performance of Web or e-commerce servers.

The aim of this section is to be able to determine whether the IPS is capable of monitoring stateful sessions established through the device at various traffic loads without either losing state or incorrectly inferring state.

An IPS that does not maintain TCP session state can flood the management console with false-positive alerts. Although this should not directly impact the IPS blocking function, it can make it very hard to perform forensic analysis of the attacks. In addition, if the default condition of the sensor is to block all traffic for which it does not believe there is a current connection in place, then an inability to maintain state under extreme conditions could result in the sensor blocking legitimate traffic by mistake.

In the first part of this test, we transmit a number of packets taken from capture files of valid exploits, but without first establishing a valid session with the target server. In order to receive a "PASS" in this test, no alerts should be raised for any of the actual exploits. However, each packet should be blocked if possible since it represents a "broken" or "incomplete" session.

In part two, we test whether the sensor is capable of preserving state across increasing numbers of open connections, as well as continuing to detect and block new exploits while not blocking legitimate traffic when the state tables are filled. Various numbers of TCP sessions from 10,000 to 1,000,000 (one million) are tested.

This test is run in both the out-of-box configuration and then repeated after applying any tuning recommended by the vendor (if applicable) to increase the size of the state tables.

Usability

After quantitatively evaluating the network performance and security effectiveness of the IPS, we qualitatively evaluate the features and usability of the product.

This evaluation provides the reader with valuable insight into product features, how easy it is to install the IPS and perform common, day-to-day operations with the management console. Areas evaluated include *installation, configuration, policy editing, alert handling,* and *reporting and analysis*.

TOP LAYER IPS 5500 V3.3

Executive Summary

The Attack Mitigator IPS 5500 is Top Layer Network's new family of Network Intrusion Prevention Systems. The IPS 5500 automatically mitigates attacks from both external and internal network sources while allowing legitimate traffic to pass. The current platform – which offers both rate-based **and** content-based IPS capabilities – is based on an extension of Top Layer's previous Attack Mitigator appliances. The IPS 5500 product line consists of multiple models that provide full coverage from 200Mbps up to 2Gbps with transaction rates up to 50,000 sessions per second.

Top Layer's second-generation ASIC technology and mitigation algorithms integrate stateful analysis techniques with its new *TopInspect* deep packet inspection technology and DoS (Denial of Service) attack protection to provide comprehensive protection from Internet-based and internal threats.

The IPS 5500 offers High Availability (HA) configurations, redundant capabilities, hot-swappable power supplies and hot-swappable fan-tray, secure custom operating system, and flexible port-bypass capabilities to provide a high degree of reliability.

The IPS 5500 is the first ever device to be tested against both our *Content-based IPS* **and** *Rate-based IPS (Attack Mitigator)* methodologies, and it performed extremely well in both. Indeed, at the time of writing the IPS 5500 is one of only a few devices capable of completing both methodologies, thus allowing a single device to be deployed to protect fully against both types of attack.

Overall, the performance of the IPS 5500 is very impressive, combining almost flawless detection rates at Gigabit wire speed with some of the lowest latency figures we have seen under any traffic conditions. We also found the IPS 5500 to be very stable, surviving our extended reliability tests without missing a beat, and without blocking any legitimate traffic or succumbing to common evasion techniques.

Attack recognition capabilities have improved significantly in the latest release, and its rate-based attack mitigation and bandwidth management features remain as impressive as ever.

The management interface and reporting capabilities are relatively limited, though the new *Central Management System* (CMS) provides more extensive management features, albeit at additional cost.

Architecture

Top Layer's architecture offers network-level and application-level protection along with the flexibility to integrate application-specific protection mechanisms. Second generation *TopFire* ASIC technology provides high-performance, providing protection at Gigabit wire speeds.

Stateful inspection firewall technology provides the network level protection, identifying undesired access, illegal packets, illegal headers, and various network attacks.

Top Layer's denial of service protection algorithms protect against flood-based attacks, such as ICMP, UDP, and TCP SYN Floods. New *TopInspect* deep packet inspection technology provides full protocol decoding and application-level protection against exploits of critical vulnerabilities, including worms and application-level attacks.

The Top Layer offering consists of:

- *IPS 5500 appliance*
- *Management Application*
- *Central Management System (CMS)*

The IPS 5500 has three model variants:

	IPS 5500-100	IPS5500-500	IPS5500-1000
Rated Throughput	200Mbps (one Full Duplex Link)	1000Mbps (One 50% loaded Full Duplex Gig Link	2Gbps (One Full Duplex link)
Raw Throughput (Mbps)	400Mbps	2.4Gbps	4.4Gbps
Session Table size	512K	512K	1M
DoS SYN Flood Filter	300K SYNs per second	1M SYNs per second	1.5M SYNs per second
DDoS SYN Flood Filter	300K SYNs per second	500K SYNs per second	500K SYNs per second
IP Address Table Size	256K, 1M	1M, 2M	2M, 4M
Mission Ports	4 x 10/100	4 x GBIC, 4 x 10/100	4 x GBIC, 4 x 10/100
Power Supplies Standard	One (2nd Optional)	Two	Two
Expandable Architecture.	Yes, Future Application security processor	Yes, Future Application security processor	Yes, Future Application security processor
Field Upgradeable	No	Yes, to -1000	N/A

In general, the features of the models in the IPS 5500 range are identical - the differences are primarily limited to the physical ports installed and the performance capabilities of the product.

Management Application

The IPS 5500 management processor and model is designed to operate in either a two-tier management architecture (using the built-in management system) or a three-tier architecture (using a centralised management platform designed to manage multiple IPS 5500's).

The built in Java Management Application provides two-tier device management, allowing the administrator to access, manage and monitor a single IPS 5500 appliance at a time. This application is specific to the software version running on the IPS-5500, and is updated as part of the firmware/microcode update process. All configuration and real-time monitoring takes place via this interface.

This application can be launched from a browser, or started from Java Web-start, and provides a self-updating, browser independent and operating system independent element management solution. The administrator has a choice of always accessing via the Web browser (in which case the application will be downloaded from the IPS 5500 appliance each time it is used) or integrating into the local desktop, in which case the application is downloaded once and installed locally.

The IPS 5500 uses Top Layer's *Remote Management Protocol* (RMP), which is implemented in one of the seven processors in the IPS 5500, and provided over two long-lived connections, one for requests and one for responses. Both connections run over TCP/IP using port 80 (HTTP) or port 443 (HTTPS) depending on the configuration. The IPS 5500 contains a secure embedded Web server to provide the device management Java application and RMP service.

The IPS 5500 can be managed over any IP network via the dedicated management port, and the IPS 5500 will not respond to management requests on its mission ports (even ARP requests).

Persistent configuration is kept on an internal 256MB Compact Flash, which is externally accessible, and can easily be removed, backed up, or replaced. Configuration files can also be backed up, or restored as required via the management interface. IP address and other unit-specific information is not kept on the Compact Flash, but is rather kept in an internal FLASH memory location. This allows the administrator to keep spare Compact Flash modules that can be used on any IPS 5500 with no modification or customisation required.

Central Management System (CMS)

Top Layer's central management platform, *SecureCommand+*, extends the reporting and analysis capabilities of the IPS 5500 device manager to multiple IPS 5500 units organised in High Availability (HA) pairs, groups, or an entire enterprise. SecureCommand+ contains security event management capabilities supplied by Top Layer's strategic partner, *OpenService*, to provide an enhanced set of historical trending and query-based reporting and analysis features.

The CMS uses the same RMP protocol as the Application Manager, and offers the following additional capabilities and functionality:

- *Manage multiple IPS 5500 units, as HA pairs or configurable groups.*
- *Simplify policy, rules, signature, and version management for multiple units.*
- *Comprehensive management of configuration files, and change reports.*
- *Full audit capabilities.*
- *Device availability and health check monitoring, plus automatic RMP connection re-establishment if connection is lost.*
- *Ability to generate events as required for device availability and health check / resource monitoring.*
- *Full implementation of user authentication and security model of IPS 5500. In the future this will be extended beyond the current IPS 5500 functionality.*
- *Provide HA supervisory role for multiple 5500's, necessary if more than two devices are configured in a high availability configuration.*
- *API to accept configuration changes from third party "umbrella management systems".*
- *Select multiple IPS 5500 units, and view aggregated real time statistics.*
- *Event aggregation of events from IPS 5500 units, and optional relay to third party event correlation systems.*
- *Fault management of managed devices.*

IPS 5500 Appliance

The IPS5500 is a dedicated 2U appliance, containing a range of ports that make it easily and transparently deployable in most single link and High Availability (HA) dual-link topologies for both Fast Ethernet and Gigabit networks.

Figure 1 describes the flexible assignments of physical ports that are possible.

The front panel sports four Gigabit GBIC ports (supporting copper or fibre connections, all of which can be used as mission ports), two redundant fibre HA link ports, and eight copper 10/100Mbps ports (up to four of which can be used as mission ports, and one of which is dedicated to management). Also on the front panel is an LED status display, two USB ports (not used in the current release) and a serial console port for command line management.

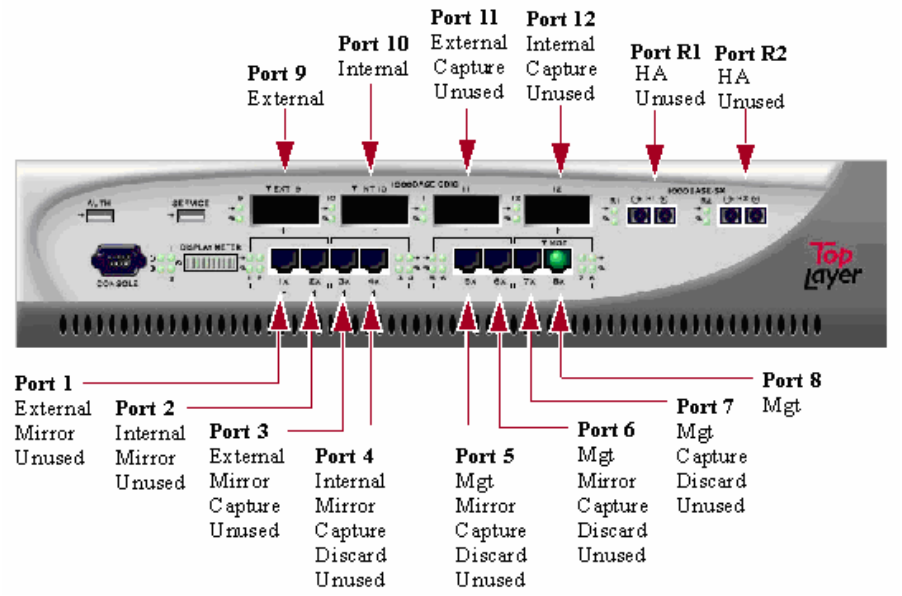

Figure 1 - Top Layer: The IPS 5500 ports

At the back of the units are the twin power supplies and fans, and access to the Compact Flash bay.

The IPS5500 architecture is a combination of stateful analysis firewall filtering, Deep Packet Inspection (DPI), and rate-based IPS (attack mitigation) functions. The components that go to make up the IPS 5500 are primarily ASICs and FPGAs, to provide the maximum performance with a high degree of flexibility.

The ASICs are interconnected with high speed buses, creating a network device with extremely low latency, high throughput, and no data-path bottlenecks which you might expect to find on a PCI bus-based platform. Five of the ASICs have a RISC core imbedded in them, allowing their function to be changed by firmware update without compromising performance. The FPGAs allow complex functions to be programmed into the hardware architecture, and a daughter card allows the IPS 5500 to be extended in a variety of ways to meet the differing needs of the market (i.e. Anti Virus processor, etc.).

The IPS 5500 runs a proprietary real time operating system (RTOS) on the RISC CPU's in the ASICs, and on the System Controller processor. All of the firmware / microcode for the Top Layer ASICs and FPGAs in the IPS 5500 is field upgradeable.

This is achieved by uploading a single image to the IPS 5500 via the management interface, or by installing a different image directly to the compact flash card. Multiple images can be stored on the IPS 5500, and an upgrade can be easily reversed if required. Firmware / microcode upgrades require a device reboot, which can be performed independently of the upgrade, or immediately afterwards.

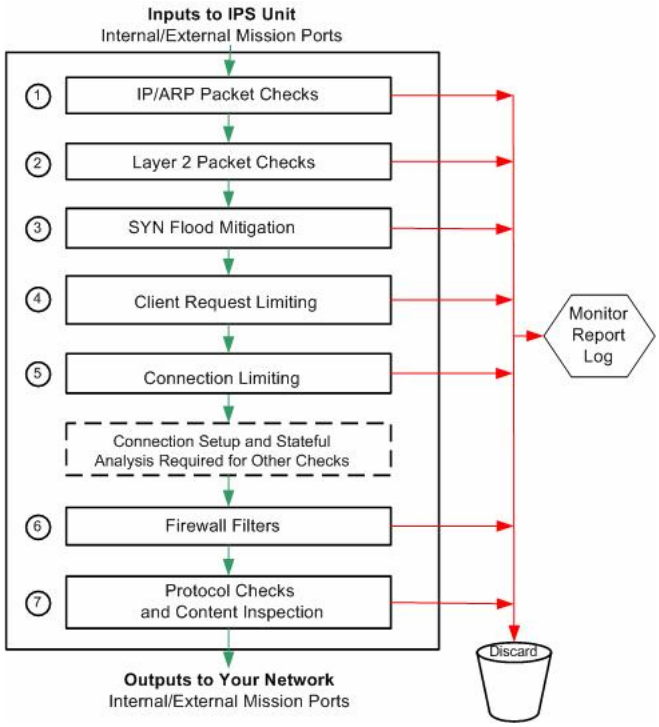

Figure 2 - Top Layer: IPS 5500 architecture

High Availability

The IPS 5500 is designed with High Availability (HA) in mind. High-MTBF hardware design with no rotating media, redundant hot-swappable power supplies, and a hot-swappable N+1 fan tray ensure non-stop operation. Port bypass on all internal and external network ports ensures network availability even in the event of an internal IPS failure.

Two or more IPS 5500 devices can be configured into a *ProtectionCluster* High Availability (HA) network configuration, which allows for full redundant operation with no single point of failure, scalable performance and capacity, and automatic reconciliation of traffic flows split by asymmetric routing paths. Multiple IPS 5500 units are connected using an additional pair of dedicated Gigabit Ethernet links. These provide flow rebalancing in high availability configurations, as well as IPS-to-IPS communications.

Currently, the IPS 5500 supports most network topologies for High Availability, including *Active-Active* and *Active-Standby* configurations. Each topology brings its own advantages and disadvantages:

- *In Active-Active HA topologies, all IPS units in the ProtectionCluster are connected to an active network segment or path, and are receiving and transmitting network data packets. In a two-unit ProtectionCluster, this setup doubles the maximum number of simultaneously active sessions that can be tracked by the IPS5500.*

Each IPS unit maintains state information for the sessions for which it is responsible. In the event that either of the two IPS units becomes inactive, state information is preserved only for the remaining active IPS unit.

- *Active-Standby topologies refer to the configuration in which both IPS units receive network data packets, but the filtered network data packets are forwarded to only one active firewall or network component. This asymmetrical topology allows only one branch of the network to be considered "active" at any given time.*

Performance

Content-Based

The aim of this section is to verify that the sensor is capable of detecting and blocking exploits when subjected to increasing loads of background traffic up to the maximum bandwidth supported as claimed by the vendor.

For each type of background traffic, we also determine the maximum load the IPS can sustain before it begins to drop packets/miss alerts. It is worth noting that devices which demonstrate 100 per cent blocking but less than 100 per cent detection in these tests will be prone to blocking **legitimate** traffic under similar loads.

The IPS 5500 is rated by Top Layer for a single Gigabit link (2Gbps aggregate) and was tested up to 1Gbps here. It turned in an outstanding performance in all our tests, achieving 100 per cent detection rates across the board, and clearly with some headroom to spare. We would be more than happy to rate this device as 1Gbps under **all** network loads.

Basic latency figures were also outstanding - almost switch like - across the board under normal traffic loads. They ranged from 20µs with 250Mbps of 256 byte packets, to 41µs with 1Gbps of 1000 byte packets.

Behaviour throughout the tests with no background traffic was extremely consistent and predictable, hardly increasing at all as additional network load was applied from 250Mbps to 1Gbps. The IPS 5500 was one of the few devices we have tested in our labs which has achieved zero packet loss and low latency at **all** packet sizes (including 64 bytes) up to 1Gbps. The latency with 64 byte packets at 1Gbps was just 17µs (which also includes the basic latency of the test infrastructure).

Placing the device under a half load of 500Mbps of HTTP traffic, we noted some slight increases in latency, though the figures still never climbed above 62µs. HTTP response times were also excellent.

100Mbps of SYN flood traffic had a negligible effect on the IPS 5500, increasing the base latencies at all packet sizes by around a microsecond only. The SYN flood was mitigated completely once it had been detected (which did not take long). Note that this is not the case with all DOS/DDOS attacks, however. With many other types of attack, the IPS 5500 does "mitigate" (i.e. reduce) the effects of the attack rather than block it completely.

Overall, latency figures were considered to be outstanding for a device of this type under all load conditions and packet sizes.

Clearly this device can be placed anywhere on the corporate network - from the perimeter to a heavily-loaded high-speed backbone - without impacting overall network performance in any way.

The IPS 5500 performed consistently and completely reliably throughout our tests, continuing to block attack traffic in a consistent manner whilst passing 100 per cent of the legitimate traffic, even when under extended attack. Exposing the sensor interface to ISIC-generated traffic had no adverse effect, and the device continued to detect and block all other exploits throughout and following the ISIC attack.

Rate-Based (Attack Mitigation)

In the rate-based (attack mitigator) attacks the IPS 5500 performed equally well – indeed, the Top Layer device is currently one of only a few devices on the market capable of completing both our content- and rate-based methodologies. Performance at all levels of our load tests was impeccable, with 100 per cent of all attacks being detected and mitigated under all load conditions, and no interruptions to legitimate sessions. Latency too was very low across all tests, even when under heavy DOS attack.

DDOS attacks (multiple source IPs) proved trickier to handle, with CPU becoming a bottleneck much earlier (between 200Mbps and 400Mbps), causing packet loss. the IPS5500 device is rated by Top Layer for DDoS protection at up to 500,000pps, (approximately 333Mbps with 64 byte packets), and the ProtectionCluster feature (not tested) can be used to scale this solution to higher rates.

Overall latency performance under all normal and DOS conditions was considered to be excellent, and HTTP response times remained remarkably consistent throughout all our DOS attacks.

Please refer to the *Testing Methodology* section for full details of the methodology used and performance results.

Security Effectiveness

Content-Based

We installed one sensor with the latest updates, and used the "*Very Strict*" policy, which has almost every signature enabled (barring approximately eight which are considered audit only, and too noisy for most environments – these are disabled by default in this policy). Default settings were used for SYN Floods.

Top Layer has made significant additions to its protocol decode and validation modules, and significant enhancements to its signature set for this release. Signature recognition (with blocking disabled) was good out of the box (85 per cent), and was increased to a creditable 94 per cent after the application of a signature pack update which was provided to us within 48 hours. Blocking performance was identical throughout the tests.

We noted a minimum of "noise", with very few test cases raising multiple alerts for a single exploit. Performance in our "false negative" tests was reasonable out of the box, but did not improve following the signature update.

A major concern in deploying an IPS is the blocking of legitimate traffic. Providing the signatures which are disabled by default in the standard policies are left that way, the IPS 5500 resistance to false positives is good. However, it should be noted that when the various attack mitigation features are employed, careful tuning of the mitigation parameters is required in order that legitimate traffic is not blocked accidentally. There is no automatic "learning" capability, meaning the responsibility for determining the optimum mitigation thresholds lies squarely with the administrator. The "always bypass" mode allows the administrator to test many (although not all) settings before putting the device fully in line.

The IPS 5500 appliance now arrives with four default policies: *Detect Only*, *Recommended*, *Strict* and *Very Strict*, each imposing different combinations of detection and blocking. The *Recommended* policy will suit most organisations, since it disables all signatures that are not considered "*likely exploits*" (i.e. audits/information signatures) and all those where the confidence setting is "*low*" (i.e. possibly susceptible to false positives).

Resistance to known evasion techniques was excellent, with the IPS 5500 achieving a clean sweep across the board in most of our evasion tests. *Fragroute* and *Whisker* both failed to trick the device into ignoring valid attacks. Not only were the fragmented and obfuscated attacks blocked successfully, but many of them were decoded accurately as well.

Out of the box, the IPS 5500 is designed to handle 1,000,000 (one million) open connections without tuning. It was thus able to handle our 1 million open connection test with ease.

Default operation of the device is to age out old connections in order to accept new ones when the state tables are full or resources are low. This behaviour is configurable, a feature we would like to see in all IPS products since there is no real right or wrong way to handle this situation.

Stateless "exploits" are not alerted upon (this is correct behaviour in order to be resistant to *Stick* and *Snot* tools) and mid-flows are blocked by default. It is, however, possible to configure the device to allow mid-flows, and there is a configurable "grace period" where they are not enforced following a power-cycle to prevent blocking of legitimate traffic should the device come on-line in mid session.

Rate-Based (Attack Mitigation)

Configuring for the rate-based (attack mitigator) attacks was tricky, requiring much more care and consideration in order to avoid self-imposed DOS conditions. However, once configured, the device detected and mitigated most of our attacks successfully.

Performance in the high volume detection/mitigation test was almost impeccable across the board, with perfect detection and mitigation at all load levels. This is the real strong suit of the IPS 5500 when it comes to rate-based attacks. Some problems were noted in passing legitimate traffic at the highest load levels of some of the DDOS attacks due to high CPU utilisation and subsequent packet loss. However, these load levels can be considered excessive, and the device performed almost impeccably up to the 600Mbps level of attack traffic. In addition, the ProtectionCluster HA configuration can be used to scale performance beyond these levels if required.

We feel that the IPS 5500 is lacking in its ability to detect and mitigate certain scan and probe attempts, relying on its firewall filtering to catch some, and mitigating others only partially. Scan and probe attempts aside, however, the DOS and DDOS mitigation proved to be excellent, as did the resistance to common evasion techniques.

Please refer to the *Testing Methodology* section for full details of the methodology used and performance results.

Usability

This part of the test procedure consists of a subjective evaluation of the features and capabilities of the product, and covers *installation, configuration, policy editing, alert handling,* and *reporting and analysis*.

Installation

With its roots in the attack mitigator space, the IPS 5500 would be installed between the firewall and external router in most networks. In this position it can provide protection for both the firewall and internal network against various DoS, DDoS or other types of attack, as well as providing content-based IPS protection. Naturally, there is nothing to prevent its installation behind the firewall, or directly in front of a server farm, as with a normal content-based IPS solution.

The IPS 5500 has two distinct packet forwarding modes, designed to simplify deployments in complex networks, without the need for redesign of those networks:

- ***Bridging mode -*** *In bridging mode the IPS 5500 behaves as a Layer 2 switch when it comes to forwarding frames, with all that that implies (MAC address learning, etc.) The management ports of the IPS 5500 form a separate bridge, and packets are never forwarded between the management domain and the mission domain by the IPS 5500.*
- ***Port-pair forwarding mode -*** *In port pair forwarding mode, frames are forwarded out of the corresponding paired port. This effectively means that the IPS 5500 can be deployed in multiple links in a complex network without creating bridging loops. It is also currently a requirement for HA operation.*

The IPS 5500 also supports installation within asymmetric networks - networks where frames from an individual TCP connection do not always follow a single network path. This causes problems for stateful devices, as they typically need to ‘see’ all the frames for a given connection to work correctly.

Individual AM-IPS 5500’s can take care of network asymmetry, provided a single device can ‘see’ all of the connections frames on one of it’s mission ports.

For multi-device HA solutions, asymmetry is typically split across multiple devices. Top Layer’s HA solution takes care of this automatically as part of the basic HA configuration. Currently this is limited to two IPS 5500’s, but up to eight will be supported in a future release.

Given the simple two-tier management structure and the appliance approach, the IPS 5500 is extremely simple to install. The first step is to connect to the serial console port in order to assign the appropriate IP address, net mask, time zone and name/ID of the appliance. Once this has been done, further configuration is performed via the GUI over the Ethernet management port.

Figure 3 - Top Layer: The IPS 5500 appliance

That IPS 5500 *Management Application* is a self-updating Java Web Start application, and requires that the Java Runtime Environment (JRE) is installed. A browser is then used to access the built-in Web server of the appliance, which displays a Welcome screen from where the administrator can access the *Management Application* or the *Internal State Browser*. The latter is a hierarchical tree display of every configurable parameter within the IPS 5500 and is used for troubleshooting purposes rather than management and configuration.

Selecting the IPS 5500 *Management Application* for the first time from the Welcome screen offers the administrator the option to integrate the application into the desktop. If he chooses not to do this, the application remains on the IPS 5500 and will need to be downloaded each time it is accessed. If he chooses integration, however, the GUI is automatically downloaded and installed on the local PC as an independent application. From that point on, the software is cached locally and can be accessed without going through the 5500's Web server and Welcome screen. This is a nice way of facilitating initial deployment of the Management Application.

Installation is completed by running the *Getting Started Wizard*. This is where the administrator can configure port roles, port forwarding, port bypass, and other settings. Based on the information provided, the Wizard automatically determines the optimal port configuration for the IPS Unit.

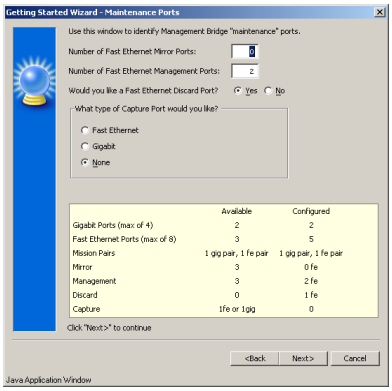

Figure 4 - Top Layer: Getting Started Wizard

When first running the *Getting Started Wizard*, it is normal to set the *Port Bypass* feature to *Always Bypass*.

With this setting, the IPS Unit evaluates and reports on possible mitigation issues it detects, but permits all traffic to pass through. This setting is used for evaluation when first configuring traffic security policies, allowing the administrator to examine the feedback and modify settings to best suit his own network situation. Once the configuration has been finalised, a single mouse click switches to one of the active mitigation settings such as *Never Bypass*, which forces the 5500 to enforce the applied security policy. A third setting, "*Bypass During System Reset*" ensures that the device is active when running, but passes all traffic when powered down or during a power cycle.

Documentation is very comprehensive, and appears to be accurate and well written. The main manuals provide far more than basic instructions on using the GUI – instead they offer plenty of background information covering the functions of the various options and parameters in depth. The following documents are provided both as hard copy PDF versions:

- ***IPS 5500 Release Notes -*** *Information on known problems, bug fixes, technical tips, installation, and Top Layer Customer Support.*
- ***IPS 5000-Series Hardware Installation -*** *Detailed information on IPS 5000 series product hardware features, installation, and basic parameter settings.*
- ***IPS 5500 Planning and Deployment -*** *Detailed instructions and cross-references to information needed to plan and implement a network security strategy using the IPS 5500 or a set of IPS 5500 units.*
- ***IPS 5500 Network Configuration and Management -*** *Configuring and managing the integration of the IPS 5500 product into a production network. Includes network and port role settings, establishing client and server groups, and other configuration features (except security).*
- ***IPS 5500 Security Configuration and Management -*** *Configuring and managing the IPS 5500 security features for a production network.*
- ***IPS 5500 On-line Help -*** *Detailed descriptions for configuration parameters as well as set-up details and notes for IPS 5500 features.*

Configuration

The IPS Unit's Management Application is a Java Web Start application that runs as a stand-alone application, and is the principal form of management configuration and monitoring access. The application's interface consists of two main access points:

- ***Navigation Tree –*** *A hierarchical tree menu provides access to IPS 5500 port, network, and basic configuration options, plus basic device information windows, reports, and graphs.*
- ***Security Configuration Access -*** *Provides access to the configuration of security objects and security policies, as well as two fixed graphs:*
    - *Connection Setup Rates*
    - *Dropped Packets*

The *Navigation Tree* contains the following top-level menu options:

- ***Getting Started Wizard -*** *Configure port roles, port forwarding, port bypass, and other port settings. Based on the information supplied, the Wizard automatically determines the optimal port configuration for the IPS.*

- ***Front Panel View -*** *Dynamic representation of the IPS ports showing their roles, status, and critical settings. Port settings can also be changed from this window.*
- ***Port Bypass -*** *Switch between mitigation and bypass modes during initial configuration.*
- ***High Availability -*** *Access setup parameters for identifying a second, redundant IPS Unit (shares load with the first IPS Unit when both are operational).*

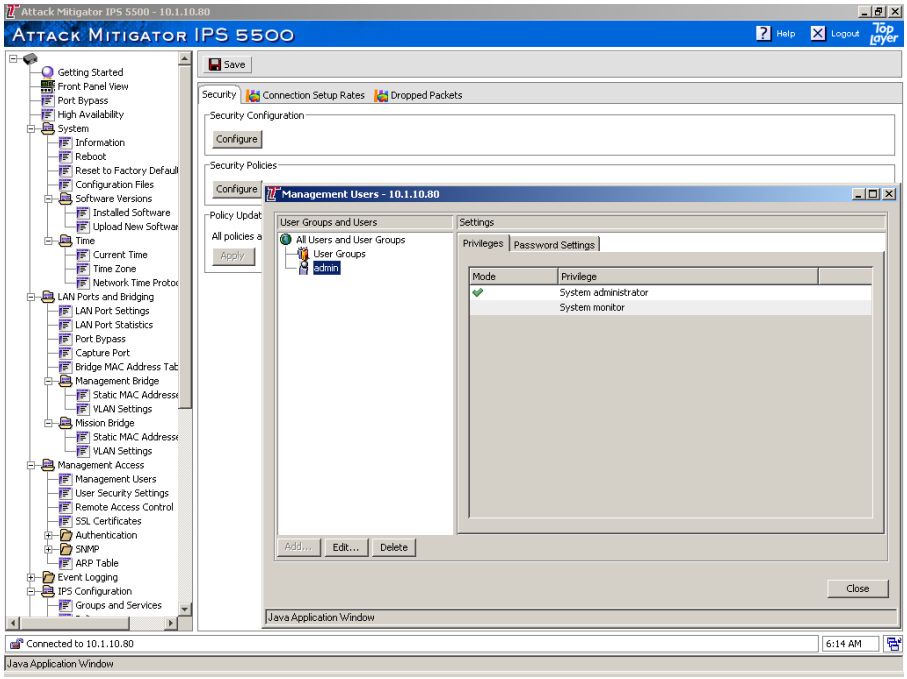

Figure 5 - Top Layer: Management Application Navigation Tree

- ***System -*** *View system information, reboot the IPS Unit, reset to factory configuration, save and manage custom configurations, upgrade software, update the system time, and set up NTP server time updates.*
- ***LAN Ports and Bridging -*** *View and configure port roles, port settings, VLAN settings, and static MAC addresses.*
- ***Management Access -*** *Allow or deny specific management services such as Telnet or SNMP, upload custom SSL certificates, set up SNMP community strings and trap server access, and create and edit users and user groups.*
- ***Event Logging -*** *Configure the event logging capabilities, such as message priorities, message groups, event thresholds, and Syslog server locations. These refer to system events rather than security events, creating entries when user-configurable thresholds are exceeded such as CPU activity, excessive network activity, hardware failure, and so on.*
- ***IPS Configuration -*** *Download the IPS Unit's reports to the management station or other network management device; view graphs and tables showing IPS Unit security operations; query a specific IP Address to monitor its associated security events; and monitor the statistics of various security events.*
- ***Reports and Statistics -*** *Download the reports to the management station or other network management device, and view graphs and tables showing IPS 5500 security operations.*

Also, query a specific IP Address to monitor its associated security events, and monitor the statistics of various security events.

- ***Maintenance -*** *Download the configuration and diagnostics file to the management station.*
- ***Help –*** *On-line help for configuring and managing the IPS 5500.*

The *Security Configuration Access* window does not actually contain any of the information that is selected via the three buttons within it - *Configure Security Configuration, Configure Security Policies* and *Apply Policy Update* – or the Navigation Tree. Instead, choosing any of the buttons or any of the entries in the Navigation Tree spawns new windows to provide access to the configuration data or graphs selected.

This can lead to a cluttered look to the screen on occasion, but it is the only way that it is possible to have multiple graphs and status monitoring windows active at the same time. The *Window Manager* provides some element of control over multiple windows, allowing the administrator to list all open windows and select any one of them from the list. One possible improvement here would be to allow the administrator to spawn the windows he uses most often, lay them out as he wishes, and then save that window layout for instant recall in the future.

One Admin account is created during install, and any number of additional users can be created via the GUI. There are two privilege levels available, one for administration and one for monitoring only, and there is no granularity of administrative functions other than these two levels. Any number of user groups can be created, and these can also be assigned *admin* or *monitor* privilege. The next release will increase this to five levels of authorisation and provide segregation of function management within the IPS 5500 for such things as signature updates, policy management, and so on. The optional *Centralised Management System* (CMS) will also make it possible for multiple levels of specific access to be granted

Passwords are stored in an encrypted hash in a file on the compact flash card. A RADIUS server can be used for authentication if required, and additional methods of access can be configured for management and configuration tasks, including Telnet, SSH, HTTPS and SNMP.

Although Top Layer does offer updates to firmware and content-based IPS signatures, at present there is no slick, automated means to achieve this. Patches to software and Protocol Validation Modules are available via firmware upgrades, whilst new signatures are provided as a complete “forklift upgrade” to the existing signature files – unfortunately, any custom changes effected by the administrator will be overwritten by such an upgrade. Both of these are provided for manual download from the Top Layer Web site and must be applied manually. Top Layer has acknowledged that this area needs considerable improvement and plans to introduce updates to its TopResponse subscription service in forthcoming releases.

Policy Management

The IPS 5500 stores its configuration in a set of configuration files on the appliance itself. Each file contains specific configuration values for a single subsystem within the appliance. Taken as a set, the files provide a snapshot of the entire device configuration.

Configuration files can be managed via the *System* option of the *Navigation Tree*, allowing the administrator to save and restore the configuration for an IPS 5500, several subsystems, or a single subsystem, as desired.

The IPS 5500 implements all configuration changes as soon as they are made (except for policy updates, which must be applied first). However, it does not save those changes to its non-volatile memory until they are explicitly saved. Unfortunately, there is no “discard changes” option – the GUI just keeps prompting for changes to be saved until the administrator complies.

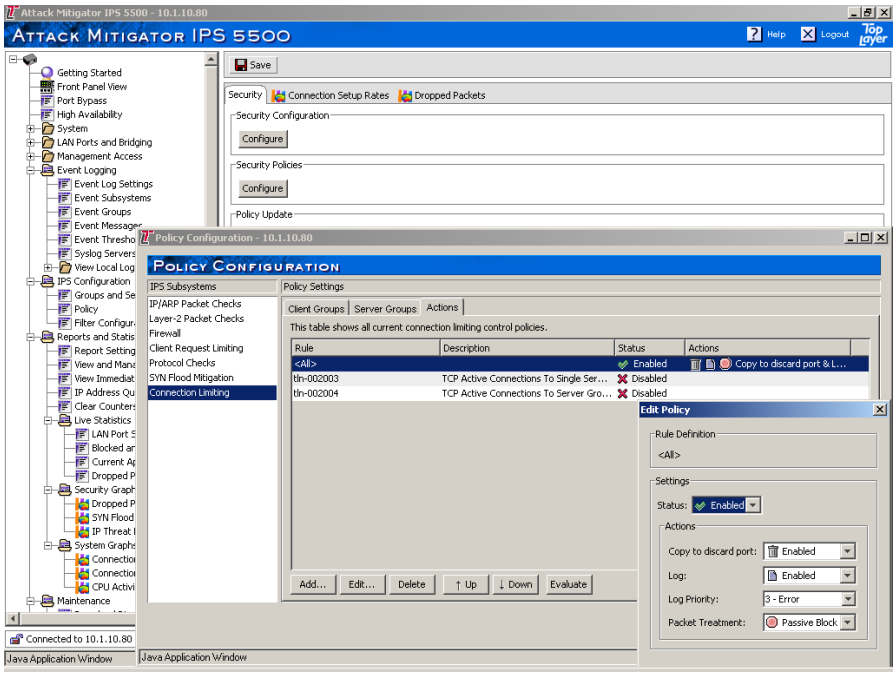

Figure 6 - Top Layer: Policy configuration

Whenever a new policy is applied to the device it is given a new version number and marked as the *active* policy - the previous policy version is retained and marked as *inactive*. Each policy is divided into several configuration files covering *LAN port, management, security, policy* and *environment* settings, and any one of these can be “rolled back” to a previous version by simply *Activating* the required version. Any or all of the individual configuration files can be downloaded to the host PC, or uploaded to the appliance, as required.

This level of granularity within the configuration seems like overkill when managing a single device, but really comes into its own when using the *Central Management System* (CMS) where it is possible to apply the same policy and security settings across multiple devices without affecting, for example, the LAN port configuration on each device.

The first step in defining a Policy is to define the various *Security Objects* via the *Security Configuration* menu options:

- ***Named IP Address Ranges -*** *A default or user-defined range of IP addresses (can be a single address) whose traffic the IPS Unit should automatically block, or otherwise handle in some special manner. This feature provides the ability to filter IP packets by source and/or destination IP address or IP address range.*

- ***Client & Server Groups -*** *Groups of clients/servers made up of one or more Named IP Address Ranges*
- ***Services -*** *Some security features (such as, firewall checks and service rate limiting) are based on the ability of the IPS to identify and handle specific network services – or Applications. A service (in this context) is any protocol that can be described to the IPS 5500 in terms of port number and protocol, for example: HTTP, DNS, or FTP. The IPS comes with a large set of predefined services (including definitions for many Trojans), and custom services can be added if required.*

  In order to detect malicious traffic, each service can be processed by a protocol decode and validation module (where one exists) or by the ***Payload String Matching*** *module, which provides the ability to create signatures for protocols that the 5500 does not otherwise parse. These signatures can be case insensitive or sensitive ASCII printable characters strings or hexadecimal byte sequences. Rate limits can also be applied per service - Kilobytes per second for IP applications, or packets per second for non-IP applications.*

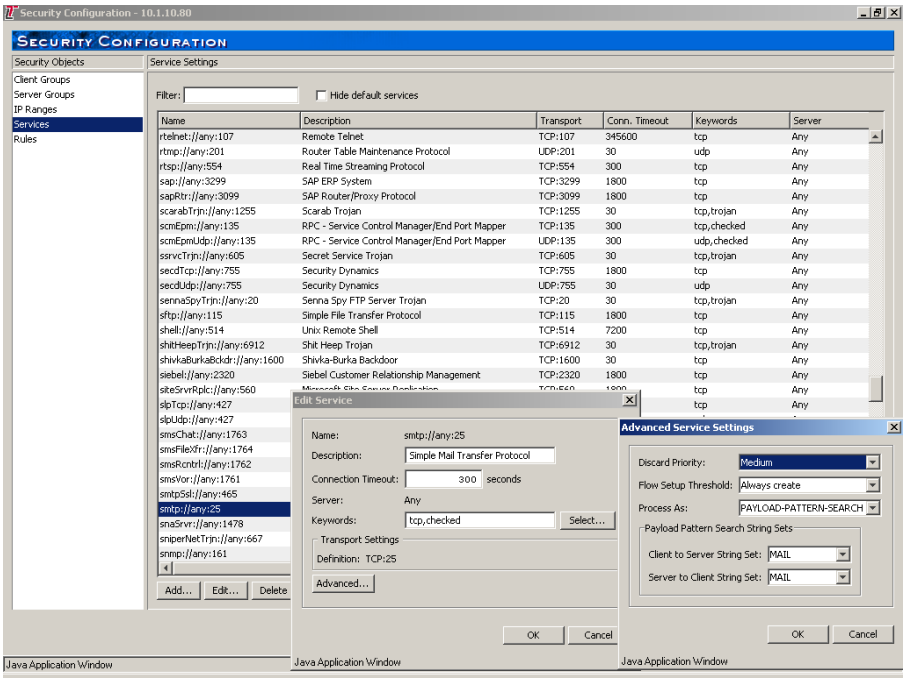

Figure 7 - Top Layer: Defining services

- ***Rules -*** *The IPS 5500 includes a library of rules that define instances of how it will examine and process network traffic. When creating policies, the administrator defines the actions that the IPS should take when it detects traffic that meets the criteria for each rule (i.e. that a particular Service has exceeded a specified connection limit). Also, within the context of a policy, specific rules can be enabled or disabled. Most rules define Boolean conditions (condition exists or does not exist), but some rules define conditions with limits, and with user-definable parameters.*
- *For example, this is where the administrator will define specific values for connection limits, bandwidth limits and SYN flood thresholds, to be applied later in a policy. It is possible to duplicate a rule with its associated parameters in order to create another rule with different parameter values, but it is not possible to create completely new rules from scratch.*

The *Filter Configuration* menu option brings together a number of security parameters that were scattered around the user interface in previous versions.

Here the administrator can define protocol-related restrictions such as maximum URI length, maximum URI path length, maximum ping length, maximum HTTP header length, maximum length of FTP authentication fields, maximum ICMP rate, and so on. Here is also where the patterns are defined for the *Payload String Matching* module. This is a much easier way to define the various limits and parameters than by creating multiple individual Rules, and it will be a huge improvement when the Filter Configuration menu becomes the main point of rule definition in a future release.

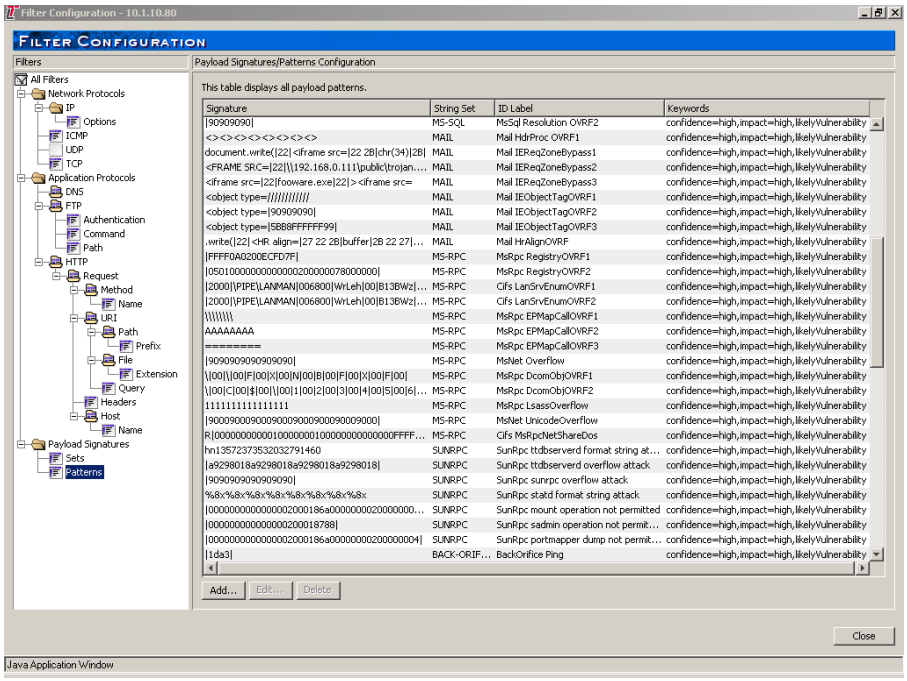

Figure 8 - Top Layer: Filter configuration

All of the pre-defined security objects and filter configurations are combined to make a security policy via the *Security Policies* menu option. Clicking the *Configure* button presents one or more windows listing the settings for each of the security subsystems:

- ***IP/ARP Packet Checks -*** *Examine the IP headers of IP traffic and the payload of ARP packets. These checks look for badly mangled packets that the IPS cannot reliably parse or even categorise by client group, server group, or application protocol.*
- ***Layer 2 Packet Checks -*** *Focuses on the MAC headers of network packets. Some of the checks look for malformed or abused fields in the MAC headers. Other checks represent normal bridging and VLAN filtering rules, which are controlled by the configuration of the Mission bridge on the IPS.*
- ***Firewall -*** *Provides classic firewall blocking for traffic, based on IP addresses, Layer 4 ports, and circuits (inside port versus outside port). The firewall's capabilities break down into the following sub-categories:*
    - *Blocks clients from internal or external circuits*

  - *Performs spoof checking (that is, enforce allowed ingress circuits per IP address range)*
  - *Enforces client access to services*
  - *Blocks packets from IP addresses deemed "malicious" by the SYN Flood mitigation subsystem.*
- ***Protocol Checks -*** *Examines the payload of application datagrams and application streams. This release performs deep packet and protocol inspection for the HTTP, DNS, MS-RPC, MS-CIFS, SSH, Telnet, and FTP protocols, with SMTP, POP3, and other protocol parsers on the way. For those protocols where a specific parser is not yet available, generic pattern matching can be used to detect malicious traffic.*
- ***SYN Flood Mitigation -*** *Tracks the dynamic behaviour of millions of individual IP addresses and manages the classification of those addresses into security risk categories. Helps protect against a flood of TCP connection requests that the attacker initiates, but does not complete, in an attempt to exhaust a server's ability to open additional new connections.*
- ***Connection Limiting -*** *Protects network resources (such as servers and routers) from being overwhelmed by too many active connections. This feature can be used to protect resources from both malicious activity and non-malicious, resource-intensive traffic demands. This feature limits the number of open connections allowed at one time from any and all of the following groups:*
  - *From all addresses in a client group*
  - *From any one address in a client group*
  - *To all servers in a server group*
  - *To a specific server in a server group.*
- ***Client Request Limiting -*** *Limits the number of requests/packets a client within a particular client group can generate and direct at a specific service or group of services within a defined period of time. An example of this would be to rate limit the number of DNS requests allowed from a particular IP address. Once the client exceeds the Request Limit, the IPS limits further requests for any service in the set of services identified as request-limited services. As additional protection, once a client exceeds the limit by a configurable "overdraft" amount, the IPS blocks all packets (regardless of service) from that source IP address. The IPS Unit initially performs client request limiting at close to the system's connection setup rate, but once the overdraft limit is exceeded, the blocking of subsequent requests is performed at much higher speeds.*

By default, each security subsystem on the device has a number of rules defined and enabled for it. This collection of rules and their associated items (actions, client groups, server groups, services, etc.) make up the security policy for each IPS subsystem. The configuration window provides a table showing each policy element, which consists of a rule and the associated item on which the rule acts.

For example, this is where the administrator can specify that **all** clients are to be assigned the default rate limit, **except** for a privileged small group of clients who will have no rate limit applied. As with a typical firewall, rules are processed in order, thus allowing the administrator to apply broad rules to begin with, and then refine them on a per-host basis.

One nice feature is the use of regular expressions within rules. This means that where the administrator wishes to restrict a rule to a specific group of services, for example, rather than define multiple rules - one for each service - a regular expression can be entered to cover them all with a single rule.

Unfortunately, it is possible only to **apply** previously a defined rule at this point in the GUI (the *default rate limit rule*, in this case) and not modify it or even view it. Thus, if you wish to check the actual rate limit or change it, you must first access the base rule in the *Security Configuration* window, before applying it in the *Security Policies* window. Occasionally, changes will also be required in the *Filter Configuration* window, since it duplicates some of the functionality of the *Security Configuration* window at the moment.

The use of rules which can be combined, and even re-used many times, to make security policies makes for a very flexible system. Unfortunately, the way it has been implemented to date makes if very difficult for the administrator, who often needs to make changes in multiple places to effect just a single amendment or addition to a security policy. Having to make changes in up to three different places can be confusing, making it difficult to locate exactly what needs to be changed.

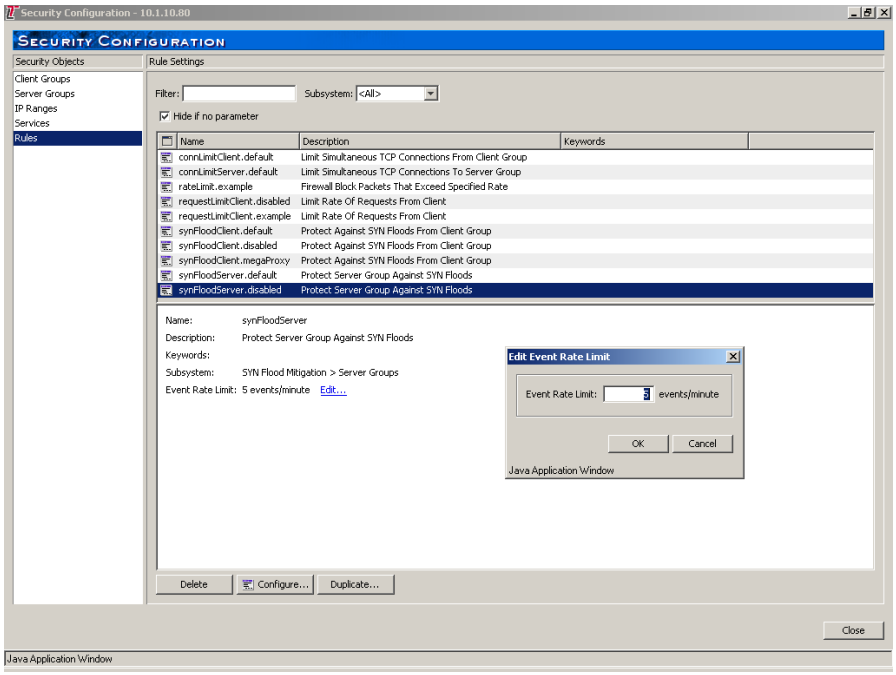

Figure 9 - Top Layer: Defining Rules

Top Layer is improving this with each new release as it moves more and more towards hiding the more obscure parameters and making the rest available through the Filter Configuration window – until that migration is complete, the current interface remains confusing.

The *Protocol Checks* subsystem is the mechanism which controls the deep packet inspection capabilities of the IPS 5500, and is configured via three tabs.

The *Rule Sets* tab contains the name of the set of rules and a brief description of the rule set.

The *Rule Set Actions* tab contains the actual list of rules within each *Rule Set*, and this tab controls which separate protocol check capabilities are in effect and which of the following actions to apply when packets match the rules:

- ***Copy to Discard Port -*** *Blocked traffic is copied to the Discard port, where it can be further processed by an external system (i.e. forensic recorder or IDS, if required.*
- ***Log -*** *The IPS sends all messages matching this rule to the syslog servers which have been configured.*
- ***Log Priority -*** *Provides an overall threshold that messages must have before the IPS processes them. Settings follow Syslog severity conventions with zero indicating the most severe events and seven the lowest*
- ***Detect Only -*** *The IPS has detected an occurrence of the rule. The attack is logged (if required) but not blocked*
- ***Passive Block -*** *The packet that caused the match is silently blocked. If the packet was part of an established flow, subsequent packets that arrive on the same flow are also dropped.*
- ***Active Block -*** *The packet that caused the match is blocked in the same way as the Passive Block, and the IPS then resets the connection.*

The first line of the Rule Set Actions causes the specified actions to be applied to all rules by default. It is then possible to create additional Rule Set Actions to override the default on a case by case basis. Regular expressions can be used to specify groups of signatures in a single line, or individual signatures can be specified.

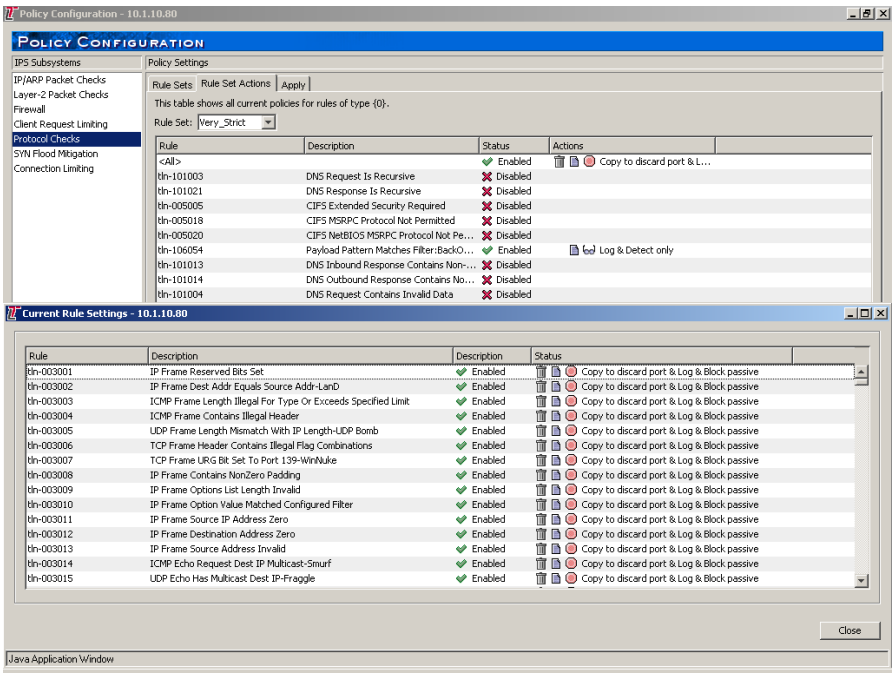

Figure 10 - Top Layer: Defining Rule Set Actions

With each additional Rule Set Action created the administrator can disable the specified signatures, or override any of the Actions defined in the default rule.

A search capability allows the administrator to search the available signatures. However, it is not easy to find exactly the signature you require if all you have to go on is a CVE reference or a partial name, since the Top Layer signatures do not have a CVE reference associated with them, and often have very generic or confusing names.

An *Evaluate* button provides the means to verify the final policy by applying all the Rules to the available signatures and listing them on screen individually with the appropriate Rule Set Actions against each one. This is a very useful feature, and essential when using regular expressions to confirm that they have had exactly the desired effect.

To make life simple, four complete sample Rule Sets are provided out of the box – *Detect Only, Recommended, Strict,* and *Very Strict*. We tested using the *Very Strict* policy, since this enables **all** Rules, but most organisations would probably start with the *Recommended* policy and refine it as required.

Every signature has a *Confidence Level, Impact* and *Likely Vulnerability* indicator against it to highlight those which are the biggest threat, and those which are more for auditing purposes. The *Recommended* policy excludes all signatures where the *Confidence Level* is *Low*, and all those which are not considered to be likely vulnerabilities and where the potential *Impact* is *Low*.

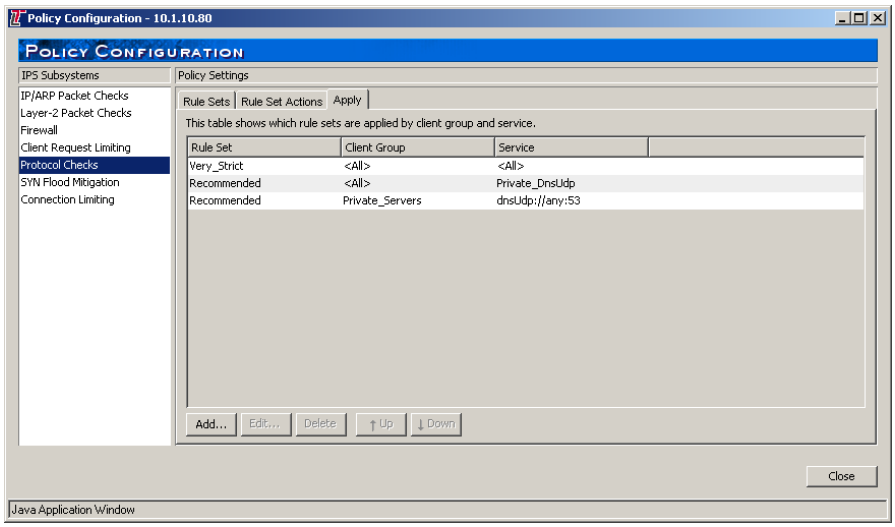

Figure 11 - Top Layer: Applying Rule Sets

Finally, the *Apply* tab in the *Protocol Checks* subsystem determines which rule sets are applied to selected client groups and services. This means that it is possible to define a strict policy for the majority of clients, whilst overriding that with a more lenient policy for a select group, and applying a different policy again to all Web applications. Any number of policies can be applied in this way, making this a very flexible system.

Alert Handling

The IPS 5500 event management processor configuration allows granular configuration of the events generated - which should be logged, which sent to which syslog servers, and so on - as well as incorporating an event suppression mechanism to prevent log servers becoming overwhelmed by event volume.

Whenever a Rule is matched by network traffic, in addition to the Rule Set Actions defined as part of the Security Policy, the IPS 5500 also generates an alert which is displayed in the *Blocked and Detected Attacks* window.

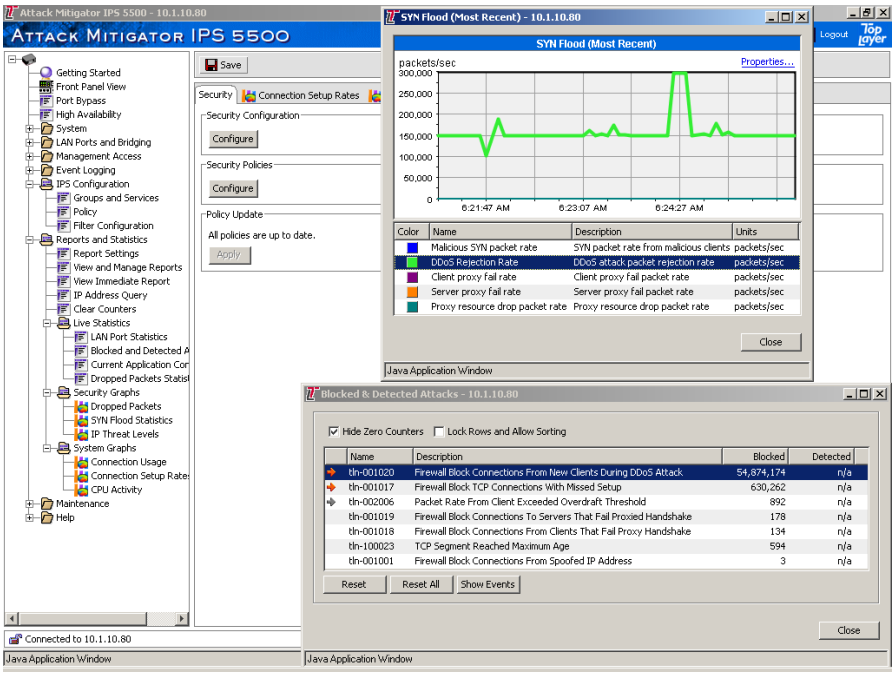

Figure 12 - Top Layer: Monitoring Blocked and Detected Attacks

This provides a one-line summary for each attack detected, with the Top Layer attack ID, attack description, and a total count of all instances of this attack which have been blocked, or detected only. Unfortunately, there is no CVE reference or other similar means to accurately identify an exploit, and no link from this window to any external database to provide further information (Top Layer plans to introduce this capability in a forthcoming release). Add to this the fact that the Top Layer descriptions frequently bear little or no relationship to the actual exploit (for example, "*Payload Packet Matches Filter: Bad URI5*") and forensic analysis soon becomes more difficult than it needs to be. Rate-based alerts are much more descriptive, of course, but then, there is little or no forensic analysis to perform for those anyway.

Selecting any individual summary line from the *Blocked and Detected Attacks* window and clicking on the *Show Events* button brings up another window listing the individual events which were detected for that attack. These are listed in a table containing the protocol, source IP/port, destination IP/port, origin (the port on the IPS appliance) and a time stamp.

There is no separate column for source or destination port, and no means to group together by IP address or port in order to try to identify trends like *"which IP address is launching the most attacks at our Web server?"*. Data can be sorted by clicking on column headers, but more sophisticated means of slicing and dicing data would be welcome.

One thing that *is* available is a complete raw packet display. This can be very useful when performing forensic analysis (and is often the only way to identify just what triggered an alert, given that the alert names are often so unhelpful), but we can't help but feel that more extensive drill-down and dynamic reporting capabilities will be sorely missed by most administrators.

The IPS 5500 provides local storage on the built-in compact flash card for recent events. These are stored in files, and are overwritten once the maximum storage capacity for events is reached, which is seven files, each 1Mbyte in size. These files can be downloaded via the management application if required.

Reporting and Analysis

The IPS5500 reporting capabilities comprise the following:

- *A device management application with real-time statistics covering detected and blocked security events, network traffic and utilisation, and IPS system capacity and utilisation*
- *A periodic security report, which can be configured to include security information and additional diagnostic information, which can be generated on an hourly, multiple hourly, or daily basis.*
- *An immediate security report, which can be generated at any time.*
- *A syslog-based event reporting scheme which is supported by third-party event management systems.*

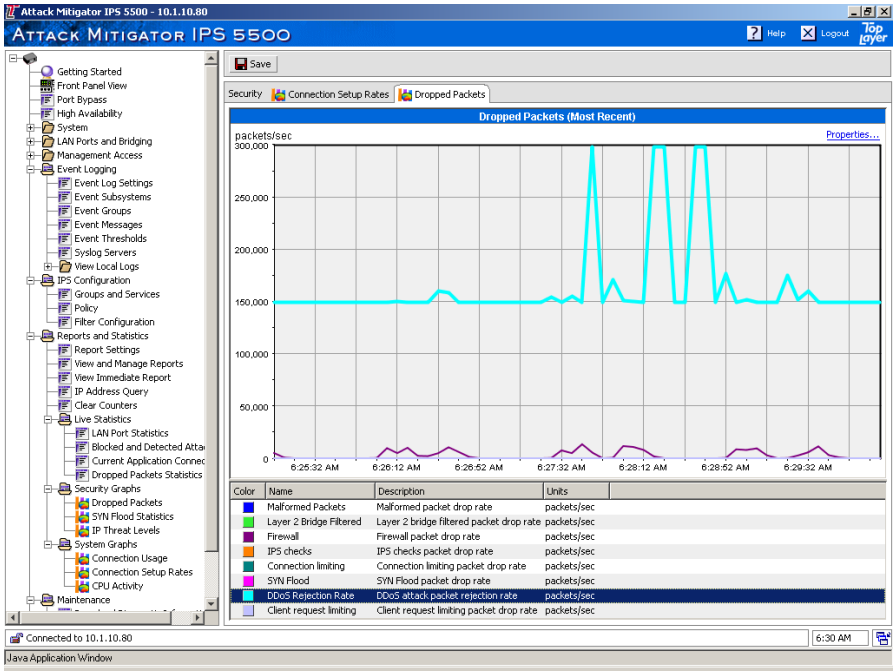

Figure 13 - Top Layer: Monitoring dropped packets

The Management Application provides several tables providing *Live Statistics* of ongoing IPS operations, which are accessible from the Navigation Tree:

- ***LAN Port Statistics -*** *Port statistics for each port.*
- ***Blocked and Detected Attacks –*** *Summary screen showing total number of times each type of event was blocked or detected*
- ***Current Application Connections -*** *Total number of current connections for each type of network application that the IPS tracks.*
- ***Dropped Packet Statistics -*** *Number of times a packet was dropped for each drop reason.*

The following *Security Graphs* are also available:

- ***Dropped Packets -*** *Provides an indication of the number of packets dropped by the different IPS subsystems and checks.*
- ***SYN Flood Statistics -*** *Provides a representation of various packets received, dropped, and failed rates related to SYN Flood attacks.*
- ***IP Threat Levels -*** *Visually represents the number of IP addresses in each of the threat level categories.*

Finally, the following *System Graphs* are available:

- ***Connection Usage -*** *Displays the types of connections that represent the traffic going through the IPS. Includes the number of TCP, UDP, and Other IP connections, as well as number of Aged Connections during the selected time period.*
- ***Connection Setup -*** *Displays the current rate of setup for various types of connections such as TCP and UDP.*
- ***CPU Activity -*** *Displays the CPU usage for different activities such as TCP connection setups.*

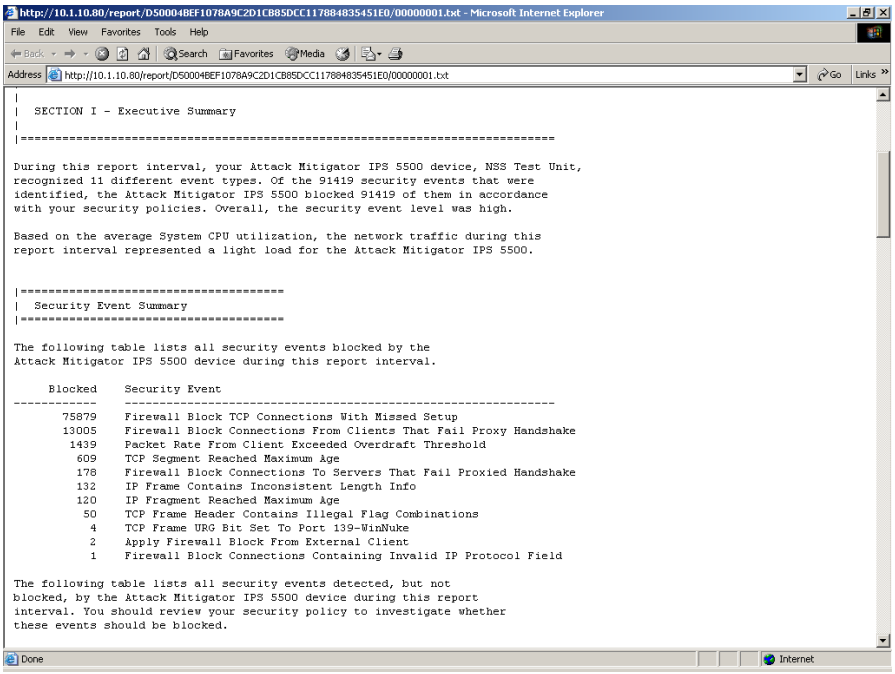

Figure 14 - Top Layer: Security Report

While the various graphs are extremely useful for monitoring purposes, particularly during an attack, the overall reporting capabilities of the IPS 5500 are very limited, although more advanced reporting capabilities are offered in the optional CMS product.

The only report to come out of the Management Application, however, is an all-in-one *Security Report*, which can either be scheduled to run periodically, or can be created on demand.

This is a text-based report that contains a “snapshot” summary of the types of statistics displayed in the various graphs.

The reports covers the entire period between one scheduled report and the next, or the last scheduled report and the current *Immediate Report* – it is not possible to specify a custom reporting period.

From the *View and Manage Reports* window the administrator can select and view (or delete) all reports generated by the IPS 5500. The IPS holds these reports in its volatile memory, and can store 50 to 100 periodic Security Reports, depending on the amount of data included in each report. Once the memory space for the reports is full, old reports are deleted automatically to make room for new ones. All currently stored reports are also lost when the IPS Unit reboots.

We would prefer to see provision for making these reports more permanent, perhaps by uploading to an FTP server on the management network at regular intervals.

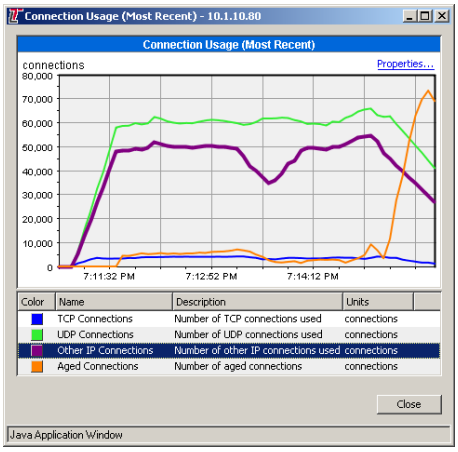

Figure 15 - Top Layer: Monitoring current connections

The Security Report is divided into three sections:

- ***Executive Summary –*** *Provides a high level overview of the appliance's performance and security events it encountered during the reporting period. Includes an overview of how many different types of events the IP detected, total number of events detected and blocked, and the overall traffic load on the IPS, plus a Security Event Summary (events detected and blocked listed by name) and Blocked Packet Details (total number of blocked packets listed by attack category).*
- ***System Resource Details –*** *Provides usage statistics for the main traffic handling processor, including system processor utilisation, system session table usage, system session set-up rate, system IP address summary, and LAN port utilisation.*
- ***System Diagnostic Information –*** *Lists the number of times the IPS entered into the various levels of CPU overload protection, number of times system resource limits were exceeded, and SYN flood mitigation details. This section is optional.*

Unfortunately, this report is completely static, and it is not possible to drill down into any of the various tables to examine that data that lies behind them.

Raw *Event Log* (system events, such as port failures, etc.) and *Alert Log* (security events) contents can be viewed directly from the Management Application, or downloaded to a local hard drive for further analysis.

Verdict

Performance

The IPS 5500 is rated by Top Layer for a single Gigabit link (2Gbps aggregate throughput) and was tested to 1Gbps here. It turned in an outstanding performance in all our tests, achieving 100 per cent detection rates across the board, and clearly with some headroom to spare. We would be more than happy to rate this device at a minimum of 1Gbps under **all** network loads.

Basic latency figures were outstanding - almost switch like - across the board under all traffic loads. They ranged from 20µs with 250Mbps of 256 byte packets, to 41µs with 1Gbps of 1000 byte packets. Behaviour throughout the tests with no background traffic was extremely consistent and predictable, hardly increasing at all as additional network load was applied from 250Mbps to 1Gbps.

The IPS 5500 was also one of the few devices we have tested in our labs which has achieved zero packet loss and low latency at **all** packet sizes (including 64 bytes) up to 1Gbps. The latency with 64 byte packets at 1Gbps was just 17µs (which also includes the basic latency of the test infrastructure).

Placing the device under a half load of 500Mbps of HTTP traffic, we noted some slight increases in latency, though the figures still never climbed above 62µs. HTTP response times were also excellent.

100Mbps of SYN flood traffic had a negligible effect on the IPS 5500, increasing the base latencies at all packet sizes by around a microsecond only. The SYN flood was mitigated completely once it had been detected (which did not take long). Note that this is not the case with all DOS/DDOS attacks, however. With many other types of attack, the IPS 5500 does “mitigate” (i.e. reduce) the effects of the attack rather than block it completely.

Overall, latency figures were considered to be outstanding for a device of this type under all load conditions and packet sizes. Clearly this device can be placed anywhere on the corporate network - from the perimeter to a heavily-loaded high-speed backbone - without impacting overall network performance in any way.

The IPS 5500 performed consistently and completely reliably throughout our tests, continuing to block attack traffic in a consistent manner whilst passing 100 per cent of the legitimate traffic, even when under extended attack.

In the rate-based (attack mitigator) attacks the IPS 5500 performed equally well. Mitigating high-volume rate-based attacks is a different prospect to detecting and blocking single-packet exploits, and often two different devices are deployed to achieve both. The Top Layer device is currently one of only a few devices on the market capable of completing **both** our content- and rate-based methodologies. Performance at all levels of our load tests was impeccable, with 100 per cent of all attacks being detected and mitigated under all load conditions, and no interruptions to legitimate sessions. Latency too was very low across all tests, even when under heavy DOS attack.

DDOS attacks (multiple source IPs) proved trickier to handle, with CPU becoming a bottleneck much earlier (between 200Mbps and 400Mbps), causing packet loss. the IPS5500 device is rated by Top Layer for DDoS protection at up to 500,000pps, (approximately 333Mbps with 64 byte packets), and the ProtectionCluster feature (not tested) can be used to scale this solution to higher rates.

Overall latency performance under all normal and DOS conditions was considered to be excellent, and HTTP response times remained remarkably consistent throughout all our DOS attacks.

Security Effectiveness

Top Layer has made significant additions to its protocol decode and validation modules, and significant enhancements to its signature set for this release. Signature recognition (with blocking disabled) was good out of the box (85 per cent), and was increased to a creditable 94 per cent after the application of a signature pack update which was provided to us in 48 hours. Blocking performance was identical throughout the tests.

We noted a minimum of “noise”, with very few test cases raising multiple alerts for a single exploit. Performance in our “false negative” tests was reasonable out of the box, but did not improve following the signature update.

A major concern in deploying an IPS is the blocking of legitimate traffic. Providing the signatures which are disabled by default in the standard policies are left that way, the IPS 5500 resistance to false positives is good. However, it should be noted that when the various attack mitigation features are employed, careful tuning of the mitigation parameters is required in order that legitimate traffic is not blocked accidentally.

There is no automatic “learning” capability, meaning the responsibility for determining the optimum mitigation thresholds lies squarely with the administrator.

Resistance to known evasion techniques was excellent, with the IPS 5500 achieving a clean sweep across the board in most of our evasion tests. *Fragroute* and *Whisker* both failed to trick the device into ignoring valid attacks. Not only were the fragmented and obfuscated attacks blocked successfully, but many of them were decoded accurately as well.

Out of the box, the IPS 5500 is designed to handle 1,000,000 (one million) open connections without tuning. It was thus able to handle our 1 million open connection test with ease. Stateless “exploits” – or mid-flows - are handled correctly.

Configuring for the rate-based (attack mitigator) attacks was tricky, requiring much more care and consideration in order to avoid self-imposed DOS conditions. However, once configured, the device detected and mitigated most of our attacks successfully.

We feel that the IPS 5500 is lacking in its ability to detect and mitigate certain scan and probe attempts, relying on its firewall filtering to catch some, and mitigating others only partially. Scan and probe attempts aside, however, the DOS and DDOS mitigation proved to be excellent, as did the resistance to common evasion techniques.

Performance in the high volume detection/mitigation test was almost impeccable across the board, with perfect detection and mitigation at all load levels.

Some problems were noted in passing legitimate traffic at the highest load levels of some of the DDOS attacks due to high CPU utilisation and subsequent packet loss. However, these load levels can be considered excessive, and the device performed almost impeccably up to the 600Mbps level of attack traffic.

Usability

It is important with in-line devices such as this that sufficient features are given over to the task of traffic profiling, and the IPS does provide some good graphical monitoring tools to help determine optimum bandwidth and connection rates for various applications before limiting traffic.

Both reporting and alert management are extremely basic, but the main job of the IPS 5500 is to stop malicious or suspicious traffic rather than analyse it, and it performs its blocking and mitigation tasks well.

Most users would probably be content to leave it at the fact that the bad traffic never made it onto their network, but for those who want additional forensic analysis on the mitigated traffic, the IPS does provide the Forensic port to route that traffic to a third party collection and analysis product. Top Layer also recommends the SecureCommand+ CMS as the preferred solution for more advanced reporting and event management, but we did not have the opportunity to test this product.

Providing you are willing to accept the default settings provided by Top Layer, getting the IPS 5500 up and running is fairly straightforward. However, we found that we needed to make quite a few changes to the default settings – particularly for our Attack Mitigator tests – and we found that it was not an easy matter to configure the device to behave exactly as you would like.

Frequently, changes made in one part of the system would have a knock on effect elsewhere, and the first day or two seemed to be a constant round of refining and testing in order to get everything running smoothly and reduce the risk of false positives or accidental, self-inflicted Denial of Service conditions.

This was made harder by the fact that, in general, the IPS 5500 Management Application is still too confusing in places. Changes need to be made in two or three different places sometimes just to effect a single adjustment to the configuration, and this makes life hard on the administrator.

Having said that, although the learning curve is steep, day to day running of the device is well catered for by the Management Application. Significant improvements have been made since the last time we looked at it, and more are on the cards for future releases. Given that it is designed to connect to a single device at a time, the Management Application clearly will not scale well in larger deployments.

For those, Top Layer has produced the central management platform, SecureCommand+, which provides centralised management and reporting, as well as correlation across all devices and post-processing of events in order to catch “low and slow” attacks such as slow port scans.

As such, SecureCommand+ provides a richer set of historical trending and query-based reporting and analysis features than the basic Management Application, albeit at additional cost.

Contact Details

Company name: Top Layer Networks

E-mail: info@toplayer.com

Internet: www.toplayer.com

Address:
2400 Computer Drive
Westboro
MA 01581
USA

Tel: +1 508 870 1300
Fax: +1 508 870 9797

APPENDIX A – TEST RESULTS (CONTENT-BASED)

The aim of this procedure is to provide a thorough test of all the main components of an in-line Intrusion Prevention System (IPS) 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 network is 100/1000Mbit Ethernet with CAT 5e cabling and Cisco 6500-Series switches (these have a mix of fibre and copper Gigabit interfaces). All devices are expected to be provided as appliances - if software-only, the supplier pre-installs the software on the recommended hardware platform. The sensor is configured as a perimeter device during testing (i.e. as if installed behind the main Internet gateway/firewall). There is no firewall protecting the target subnet.

Traffic generation equipment - such as the machines generating exploits, Spirent Avalanche and Spirent Smartbits *transmit* port - is connected to the “external” network, whilst the “receiving” equipment - such as the “target” hosts for the exploits, Spirent Reflector and Spirent Smartbits *receive* port - is connected to the internal network. The device under test 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 device under test, from external to internal. The same traffic is mirrored to a single SPAN port of the external gateway switch, to which an Adtech network monitoring device is connected. The Adtech AX/4000 monitors the same mirrored traffic to ensure that the total amount of traffic never exceeds 1Gbps (which would invalidate the test run).

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.

Section 1 – Detection Engine

The aim of this section is to verify that the sensor is capable of detecting and blocking a wide range of common exploits accurately, whilst remaining resistant to false positives. All tests in this section are completed with **no background network load**. The latest signature pack is acquired from the vendor, and sensors are deployed with **all** available attack signatures enabled (some audit/informational signatures may be disabled).

Test 1.1 - Attack Recognition

Whilst it is not possible to validate completely the entire signature set of any sensor, this test attempts to demonstrate how accurately the sensor detects and blocks a wide range of common exploits, port scans, and Denial of Service attempts. These are updated/changed for every new test, and all exploits are run with no load on the network and no IP fragmentation.

Our attack suite contains over 100 basic exploits (plus variants) covering the following areas:

- ***Test 1.1.1 -*** *Backdoors (standard ports and random ports)*
- ***Test 1.1.2 -*** *DNS/WINS*
- ***Test 1.1.3 -*** *DOS*
- ***Test 1.1.4 -*** *False negatives (common exploits which have been modified to remove or alter obvious "triggers" - this ensures that the signatures are coded for the underlying vulnerability rather than a particular exploit)*
- ***Test 1.1.5 -*** *Finger*
- ***Test 1.1.6 -*** *FTP*
- ***Test 1.1.7 -*** *HTTP*
- ***Test 1.1.8 -*** *ICMP (including unsolicited ICMP response)*
- ***Test 1.1.9 -*** *Reconnaissance*
- ***Test 1.1.10 -*** *RPC*
- ***Test 1.1.11 -*** *SSH*
- ***Test 1.1.12 -*** *Telnet*
- ***Test 1.1.13 -*** *Database*
- ***Test 1.1.14 -*** *Mail*
- ***Test 1.1.15 -*** *Voice*

A wide range of vulnerable target operating systems and applications are used, and the majority of the attacks are successful, gaining root shell or administrator privileges on the target machine.

We expect all the attacks to be reported in as straightforward and clear a manner as possible (i.e. an "RDS MDAC attack" should be reported as such, rather than a "Generic IIS Attack"). Wherever possible, attacks should be identified by their assigned CVE reference. It will also be noted when a response to an exploit is considered too "noisy", generating multiple similar or identical alerts for the same attack. Finally, we will note whether the device blocks the attack packet only or the entire "suspicious" TCP session.

This test is repeated twice: the first run with blocking disabled on the sensor (monitor mode only) in order to determine which attacks are detected and how accurately they are detected (*Attack Recognition Rating*); the second run with blocking enabled in order to determine which attacks are blocked successfully regardless of how they are detected or what alerts are raised (*Attack Blocking Rating*)

The "**default**" *Attack Recognition Rating-Detect Only* (ARRD) and *Attack Recognition Rating-Block* (ARRB) are each expressed as a percentage of detected/blocked exploits against total number of exploits launched with the default signature set as received by NSS. This demonstrates how effective the sensor can be when simply deploying the default configuration.

Following the initial test run, each vendor is provided with a list of CVE references of the attacks missed, and is then allowed 48 hours to produce an updated signature set. This updated signature set **must** be released to the general public as a standard signature/product update before the report is published - this ensures that vendors do not attempt to code signatures just for this test.

The sensor is then exposed to a second round of identical tests and the "**custom**" ARRD/ARRB is determined. This demonstrates how effective the vendor is at responding to a requirement for new or updated signatures.

Both the *default* and *custom* ARRD/ARRB figures are reported.

Test 1.2 - Resistance To False Positives

The aim of this test is to demonstrate how likely it is that a sensor raises a false positive alert - particularly critical for IPS devices.

We have a number of trace files of normal traffic with "suspicious" content, together with several "neutered" exploits which have been rendered completely ineffective. If a signature has been coded for a specific piece of exploit code rather then the underlying vulnerability, or if it relies purely on pattern matching, some of these false alarms could be alerted upon.

The product attains a "PASS" for each test case if it does **not** raise an alert and does **not** block the traffic. Raising an alert on any of these test cases is considered a "FAIL", since none of the "exploits" used in this test represents a genuine threat. A "FAIL" would thus indicate the chance that the sensor could block legitimate traffic inadvertently.

- ***Test 1.2.1 -*** *False positives*

Section 2 – Evasion

The aim of this section is to verify that the sensor is capable of detecting and blocking basic exploits when subjected to varying common evasion techniques.

Test 2.1 - Baselines

The aim of this test is to establish that the sensor is capable of detecting and blocking a number of common basic attacks (our baseline suite) in their normal state, with no evasion techniques applied. Note that common/older attacks have been chosen deliberately for this particular test to ensure that ALL products tested have signatures in place for the evasion tests.

- ***Test 2.1.1 -*** *Baseline attack replay*

Test 2.2 - Packet Fragmentation and Stream Segmentation

The baseline HTTP attacks are repeated, running them through fragroute using various evasion techniques, including:

- ***Test 2.2.1 -*** *IP fragmentation - ordered 8 byte fragments*
- ***Test 2.2.2 -*** *IP fragmentation - ordered 24 byte fragments*
- ***Test 2.2.3 -*** *IP fragmentation - out of order 8 byte fragments*
- ***Test 2.2.4 -*** *IP fragmentation - ordered 8 byte fragments, duplicate last packet*
- ***Test 2.2.5 -*** *IP fragmentation - out of order 8 byte fragments, duplicate last packet*
- ***Test 2.2.6 -*** *IP fragmentation - ordered 8 byte fragments, reorder fragments in reverse*

- ***Test 2.2.7 -*** *IP fragmentation - ordered 16 byte fragments, fragment overlap (favour new)*
- ***Test 2.2.8 -*** *IP fragmentation - ordered 16 byte fragments, fragment overlap (favour old)*
- ***Test 2.2.9 -*** *TCP segmentation - ordered 1 byte segments, interleaved duplicate segments with invalid TCP checksums*
- ***Test 2.2.10 -*** *TCP segmentation - ordered 1 byte segments, interleaved duplicate segments with null TCP control flags*
- ***Test 2.2.11 -*** *TCP segmentation - ordered 1 byte segments, interleaved duplicate segments with requests to resync sequence numbers mid-stream*
- ***Test 2.2.12 -*** *TCP segmentation - ordered 1 byte segments, duplicate last packet*
- ***Test 2.2.13 -*** *TCP segmentation - ordered 2 byte segments, segment overlap (favour new)*
- ***Test 2.2.14 -*** *TCP segmentation - ordered 1 byte segments, interleaved duplicate segments with out-of-window sequence numbers*
- ***Test 2.2.15 -*** *TCP segmentation - out of order 1 byte segments*
- ***Test 2.2.16 -*** *TCP segmentation - out of order 1 byte segments, interleaved duplicate segments with faked retransmits*
- ***Test 2.2.17 -*** *TCP segmentation - ordered 1 byte segments, segment overlap (favour new)*
- ***Test 2.2.18 -*** *TCP segmentation - out of order 1 byte segments, PAWS elimination (interleaved dup segs with older TCP timestamp options)*
- ***Test 2.2.19 -*** *IP fragmentation - out of order 8 byte fragments, interleaved duplicate packets scheduled for later delivery*
- ***Test 2.2.20 -*** *TCP segmentation - ordered 16 byte segments, segment overlap (favour new (Unix))*

For each of the evasion techniques, we note if (i) the attempted attack is blocked successfully (the primary aim of any IPS device), (ii) the attempted attack is detected and an alert raised in **any** form, and (iii) if the exploit is successfully “decoded” to provide an accurate alert relating to the original exploit, rather than alerting purely on anomalous traffic detected as a result of the evasion technique itself.

Test 2.3 - URL Obfuscation

The baseline HTTP attacks are repeated, this time applying various URL obfuscation techniques made popular by the Whisker Web server vulnerability scanner, including:

- ***Test 2.3.1 -*** *URL encoding*
- ***Test 2.3.2 -*** */./ directory insertion*
- ***Test 2.3.3 -*** *Premature URL ending*
- ***Test 2.3.4 -*** *Long URL*
- ***Test 2.3.5 -*** *Fake parameter*
- ***Test 2.3.6 -*** *TAB separation*
- ***Test 2.3.7 -*** *Case sensitivity*
- ***Test 2.3.8 -*** *Windows \ delimiter*
- ***Test 2.3.9 -*** *Session splicing*

For each of the evasion techniques, we note if (i) the attempted attack is blocked successfully, (ii) the attempted attack is detected and an alert raised in **any** form, and (iii) if the exploit is successfully "decoded" to provide an accurate alert relating to the original exploit, rather than alerting purely on anomalous traffic detected as a result of the evasion technique itself.

Test 2.4 - Miscellaneous Evasion Techniques

Certain baseline attacks are repeated, and are subjected to various protocol- or exploit-specific evasion techniques, including:

- ***Test 2.4.1 -*** *Altering default ports/passwords for backdoors*
- ***Test 2.4.2 -*** *Inserting spaces in FTP command lines*
- ***Test 2.4.3 -*** *Inserting non-text Telnet opcodes in FTP data stream*
- ***Test 2.4.4 -*** *Polymorphic mutation (ADMmutate)*
- ***Test 2.4.5 -*** *Altering protocol and RPC PROC numbers*
- ***Test 2.4.6 -*** *RPC record fragging (MS-RPC and Sun)*
- ***Test 2.4.7 -*** *HTTP exploits to non-standard port*

For each of the evasion techniques, we note if (i) the attempted attack is blocked successfully, (ii) the attempted attack is detected and an alert raised in **any** form, and (iii) if the exploit is successfully "decoded" to provide an accurate alert relating to the original exploit, rather than alerting purely on anomalous traffic detected as a result of the evasion technique itself.

Section 3 – Stateful Operation

The aim of this section is to be able to determine whether the sensor is capable of monitoring stateful sessions established through the device at various traffic loads without either losing state or incorrectly inferring state.

Test 3.1 - Stateless Attack Replay (Mid-Flows)

This test determines whether the sensor is resistant to stateless attack flooding tools - these utilities are used to generate large numbers of false alerts on the protected subnet using valid source and destination addresses and a range of protocols.

The main characteristic of many flooding tools is the fact that they generate single packets containing "trigger" patterns without first attempting to establish a connection with the target server. Whilst this can be effective in raising alerts with some stateless protocols such as UDP and ICMP, they should never be capable of raising an alert for exploits based on stateful protocols such as FTP and HTTP.

In this test, we transmit a number of packets taken from capture files of valid exploits, but without first establishing a valid session with the target server. We also remove the session tear down and acknowledgement packets so that the sensor can not "infer" that a valid connection was made.

In order to receive a "PASS" in this test, no alerts should be raised for any of the actual exploits (although "mid-flow" alerts are permitted).

However, each packet should be blocked if possible since it represents a "broken" or "incomplete" session.

- ***Test 3.1.1 -*** *Stateless attack replay*

Test 3.2 - Simultaneous Open Connections (default settings)

This test determines whether the sensor is capable of preserving state across increasing numbers of open connections, as well as continuing to detect and block new exploits when the state tables are filled. It also attempts to determine whether or not the sensor will block legitimate traffic once state tables are filled. This test is run using the default sensor settings (no tuning of sensor parameters).

A legitimate HTTP session is opened and the first packet of a two-packet exploit is transmitted. The Spirent Avalanche (on the "external" interface of the sensor) then opens various numbers of TCP sessions from 10,000 to 1,000,000 (one million) with the Spirent Reflector (on the "internal" interface of the sensor). 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 first session established, 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.

At each step, we ensure that the sensor is still capable of detecting and blocking freshly-launched exploits once all the connections are open, as well as confirming that the device does not block legitimate traffic (perhaps as a result of state tables filling up). We then launch further exploits whilst the Avalanche/Reflector devices "churn" connections at the maximum level set, ensuring that the sensor is still capable of detecting and blocking freshly-launched exploits as old connections are torn down and new ones recreated constantly.

- ***Test 3.2.1 - Attack Detection:*** *This test ensures that the sensor continues to detect new exploits as the number of open sessions is increased in stages from 10,000 to 1,000,000*
- ***Test 3.2.2 - Attack Blocking:*** *This test ensures that the sensor continues to block new exploits as the number of open sessions is increased in stages from 10,000 to 1,000,000*
- ***Test 3.2.3 - State Preservation:*** *This test ensures that the sensor maintains the state of pre-existing sessions as the number of open sessions is increased in stages from 10,000 to 1,000,000*
- ***Test 3.2.4 - Legitimate Traffic Blocking:*** *This test ensures that the sensor does not begin to block legitimate traffic as the number of open sessions is increased in stages from 10,000 to 1,000,000*

Test 3.3 - Simultaneous Open Connections (after tuning)

Test 3.2 is repeated after any tuning recommended by the vendor (if applicable) to increase the size of the state tables.

- ***Test 3.3.1 - Attack Detection:*** *As Test 3.2.1 following tuning*
- ***Test 3.3.2 - Attack Blocking:*** *As Test 3.2.2 following tuning*
- ***Test 3.3.3 - State Preservation:*** *As Test 3.2.3 following tuning*
- ***Test 3.3.4 - Legitimate Traffic Blocking:*** *As Test 3.2.4 following tuning*

Section 4 – Detection/Blocking Performance Under Load

The aim of this section is to verify that the sensor is capable of detecting and blocking exploits when subjected to increasing loads of background traffic up to the maximum bandwidth supported as claimed by the vendor.

The latest signature pack is acquired from the vendor, and sensors are deployed with **all** available attack signatures enabled (some audit/informational signatures may be disabled). Each sensor is configured to **detect and block** suspicious traffic.

Our "attacker" host launches a fixed number of exploits at a target host on the subnet being protected by the device under test. The Adtech network monitor is configured to monitor the switch SPAN port consisting of normal, exploit and background traffic, and is capable of reporting the total number of exploit packets seen on the wire as verification.

A fixed number of exploits are launched with zero background traffic to ensure the sensor is capable of detecting our baseline attacks. Once that has been established, increasing levels of varying types of background traffic are generated **through** the sensor in order to determine the point at which the sensor begins to miss attacks - all tests are repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic (or up to the maximum rated throughput of the device should this be less than 1Gbps).

At all stages, the Adtech network monitor verifies both the overall traffic loading and the total number of exploits seen on the target subnet. An additional confirmation is provided by the target host which reports the number of exploits which actually made it through.

The *Attack Blocking Rate* (ABR) at each background load is expressed as a percentage of the number of exploits blocked by the sensor (when in blocking mode) against the number verified by the Adtech network monitor and target host. The *Attack Detection Rate* (ADR) at each background load is expressed as a percentage of the number of exploits detected by the sensor (with blocking mode disabled) against the number verified by the Adtech network monitor and target host.

For each type of background traffic, we also determine the maximum load the sensor can sustain before it begins to drop packets/miss alerts. It is worth noting that devices which demonstrate 100 per cent ABR (blocking) but less than 100 per cent ADR (detection) in these tests will be prone to blocking **legitimate** traffic under similar loads.

Test 4.1 - UDP Traffic To Random Valid Ports

This test uses UDP packets of varying sizes generated by a **Smartbits SMB6000** with LAN-3301A 10/100/1000Mbps **TeraMetrics** cards installed.

A constant stream of the appropriate mix of packets - with variable source IP addresses and ports transmitting to a single fixed IP address/port - is transmitted through the sensor (bi-directionally, maximum of 1Gbps).

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 are verified by the Adtech Gigabit network monitoring tool before each test begins. Multiple tests are run and averages taken where necessary.

This traffic does not attempt to simulate any form of “real world” network condition. The aim of this test is purely to determine the raw packet processing capability of the sensor, and its effectiveness at passing “useless” packets quickly in order to pass potential attack packets to the detection engine. The range of packet sizes has been selected to mirror the maximum, minimum and average packet sizes used in our HTTP stress tests.

- ***Test 4.1.1 - 256 byte packets - maximum 453,000 packets per second:*** *This test is roughly equivalent to a 40,000 connections per second test in our HTTP stress tests (in terms of packet size and packets per second rate), and has been included to provide an indication of the packet processing performance under the most extreme conditions for most devices - it is unlikely that any real-life network will ever see network loads of over 450,000 256-byte packets per second unless under severe DOS conditions. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic.*
- ***Test 4.1.2 - 550 byte packets - maximum 220,000 packets per second:*** *This test has been included to provide a comparison with our “real world” packet mixes, since the average packet size is similar. No sessions are created during this test and there is very little for the detection engine to do in the way of protocol analysis. This test provides a reasonable indication of the ability of a device to process packets from the wire on an “average” network, and we would expect all products to demonstrate good performance levels. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic.*
- ***Test 4.1.3 - 1000 byte packets - maximum 122,000 packets per second:*** *This test is the complete opposite of the 256 byte packet test, in that we would expect every single product to be capable of returning 100 per cent detection rates across the board when using only 1000 byte packets. We have included this test 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 (or larger) packet sizes. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic.*

Test 4.2 - HTTP “Maximum Stress” Traffic With No Transaction Delays

HTTP is the most widely used protocol in most normal networks, as well as being one of the most widely exploited. The number of potential HTTP exploits for the protocol makes a pure HTTP network something of a torture test for the average sensor.

The use of multiple Spirent Communications **Avalanche 2500** and **Reflector 2500** devices allows us to create true “real world” traffic at speeds of up to 4.2 Gbps as a background load for our tests. Our Avalanche configuration is capable of simulating over 5 million users, with over 5 million concurrent sessions, and over 200,000 HTTP requests per second.

By creating genuine session-based traffic with varying session lengths, the sensor is forced to track valid 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, whilst ensuring absolute accuracy and repeatability.

The aim of this test 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.

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.

- ***Test 4.2.1 -*** *Max 2,500 new connections per second - average packet size 1000 bytes - maximum 120,000 packets per second. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With relatively low connection rates and large packet sizes, we expect all sensors to achieve 100% blocking rates throughout this test.*
- ***Test 4.2.2 -*** *Max 5,000 new connections per second - average packet size 540 bytes - maximum 225,000 packets per second. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With average connection rates average packet sizes, this is a good approximation of a real-world production network, and we expect all sensors to perform well in this test.*
- ***Test 4.2.3 -*** *Max 10,000 new connections per second - average packet size 440 bytes - maximum 275,000 packets per second. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With average packet sizes coupled with very high connection rates, this is a strenuous test for any sensor, and represents a very heavily used production network.*
- ***Test 4.2.4 -*** *Max 20,000 new connections per second - average packet size 360 bytes - maximum 320,000 packets per second. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With small packet sizes and extremely high connection rates this is an extreme test for any sensor. Not many sensors will perform well at all levels of this test.*

Test 4.3 - HTTP “Maximum Stress” Traffic With Transaction Delays

This test is identical to Test 4.2 except that we introduce 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 utilise additional resources to track those connections.

- ***Test 4.3.1 -*** *Max 5,000 new connections per second - average packet size 540 bytes - maximum 225,000 packets per second - 10 second transaction delay - maximum 50,000 open connections. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With average connection rates average packet sizes, this is a good approximation of a real-world production network, and we expect all sensors to perform well in this test.*
- ***Test 4.3.2 -*** *Max 10,000 new connections per second - average packet size 440 bytes - maximum 275,000 packets per second - 10 second transaction delay - maximum 100,000 open connections. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With average packet sizes coupled with very high connection rates, this is a strenuous test for any sensor, and represents a very heavily used production network.*

Test 4.4 - Protocol Mix Traffic

Whereas 4.2 and 4.3 provide a pure HTTP environment with varying connection rates and average packet sizes, the aim of this test is to simulate more of a “real world” environment by introducing additional protocols whilst 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, whilst less stressful than previous tests, is closer to what may be found on a heavily-utilised “normal” production network.

- ***Test 4.4.1 -*** *72% HTTP traffic (540 byte packets) + 20% FTP traffic + 6% UDP traffic (256 byte packets). Max 4000 new connections per second - average packet size 540 bytes - maximum 215,000 packets per second - maximum 750 open connections. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With lower connection rates, average packets sizes and a common protocol mix, this is a good approximation of a heavily-used production network, and we expect all sensors to perform well throughout this test.*

Test 4.5 - “Real World” Traffic

This is as close as it is possible to come to a true “real world” environment under lab conditions. For this test we eliminate the Reflector device and substitute an IIS Web server installed on a dual-Xeon server with Gigabit interface and 4GB RAM. This server holds a copy of The NSS Group Web site, and is capable of handling a full 1Gbps of traffic. We then capture a typical client browsing session on the NSS Group Web site, accessing a mixture of menu pages, lengthy text-based reports and multiple graphical images (screen shots) and have Avalanche replay multiple identical sessions from up to **20 new users per second**.

It should be noted that whereas the goal of the previous tests is a very predictable, consistent and repeatable background load that never varies, the nature of this test means that traffic is slightly more “bursty” in nature.

- ***Test 4.5.1 - Pure HTTP Traffic (simulated browsing session on NSS Web site):*** *Max 4700 new connections per second - 20 new users per second - average packet size 560 bytes - maximum 210,000 packets per second.*

*Repeated with 250Mbps, 500Mbps, 750Mbps and 950Mbps of background traffic. With genuine server responses to genuine **browser** sessions consisting **of multiple transactions per** session, this is a typical "real world" background load, albeit pure HTTP. Although the Web server and the network are extremely busy at the higher traffic loads, the "normal" connection rates and packet sizes should enable most sensors to perform well at all load levels in this test.*

- ***Test 4.5.2 - Protocol Mix (72% HTTP traffic (simulated browsing sessions as 4.5.1)) + 20% FTP traffic + 6% UDP traffic (256 byte packets)):** Max 3700 new connections per second - average packet size 560 bytes - maximum 205,000 packets per second - maximum 1,500 open connections.*

  *Repeated with 250Mbps, 500Mbps, 750Mbps and 950Mbps of background traffic. With genuine server responses to genuine browser sessions consisting of multiple **transactions** per session, mixed with FTP and UDP traffic, this is a typical "real world" background load. Although the Web server and the network are extremely busy at the higher traffic loads, the "normal" connection rates and packet sizes should enable most sensors to perform well at all load levels in this test.*

To gauge the effects of varying (smaller) packet sizes, connection rates and transaction delays, the results of tests 4.2 - 4.4 should be examined.

Section 5 – Latency & User Response Times

The aim of this section 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. We also determine the effect of high levels of normal HTTP traffic and a basic DOS attack on the average latency and user response times.

Test 5.1 - Latency

We use Spirent SmartFlow software and The Smartbits SMB6000 with Gigabit TeraMetrics cards to create multiple traffic flows through the appliance and measure the basic throughput, packet loss, and latency through the sensor. This test - whilst 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.

SmartFlow runs through several iterations of the test varying the traffic load from 250Mbps to 1Gbps bi-directionally (or up to the maximum rated throughput of the device should this be less than 1Gbps) in steps of 250Mbps. This is repeated for a range of packet sizes (256 bytes, 550 bytes and 1000 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.

- ***Test 5.1.1 - Latency With No Background Traffic:*** *SmartFlow traffic is passed across the infrastructure switches and through the device (the latency of the basic infrastructure is known and is constant throughout the tests). The packet loss and average latency are recorded at each packet size and each load level from 250Mbps to 1Gbps (in 250Mbps steps).*
- ***Test 5.1.2 - Latency With Background Traffic Load:*** *The Avalanche and Reflector are configured to generate a fixed amount of background HTTP traffic through the sensor (up to 50 per cent of the maximum rated bandwidth of the device under test - maximum 500Mbps - maximum 2,500 new connections per second - average packet size 540 bytes - maximum 112,500 packets per second).*

  A 250Mbps bi-directional load of SmartFlow traffic at various packet sizes (256 bytes, 540 bytes and 1000 bytes) is then passed across the infrastructure switches and through the device and the packet loss and average latency are recorded.
- ***Test 5.1.3 - Latency When Under Attack:*** *The Spirent WebSuite software is used to generate a fixed load of DOS/DDOS traffic of 10 per cent of the maximum rated bandwidth of the device under test (maximum 100Mbps). A 250Mbps bi-directional load of SmartFlow traffic at various packet sizes (256 bytes, 540 bytes and 1000 bytes) is then passed across the infrastructure switches and through the device and the packet loss and average latency are recorded. The device should be configured to detect/block/mitigate the DOS attack by the most efficient method available.*

Test 5.2 - User Response Times

Avalanche and Reflector devices are used to generate HTTP sessions through the device in order to gauge how any increases in latency will impact the user experience in terms of failed connections and increased Web response times.

- ***Test 5.2.1 - Web Response With No Background Traffic:*** *The Avalanche and Reflector are configured to generate HTTP traffic through the sensor (up to 50 per cent of the maximum rated bandwidth of the device under test - maximum 500Mbps - maximum 2,500 new connections per second - average packet size 540 bytes - maximum 112,500 packets per second).*

  The minimum, maximum and average page response times and number of failed connections are recorded by Avalanche to provide an indication of the expected response times under normal traffic conditions.
- ***Test 5.2.2 - Web Response When Under Attack:*** *The Avalanche and Reflector are configured to generate HTTP traffic through the sensor as for Test 5.2.1. The Spirent WebSuite software is then used to generate DOS/DDOS traffic up to 10 per cent of the maximum rated bandwidth of the device under test (maximum 100Mbps).*

  The minimum, maximum and average page response times and number of failed connections are recorded by Avalanche to provide an indication of the expected response times when the device is under attack.

Section 6 – Stability & Reliability

These tests attempt to verify the stability of the device under test under various extreme conditions. Long term stability is particularly important for an in-line IPS device, where failure can produce network outages.

- ***Test 6.1.1 - Blocking Under Extended Attack:*** *For this test, we expose the external interface of the device to a constant stream of alerts over an extended period of time. The device is configured to block and alert, and thus this test provides an indication the effectiveness of both the blocking and alert handling mechanisms. A continuous stream of exploits mixed with some legitimate sessions is transmitted through the device at a maximum of 100Mbps (max 50,000 packets per second, average packet sizes in the range of 120-350 bytes) for 8 hours with no additional background traffic. This is not intended as a stress test in terms of traffic load - merely a reliability test in terms of consistency of blocking performance.*

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

- ***Test 6.1.2 - Passing Legitimate Traffic Under Extended Attack:*** *This test is identical to 6.1.1, where we expose the external interface of the device to a constant stream of alerts over an extended period of time. The device is expected to remain operational and stable throughout this test, and to pass 100 per cent of legitimate traffic. Results are presented as a simple PASS/FAIL. If any legitimate traffic is blocked - caused by either the volume of traffic* ***or*** *the sensor failing closed for any reason - this will result in a FAIL.*

- ***Test 6.1.3 - ISIC/ESIC/TCPSIC/UDPSIC/ICMPSIC:*** *This test attempts to stress the protocol stack of the device under test by exposing it to traffic from the ISIC test tool. The ISIC test tool host is connected directly to the external interface of the sensor, and the ISIC target directly to the internal interface. ISIC traffic is transmitted through the sensor (without passing through any other network equipment) and the effects noted. 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 to attain a PASS.*

Section 7 – Management and Configuration

The aim of this section is to determine the features of the management system, together with the ability of the management port on the device under test to resist attack.

Test 7.1 - Management Port

Clearly the ability to manage the alert data collected by the sensor is a critical part of any IDS/IPS 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.

- ***Test 7.1.1 - Open ports:*** *We will scan the open ports and active services on the management interface and report on known vulnerabilities.*
- ***Test 7.1.2 - ISIC/ESIC/TCPSIC/UDPSIC/ICMPSIC:*** *This test attempts to stress the protocol stack of the management interface of the device under test by exposing it to traffic from the ISIC test tool. The ISIC test tool host is connected directly to the management interface of the IPS sensor, and that interface is also the target. ISIC traffic is transmitted to the management interface of the IPS device (without passing through any other network equipment) and the effects noted.*

  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 to attain a PASS.
- ***Test 7.1.3 -*** *We note whether the ISIC attacks themselves are detected by the sensor even though targeted at the management port.*

Top Layer IPS 5500 V3.3 Test Results (Content-based)

Section 1 - Detection Engine

Test 1.1 – Attack Recognition	Attacks	Default ARRD	Default ARRB	Custom ARRD	Custom ARRB
Test 1.1.1 - Backdoors	7	6	6	7	7
Test 1.1.2 - WINS/DNS	3	2	2	3	3
Test 1.1.3 - DOS	10	8	8	9	9
Test 1.1.4 - False negatives (modified exploits)	14	11	11	11	11
Test 1.1.5 - Finger	4	4	4	4	4
Test 1.1.6 - FTP	5	5	5	5	5
Test 1.1.7 - HTTP	43	38	38	40	40
Test 1.1.8 - ICMP	2	2	2	2	2
Test 1.1.9 - Reconnaissance	8	8	8	8	8
Test 1.1.10 - RPC	9	8	8	9	9
Test 1.1.11 - SSH	1	1	1	1	1
Test 1.1.12 - Telnet	1	0	0	1	1
Test 1.1.13 - Database	1	1	1	1	1
Test 1.1.14 - Mail	1	0	0	1	1
Test 1.1.15 - Voice	1	0	0	1	1
Total	**110**	**94 / 110**	**94 / 110**	**103 / 110**	**103 / 110**
		85%	**85%**	**94%**	**94%**

Test 1.2 – Resistance to False Positives	Pass/Fail
Test 1.2.1 - Suspicious FTP traffic	PASS
Test 1.2.2 - HTTP “exploit” using incorrect method	PASS
Test 1.2.3 - Retrieval of Web page containing “suspicious” URLs	PASS
Test 1.2.4 - Simple SMTP QUIT command	PASS
Test 1.2.5 - Normal NetBIOS copy of “suspicious” files	PASS
Test 1.2.6 - Normal NetBIOS traffic	PASS
Test 1.2.7 - POP3 e-mail containing “suspicious” URLs	PASS
Test 1.2.8 - POP3 e-mail with “suspicious” DLL attachment	PASS
Test 1.2.9 - POP3 e-mail with “suspicious” Web page attachment	PASS
Test 1.2.10 - SMTP e-mail transfer containing “suspicious” URLs	PASS
Test 1.2.11 - SMTP e-mail transfer with “suspicious” DLL attachment	PASS
Test 1.2.12 - SMTP e-mail transfer with “suspicious” Web page attachment	PASS
Test 1.2.13 - SNMP V3 packet with invalid parameter	PASS
Test 1.2.14 - Fake DNS /bin/sh buffer overflow	PASS
Test 1.2.15 - Inter-firewall communication traffic	PASS
Test 1.2.16 - Fake SQL Slammer traffic	PASS
Test 1.2.17 - File copy of GIF file (contains bytes which look like NOP sled)	PASS
Total Passed	**17 / 17**

Section 2 - IPS Evasion

Test 2.1 – Evasion Baselines	Detected?	Blocked?
Test 2.1.1 - NSS Back Orifice ping	YES	YES
Test 2.1.2 - Back Orifice connection	YES	YES
Test 2.1.3 - FTP CWD root	YES	YES
Test 2.1.4 - ISAPI printer overflow	YES	YES
Test 2.1.5 - Showmount export lists	YES	YES
Test 2.1.6 - Test CGI probe	YES	YES
Test 2.1.7 - PHF remote command execution	YES	YES
Total	**7 / 7**	**7 / 7**

Test 2.2 – Packet Fragmentation/Stream Segmentation	Detected?	Decoded?	Blocked?
Test 2.2.1 - IP fragmentation - ordered 8 byte fragments	YES	YES	YES
Test 2.2.2 - IP fragmentation - ordered 24 byte fragments	YES	YES	YES
Test 2.2.3 - IP fragmentation - out of order 8 byte fragments	YES	YES	YES
Test 2.2.4 - IP fragmentation - ordered 8 byte fragments, duplicate last packet	YES	YES	YES
Test 2.2.5 - IP fragmentation - out of order 8 byte fragments, duplicate last packet	YES	YES	YES
Test 2.2.6 - IP fragmentation - ordered 8 byte fragments, reorder fragments in reverse	YES	YES	YES
Test 2.2.7 - IP fragmentation - ordered 16 byte fragments, fragment overlap (favour new)	YES	YES	YES
Test 2.2.8 - IP fragmentation - ordered 16 byte fragments, fragment overlap (favour old)	YES	YES	YES
Test 2.2.9 - TCP segmentation - ordered 1 byte segments, interleaved duplicate segments with invalid TCP checksums	YES	NO	YES
Test 2.2.10 - TCP segmentation - ordered 1 byte segments, interleaved duplicate segments with null TCP control flags	YES	NO	YES
Test 2.2.11 - TCP segmentation - ordered 1 byte segments, interleaved duplicate segments with requests to resync sequence nos. mid-stream	YES	NO	YES
Test 2.2.12 - TCP segmentation - ordered 1 byte segments, duplicate last packet	YES	NO	YES
Test 2.2.13 - TCP segmentation - ordered 2 byte segments, segment overlap (favour new)	YES	NO	YES[1]
Test 2.2.14 - TCP segmentation - ordered 1 byte segments, interleaved duplicate segments with out-of-window sequence numbers	YES	NO	YES
Test 2.2.15 - TCP segmentation - out of order 1 byte segments	YES	NO	YES
Test 2.2.16 - TCP segmentation - out of order 1 byte segments, interleaved duplicate segments with faked retransmits	YES	NO	YES
Test 2.2.17 - TCP segmentation - ordered 1 byte segments, segment overlap (favour new)	YES	NO	YES[1]
Test 2.2.18 - TCP segmentation - out of order 1 byte segments, PAWS elimination (interleaved dup segments with older TCP timestamp options)	YES	NO	YES[1]
Test 2.2.19 - IP fragmentation - out of order 8 byte fragments, interleaved duplicate packets scheduled for later delivery	YES	NO	YES
Test 2.2.20 - TCP segmentation - ordered 16 byte segments, segment overlap (favour new (Unix))	YES	NO	YES[1]
Total	**20 / 20**	**8 / 20**	**20 / 20**

Test 2.3 – URL Obfuscation	Detected?	Decoded?	Blocked?
Test 2.3.1 - URL encoding	YES	NO	YES
Test 2.3.2 - /./ directory insertion	YES	YES	YES
Test 2.3.3 - Premature URL ending	YES	YES	YES
Test 2.3.4 - Long URL	YES	NO	YES
Test 2.3.5 - Fake parameter	YES	NO	YES
Test 2.3.6 - TAB separation	YES	YES	YES
Test 2.3.7 - Case sensitivity	YES	YES	YES
Test 2.3.8 - Windows \ delimiter	YES	YES	YES
Test 2.3.9 - Session splicing	YES	NO	YES
Total	**9 / 9**	**5 / 9**	**9 / 9**

Test 2.4 – Miscellaneous Obfuscation Techniques	Detected?	Decoded?	Blocked?
Test 2.4.1 - Altering default ports	NO	NO	NO
Test 2.4.2 - Inserting spaces in FTP command lines	YES	YES	YES
Test 2.4.3 - Inserting non-text Telnet opcodes in FTP data stream	YES	YES	YES
Test 2.4.4 - Polymorphic mutation (ADMmutate)	YES	YES	YES
Test 2.4.5 - Altering protocol and RPC PROC numbers	YES	YES	YES
Test 2.4.6 - RPC record fragging (MS-RPC and Sun)	NO[2]	NO[2]	NO[2]
Test 2.4.7 - HTTP exploits to port <> 80	YES[3]	YES[3]	YES[3]
Total	**5 / 7**	**5 / 7**	**5 / 7**

Section 3 - Stateful Operation

Test 3.1 – Stateless Attack Replay	Alert?	Blocked?	Pass/Fail
Test 3.1.1 - Stateless Web exploits	NO	YES[4]	PASS
Test 3.1.2 - Stateless FTP exploits	NO	YES[4]	PASS

Test 3.2 – Simultaneous Open Connections (default settings)							
Number of open connections	10,000	25,000	50,000	100,000	250,000	500,000	1,000,000
Test 3.2.1 - Attack Detection	PASS	PASS	PASS	PASS	PASS	PASS	PASS
Test 3.2.2 - Attack Blocking	PASS	PASS	PASS	PASS	PASS	PASS	PASS
Test 3.2.3 - State Preservation	PASS	PASS	PASS	PASS	PASS	PASS	PASS
Test 3.2.4 - Legitimate traffic blocking	PASS	PASS	PASS	PASS	PASS	PASS	PASS

Test 3.3 – Simultaneous Open Connections (after tuning)[5]							
Number of open connections	10,000	25,000	50,000	100,000	250,000	500,000	1,000,000
Test 3.3.1 - Attack Detection	PASS	PASS	PASS	PASS	PASS	PASS	PASS
Test 3.3.2 - Attack Blocking	PASS	PASS	PASS	PASS	PASS	PASS	PASS
Test 3.3.3 - State Preservation	PASS	PASS	PASS	PASS	PASS	PASS	PASS
Test 3.3.4 - Legitimate traffic blocking	PASS	PASS	PASS	PASS	PASS	PASS	PASS

Section 4 - Detection/Blocking Performance Under Load

Test 4.1 – UDP traffic to random valid ports		250Mbps	500Mbps	750Mbps	1Gbps	Max
Test 4.1.1 - 256 byte packet test - max 453,000pps	Detected	100%	100%	100%	100%	1Gbps
	Blocked	100%	100%	100%	100%	
Test 4.1.2 - 550 byte packet test - max 220,000pps	Detected	100%	100%	100%	100%	1Gbps
	Blocked	100%	100%	100%	100%	
Test 4.1.3 - 1514 byte packet test - max 122,000pps	Detected	100%	100%	100%	100%	1Gbps
	Blocked	100%	100%	100%	100%	

Test 4.2 – HTTP "maximum stress" traffic with no transaction delays		250Mbps	500Mbps	750Mbps	1Gbps	Max
Test 4.2.1 - Max 2500 connections per second - ave packet size 1000 bytes - max 120,000 packets per second	Detected	100%	100%	100%	100%	1Gbps
	Blocked	100%	100%	100%	100%	
Test 4.2.2 - Max 5000 connections per second - ave packet size 540 bytes - max 225,000 packets per second	Detected	100%	100%	100%	100%	1Gbps
	Blocked	100%	100%	100%	100%	
Test 4.2.3 - Max 10000 connections per second - ave packet size 440 bytes - max 275,000 packets per second	Detected	100%	100%	100%	100%	1Gbps
	Blocked	100%	100%	100%	100%	
Test 4.2.4 - Max 20000 connections per second - ave packet size 360 bytes - max 320,000 packets per second	Detected	100%	100%	100%	100%	1Gbps
	Blocked	100%	100%	100%	100%	

Test 4.3 – HTTP "maximum stress" traffic with transaction delays		250Mbps	500Mbps	750Mbps	1Gbps	Max
Test 4.3.1 - Max 5000 connections per second - ave packet size 540 bytes - max 225,000 packets per second - 10 sec delay - max 50,000 open connections	Detected	100%	100%	100%	100%	1Gbps
	Blocked	100%	100%	100%	100%	
Test 4.3.2 - Max 10000 connections per second - ave packet size 440 bytes - max 275,000 packets per second - 10 sec delay - max 100,000 open connections	Detected	100%	100%	100%	100%	1Gbps
	Blocked	100%	100%	100%	100%	

Test 4.4 – Protocol mix		250Mbps	500Mbps	750Mbps	1Gbps	Max
Test 4.4.1 - 72% HTTP (540 byte packets) + 20% FTP + 6% UDP (256 byte packets). Max 4000 connections per second - ave packet size 540 bytes - max 215,000 packets per second - max 750 open connections	Detected	100%	100%	100%	100%	1Gbps
	Blocked	100%	100%	100%	100%	

Test 4.5 – Real World traffic		250Mbps	500Mbps	750Mbps	1Gbps	Max
Test 4.5.1 - Pure HTTP (simulated browsing session on NSS Web site). Max 4700 connections per second - 20 new users per second - ave packet size 560 bytes - max 210,000 packets per second	Detected	100%	100%	100%	100%	1Gbps
	Blocked	100%	100%	100%	100%	
Test 4.5.2 - Protocol mix - 72% HTTP (simulated browsing sessions as 2.5.1) + 20% FTP + 6% UDP (256 byte packets). Max 3700 connections per second - ave packet size 560 bytes - max 205,000 packets per second - max 1,500 open connections	Detected	100%	100%	100%	100%	1Gbps
	Blocked	100%	100%	100%	100%	

Section 5 - Latency & User Response Times

Test 5.1 – Latency	Packet Size	250Mbps	500Mbps	750Mbps	1Gbps
Test 5.1.1 Average latency (µs) with no background traffic	256	19.83	20.04	20.27	20.89
	550	27.29	27.65	27.64	27.93
	1000	40.11	40.15	40.46	40.63
Test 5.1.2 Average latency (µs) with background traffic (500Mbps HTTP traffic, max 2500 connections per second - ave packet size 540 bytes - max 112,500 packets per second)	256	44.39			
	550	50.43			
	1000	62.28			
Test 5.1.3 Average latency (µs) when under attack (100Mbps SYN flood)	256	21.52			
	550	28.81			
	1000	41.47			

Test 5.2 – User Response Times	Attempted Trans	Failed Trans	Min Page Response	Max Page Response	Ave Page Response
Test 5.2.1 - Web page response (ms) with no background traffic (500Mbps HTTP traffic, max 2500 connections per sec - ave packet size 540 bytes - max 112,500 packets per sec)	1543997	0	200	215	205
Test 5.2.2 - Web page response (ms) when under attack (500Mbps HTTP traffic, max 2500 connections per sec - ave packet size 540 bytes - max 112,500 packets per sec PLUS 100Mbps SYN flood)	1540730	0	203	1709	205

Section 6 - Stability & Reliability

Test ID	Result
Test 6.1.1 - Blocking Under Extended Attack	100%
Test 6.1.2 - Passing legitimate traffic under extended attack	100%
Test 6.1.3 - ISIC/ESIC/TCPSIC/UDPSIC/ICMPSIC	PASS

Section 7 - Management Interface

Test ID	Result
Test 7.1.1 - Open Ports	PASS
Test 7.1.2 - ISIC/ESIC/TCPSIC/UDPSIC/ICMPSIC	PASS
Test 7.1.3 - ISIC attacks detected against management interface?	NO

Notes:

1. Always blocked by firewall, regardless of configuration
2. No Sun RPC anti-evasion – MS-RPC only
3. Administrator can configure multiple Web server ports
4. It is possible to configure the device to allow mid-flows
5. No tuning required for Test 3.3 – the 1 million connection limit is fixed out of the box

Section 1: Detection Engine

We installed one sensor with the latest updates, and used the "*Very Strict*" policy, which has almost every signature enabled (barring approximately eight which are considered audit only, and too noisy for most environments – these are disabled by default in this policy).

The response settings were set to "block" and "log", and raw packet contents are logged automatically. Default settings were used for SYN Floods, but some adjustment was made to the ICMP rates, UDP rates, and maximum number of connections allowed per client to detect other rate-based attacks.

Top Layer has made significant additions to its protocol decode and validation modules, and significant enhancements to its signature set for this release. Signature recognition (with blocking disabled) was good out of the box (85 per cent), and was increased to a creditable 94 per cent after the application of a signature pack update which was provided to us in 48 hours. Blocking performance was identical throughout the tests.

We noted a minimum of "noise", with very few test cases raising multiple alerts for a single exploit. Performance in our "false negative" tests was reasonable out of the box, but did not improve following the signature update.

A major concern in deploying an IPS is the blocking of legitimate traffic. Providing the signatures which are disabled by default in the standard policies are left that way, the IPS 5500 resistance to false positives is good.

However, it should be noted that when the various attack mitigation features are employed, careful tuning of the mitigation parameters is required in order that legitimate traffic is not blocked accidentally. There is no automatic "learning" capability, meaning the responsibility for determining the optimum mitigation thresholds lies squarely with the administrator.

The IPS 5500 appliance now arrives with four default policies: *Detect Only*, *Recommended*, *Strict* and *Very Strict*, each imposing different combinations of detection and blocking. The *Recommended* policy will suit most organisations, since it disables all signatures that are not considered "*likely exploits*" (i.e. audits/information signatures) and all those where the confidence setting is "*low*" (i.e. possibly susceptible to false positives).

Section 2: IPS Evasion

Resistance to known evasion techniques was excellent, with the IPS 5500 achieving a clean sweep across the board in most of our evasion tests. *Fragroute* and *Whisker* both failed to trick the device into ignoring valid attacks. Not only were the fragmented and obfuscated attacks blocked successfully, but many of them were decoded accurately as well.

Of the miscellaneous evasion techniques, changing ports on Trojan programs and using RPC fragmentation with Sun RPC exploits both proved troublesome (Top Layer has implemented an MS RPC Protocol Validation Module, but does not yet have one for Sun RPC).

Section 3: Stateful Operation

Out of the box, the IPS 5500 is designed to handle 1,000,000 (one million) open connections without tuning. It was thus able to handle our 1 million open connection test with ease.

Default operation of the device is to age out old connections in order to accept new ones when the state tables are full or resources are low. This behaviour is configurable, a feature we would like to see in all IPS products since there is no real right or wrong way to handle this situation.

Stateless “exploits” are not alerted upon (this is correct behaviour in order to be resistant to *Stick* and *Snot* tools) and mid-flows are blocked by default. It is, however, possible to configure the device to allow mid-flows, and there is a configurable “grace period” where they are not enforced following a power-cycle to prevent blocking of legitimate traffic should the device come on-line in mid session.

Section 4: Detection/Blocking Performance Under Load

The IPS 5500 is rated by the Top Layer for a Gigabit link (2Gbps aggregate) and was tested to 1Gbps here.

It turned in an outstanding performance in all our tests, achieving 100 per cent detection rates across the board, and clearly with some headroom to spare.

We would be more than happy to rate this device at a minimum of 1Gbps under **all** network loads.

Section 5: Latency & User Response Times

The IPS 5500’s basic latency figures were outstanding - almost switch like - across the board under normal traffic loads. They ranged from 20µs with 250Mbps of 256 byte packets, to 41µs with 1Gbps of 1000 byte packets.

Behaviour throughout the tests with no background traffic was extremely consistent and predictable, hardly increasing at all as additional network load was applied from 250Mbps to 1Gbps. The IPS 5500 was also one of the few devices we have tested in our labs which has achieved zero packet loss and low latency at **all** packet sizes (including 64 bytes) up to 1Gbps. The latency with 64 byte packets at 1Gbps was just 17µs (which also includes the basic latency of the test infrastructure).

Placing the device under a half load of 500Mbps of HTTP traffic, we noted increases in latency, with 256 byte packets rising from 20µs to 44µs, 550 byte packets rising from 27µs to 50µs, and 1000 byte packets rising from 40µs to 62µs.

100Mbps of SYN flood traffic had a negligible effect on the IPS 5500, increasing the base latencies at all packet sizes by around a microsecond only.

The SYN flood was mitigated completely once it had been detected (which did not take long).

Note that this is not the case with all DOS/DDOS attacks, however. With many other types of attack, the IPS 5500 does “mitigate” (i.e. reduce) the effects of the attack rather than block it completely.

HTTP response times were also excellent, and although the standard deviation was greater when under attack (i.e. the **maximum** response time increased somewhat) the **average** response times were largely unaffected by the 100Mbps of SYN flood traffic.

Overall, latency figures were considered to be outstanding for a device of this type under all load conditions and packet sizes. Clearly this device can be placed anywhere on the corporate network - from the perimeter to a heavily-loaded high-speed backbone - without impacting overall network performance in any way.

Section 6: Stability & Reliability

The IPS 5500 performed consistently and completely reliably throughout our tests. Under eight hours of extended attack (comprising millions of exploits mixed with genuine traffic) it continued to block 100 per cent of attack traffic, whilst passing 100 per cent of legitimate traffic.

Exposing the sensor interface to ISIC-generated traffic had no adverse effect, and the device continued to detect and block all other exploits throughout and following the ISIC attack.

A large number of ISIC-related alerts were raised during the attack and the attack was mitigated partially at first – eventually, the device recognised it as DDOS attack (ISIC spoofs a wide range of source addresses) and provided almost total mitigation.

There were no residual stability problems once the attack had been terminated.

Section 7: Management Interface

Open ports on the management interface are restricted to HTTP, HTTPS, NTP and SNMP. HTTP can be disabled once HTTPS has been configured.

The extended ISIC attack against the management interface had virtually no effect on the appliance and its ability to detect and block attacks, though there was a slight delay in sensor to console communications throughout. No alerts were raised during the attack.

The sensor continued to detect and block malicious traffic whilst passing legitimate traffic throughout and following the ISIC attack, and there were no residual stability problems.

APPENDIX B – TEST RESULTS (RATE-BASED)

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 network is 100/1000Mbit Ethernet with CAT 5e cabling and Cisco 6500-Series switches (these have a mix of fibre and copper Gigabit interfaces).

All devices are expected to be provided as appliances - if software-only, the supplier pre-installs the software on the recommended hardware platform. The sensor is configured as a perimeter device during testing (i.e. as if installed in front of the main Internet gateway/firewall). There is no firewall protecting the target subnet.

Traffic generation equipment - such as the machines generating exploits, Spirent Avalanche and Spirent Smartbits *transmit* port - is connected to the “external” network, whilst the “receiving” equipment - such as the “target” hosts for the exploits, Spirent Reflector and Spirent Smartbits *receive* port - is connected to the internal network. The device under test 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 device under test, from external to internal.

The same traffic is mirrored to a single SPAN port of the external gateway switch, to which an Adtech network monitoring device is connected. The Adtech AX/4000 monitors the same mirrored traffic to ensure that the total amount of traffic never exceeds 1Gbps (which would invalidate the test run).

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.

Section 1 – Detection Engine

The aim of this section is to verify that the sensor is capable of detecting and blocking a wide range of common rate-based exploits accurately, whilst 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).

Whilst the various replay tools are generally useful for testing signature based IPS/IDS systems, their use in conjunction with rate-based attack mitigators is unpredictable, and thus the use of live tools is preferred in this case.

Test 1.1 - Attack Detection/mitigation

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

The sensor is installed and all possible detection modes are activated. The vendor is permitted to tune the product (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.

Our attack suite covers the following areas:

- ***Test 1.1.1 -*** *SYN Flood*
- ***Test 1.1.2 -*** *TCP SYN Attack (low-rate SYN Flood)*
- ***Test 1.1.3 -*** *ICMP Flood*
- ***Test 1.1.4 -*** *Distributed Denial Of Service (DDOS)attack*
- ***Test 1.1.5 -*** *UDP Flood*
- ***Test 1.1.6 -*** *IGMP Flood*
- ***Test 1.1.7 -*** *Connection Flood (fast)*
- ***Test 1.1.8 -*** *Connection Flood (slow)*
- ***Test 1.1.9 -*** *Random protocol violations (invalid packets)*
- ***Test 1.1.10 -*** *Trojan response (external host attempts connection to internal Trojan and receives response)*
- ***Test 1.1.11 -*** *ICMP Sweep (inbound)*
- ***Test 1.1.12 -*** *ICMP Sweep (outbound)*
- ***Test 1.1.13 -*** *SQL Slammer*
- ***Test 1.1.14 -*** *Spoofed IP attack*
- ***Test 1.1.15 -*** *Web vulnerability scan*
- ***Test 1.1.16 -*** *Port Scan (full TCP connect)*
- ***Test 1.1.17 -*** *Stealth Port Scan*
- ***Test 1.1.18 -*** *FIN Port Scan*
- ***Test 1.1.19 -*** *UDP Port Scan*
- ***Test 1.1.20 -*** *Null Port Scan*
- ***Test 1.1.21 -*** *Xmas Port Scan*
- ***Test 1.1.22 -*** *IP Protocol Port Scan*
- ***Test 1.1.23 -*** *ACK Port Scan*
- ***Test 1.1.24 -*** *Window Port Scan*

We expect all the attacks to be reported in as straightforward and clear a manner as possible, and alerts to be raised in a timely manner.

It is necessary to recognise that different devices detect and mitigate rate-based attacks in different ways. For example, where SYN proxies are utilised, 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.

Test 1.2 - High Volume Attack Detection/Mitigation

Whereas the previous tests determine the device's ability to detect and mitigate a wide range of attacks under normal conditions (using the live attack tools), the level of flooding is generally fairly low.

This test generates a subset of the previous attacks at very high volumes (up to 80 per cent of the rated bandwidth) to determine if the rate of attack has any effect on the device's ability to detect and mitigate it.

The following attacks are repeated at various levels (10%, 20%, 40% and 80% of the rated bandwidth of the device under test) and we test for successful mitigation (as defined in Test 1.1) in each case.

- ***Test 1.2.1 -*** *SYN Flood (DOS from single source IP)*
- ***Test 1.2.2 -*** *SYN Flood (DDOS from multiple source IPs)*
- ***Test 1.2.3 -*** *Smurf*
- ***Test 1.2.4 -*** *Teardrop*
- ***Test 1.3.5 -*** *ICMP Flood*
- ***Test 1.2.6 -*** *UDP Flood*

Test 1.3 - Resistance To False Positives

The aim of this test is to demonstrate how likely it is that a sensor raises a false positive alert - particularly critical for in-line devices.

Throughout the test we load the network with a wide range of "normal" network traffic. We note how many - if any - false alarms are raised on this traffic once the device has been tuned/configured, and comment an what action is necessary to reduce or eliminate such false positive scenarios.

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.

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.

- ***Test 1.2.1 -*** *False positives*

Section 2 – Evasion

The aim of this section is to verify that the sensor is capable of detecting and mitigating basic attacks when subjected to varying common evasion techniques.

Test 2.1 - Baselines

The aim of this test is to establish that the sensor is capable of detecting and blocking a number of common basic attacks (our baseline suite) in their normal state, with no evasion techniques applied.

- ***Test 2.1.1 -*** *Baseline attack replay*

Test 2.2 - Fragmentation and Timing

The SYN Stealth Port Scan is repeated, subjecting the already small packets to fragmentation and delaying the amount of time between packets in order to evade detection:

- ***Test 2.2.1 -*** *Fragmented UDP Flood (Teardrop)*
- ***Test 2.2.2 -*** *Fragmented Stealth Port Scan*
- ***Test 2.2.3 -*** *Slow Stealth Port Scan (0.4 secs between packets)*
- ***Test 2.2.4 -*** *Very Slow Stealth Port Scan (15 secs between packets)*
- ***Test 2.2.5 -*** *Slow Connection Flood (1 second between packets)*
- ***Test 2.2.6 -*** *Very Slow Connection Flood (3 seconds between packets)*

For each of the evasion techniques, we note if (i) the attempted attack is blocked successfully (the primary aim of any attack mitigation device), and (ii) if the device is capable of providing an accurate alert relating to the original exploit, rather than alerting purely on anomalous traffic detected as a result of the evasion technique itself.

Test 2.3 - URL Obfuscation

The Web vulnerability scans are repeated (where applicable), this time applying various URL obfuscation techniques made popular by the Whisker Web server vulnerability scanner, including:

- ***Test 2.3.1 -*** *URL encoding*
- ***Test 2.3.2 -*** */./ directory insertion*
- ***Test 2.3.3 -*** *Premature URL ending*
- ***Test 2.3.4 -*** *Long URL*
- ***Test 2.3.5 -*** *Fake parameter*
- ***Test 2.3.6 -*** *TAB separation*
- ***Test 2.3.7 -*** *Case sensitivity*
- ***Test 2.3.8 -*** *Windows \ delimiter*
- ***Test 2.3.9 -*** *Session splicing*

For each of the evasion techniques, we note if (i) the attempted attack is blocked successfully (the primary aim of any attack mitigation device), and (ii) if the device is capable of providing an accurate alert relating to the original exploit, rather than alerting purely on anomalous traffic detected as a result of the evasion technique itself.

Section 3 – Attack Mitigation Performance Under Load

The aim of this section is to verify that the sensor is capable of detecting and blocking attacks when subjected to increasing loads of background traffic up to the maximum bandwidth supported as claimed by the vendor. In other words, we expect the device to be able to successfully mitigate attacks even when heavily loaded with normal traffic.

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. Our “attacker” hosts launch a number of attacks at target hosts on the subnet being protected by the device under test. The Adtech network monitor is configured to monitor the switch SPAN port consisting of normal, exploit and background traffic, and is capable of reporting the total level of attack and/or normal traffic seen on the wire as verification.

Having ensured that the sensor is capable of detecting our baseline attacks, increasing levels of varying types of background traffic are generated **through** the sensor.

All tests are repeated with 250Mbps, 500Mbps, 750Mbps and 1Gbps of background traffic (or up to the maximum rated throughput of the device should this be less than 1Gbps). At each level, we launch a selection of attacks from the external network and check to ensure that they are successfully detected and mitigated. We also check to ensure that the victim servers remain alive and responsive to legitimate requests from the external network throughout the tests.

At all stages, the Adtech network monitor verifies both the overall traffic loading and the total number of exploits seen on the target subnet. An additional confirmation is provided by the target host which reports the number of attack packets which actually made it through.

For each type of background traffic, we also determine the maximum load the sensor can sustain before it begins to drop packets/miss alerts.

Test 3.1 - UDP Traffic To Random Valid Ports

This test uses UDP packets of varying sizes generated by a **Smartbits SMB6000** with LAN-3301A 10/100/1000Mbps **TeraMetrics** cards installed. A constant stream of the appropriate mix of packets - with variable source IP addresses and ports transmitting to a single fixed IP address/port - is transmitted through the sensor (bi-directionally, maximum of 1Gbps).

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 are verified by the Adtech Gigabit network monitoring tool before each test begins. Multiple tests are run and averages taken where necessary.

This traffic does not attempt to simulate any form of “real world” network condition. The aim of this test is purely to determine the raw packet processing capability of the sensor, and its effectiveness at passing “useless” packets quickly in order to transfer potential attack packets to the detection engine. It is important that the device under test is tuned accurately to ensure that this traffic is not detected as a flooding attack and is thus mitigated in error.

The range of packet sizes has been selected to mirror the maximum, minimum and average packet sizes used in our HTTP stress tests.

At each level, we launch a selection of attacks from the external network and check to ensure that they are successfully detected and mitigated. We also check to ensure that the victim servers remain alive and responsive to legitimate requests from the external network throughout the tests. Both of these checks should yield a successful result in order for the device to attain a PASS at the given load level.

- ***Test 3.1.1 - 256 byte packets - maximum 453,000 packets per second:*** *This test is roughly equivalent to a 40,000 connections per second test in our HTTP stress tests (in terms of packet size and packets per second rate), and has been included to provide an indication of the packet processing performance under the most extreme conditions for most devices - it is unlikely that any real-life network will ever see network loads of over 450,000 256-byte packets per second unless under severe DOS conditions. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic.*
- ***Test 3.1.2 - 550 byte packets - maximum 220,000 packets per second:*** *This test has been included to provide a comparison with our "real world" packet mixes, since the average packet size is similar. No sessions are created during this test and there is very little for the detection engine to do in the way of protocol analysis.*

  This test provides a reasonable indication of the ability of a device to process packets from the wire on an "average" network, and we would expect all products to demonstrate good performance levels. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic.
- ***Test 3.1.3 - 1000 byte packets - maximum 122,000 packets per second:*** *This test is the complete opposite of the 256 byte packet test, in that we would expect every single product to be capable of achieving 100 per cent PASS rates across the board when using only 1000 byte packets.*

  We have included this test 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 (or larger) packet sizes. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic.*

Test 3.2 - HTTP "Maximum Stress" Traffic With No Transaction Delays

The use of multiple Spirent Communications **Avalanche 2500** and **Reflector 2500** devices allows us to create true "real world" traffic at speeds of up to 4.2 Gbps as a background load for our tests. Our Avalanche configuration is capable of simulating over 5 million users, with over 5 million concurrent sessions, and over 200,000 HTTP requests per second.

By creating genuine session-based traffic with varying session lengths, the sensor is forced to track valid 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, whilst ensuring absolute accuracy and repeatability.

The aim of this test is to stress the detection engine and determine how the sensor copes with detecting and mitigating attacks under network loads of varying average packet size and varying connections per second.

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.

At each level, we launch a selection of attacks from the external network and check to ensure that they are successfully detected and mitigated. We also check to ensure that the victim servers remain alive and responsive to legitimate requests from the external network throughout the tests. Both of these checks should yield a successful result in order for the device to attain a PASS at the given load level.

- ***Test 3.2.1 -*** *Max 2,500 new connections per second - average packet size 1000 bytes - maximum 120,000 packets per second. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With relatively low connection rates and large packet sizes, we expect all sensors to achieve 100 per cent PASS rates throughout this test.*
- ***Test 3.2.2 -*** *Max 5,000 new connections per second - average packet size 540 bytes - maximum 225,000 packets per second. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With average connection rates average packet sizes, this is a good approximation of a real-world production network, and we expect all sensors to perform well in this test.*
- ***Test 3.2.3 -*** *Max 10,000 new connections per second - average packet size 440 bytes - maximum 275,000 packets per second. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With average packet sizes coupled with very high connection rates, this is a strenuous test for any sensor, and represents a very heavily used production network.*
- ***Test 3.2.4 -*** *Max 20,000 new connections per second - average packet size 360 bytes - maximum 320,000 packets per second. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With small packet sizes and extremely high connection rates this is an extreme test for any sensor. Not many sensors will perform well at all levels of this test.*

Test 3.3 - HTTP “Maximum Stress” Traffic With Transaction Delays

This test is identical to Test 4.2 except that we introduce 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 utilise additional resources to track those connections.

- ***Test 3.3.1 -*** *Max 5,000 new connections per second - average packet size 540 bytes - maximum 225,000 packets per second - 10 second transaction delay - maximum 50,000 open connections.*

Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With average connection rates average packet sizes, this is a good approximation of a real-world production network, and we expect all sensors to perform well in this test.

- ***Test 3.3.2 -*** *Max 10,000 new connections per second - average packet size 440 bytes - maximum 275,000 packets per second - 10 second transaction delay - maximum 100,000 open connections. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With average packet sizes coupled with very high connection rates, this is a strenuous test for any sensor, and represents a very heavily used production network.*

Test 3.4 - Protocol Mix Traffic

Whereas 3.2 and 3.3 provide a pure HTTP environment with varying connection rates and average packet sizes, the aim of this test is to simulate more of a “real world” environment by introducing additional protocols whilst 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, whilst less stressful than previous tests, is closer to what may be found on a heavily-utilised “normal” production network.

- ***Test 3.4.1 -*** *72% HTTP traffic (540 byte packets) + 20% FTP traffic + 6% UDP traffic (256 byte packets). Max 4000 new connections per second - average packet size 540 bytes - maximum 215,000 packets per second - maximum 750 open connections. Repeated with 250Mbps, 500Mbps, 750Mbps and 1000Mbps of background traffic. With lower connection rates, average packets sizes and a common protocol mix, this is a good approximation of a heavily-used production network, and we expect all sensors to perform well throughout this test.*

Test 3.5 - “Real World” Traffic

This is as close as it is possible to come to a true “real world” environment under lab conditions. For this test we eliminate the Reflector device and substitute an IIS Web server installed on a dual Xeon server with Gigabit interface and 4GB RAM. This server holds a copy of The NSS Group Web site, and is capable of handling a full 1Gbps of traffic. We then capture a typical client browsing session on the NSS Group Web site, accessing a mixture of menu pages, lengthy text-based reports and multiple graphical images (screen shots) and have Avalanche replay multiple identical sessions from up to **20 new users per second**.

It should be noted that whereas the goal of the previous tests is a very predictable, consistent and repeatable background load that never varies, the nature of this test means that traffic is slightly more “bursty” in nature.

- ***Test 3.5.1 - Pure HTTP Traffic (simulated browsing session on NSS Web site):*** *Max 4700 new connections per second - 20 new users per second - average packet size 560 bytes - maximum 210,000 packets per second. Repeated with 250Mbps, 500Mbps, 750Mbps and 950Mbps of background traffic.*

*With genuine server responses to genuine **browser** sessions consisting **of multiple transactions per** session, this is a typical "real world" background load, albeit pure HTTP. Although the Web server and the network are extremely busy at the higher traffic loads, the "normal" connection rates and packet sizes should enable most sensors to perform well at all load levels in this test.*

- ***Test 3.5.2 - Protocol Mix (72% HTTP traffic (simulated browsing sessions as 3.5.1)) + 20% FTP traffic + 6% UDP traffic (256 byte packets)):** Max 3700 new connections per second - average packet size 560 bytes - maximum 205,000 packets per second - maximum 1,500 open connections. Repeated with 250Mbps, 500Mbps, 750Mbps and 950Mbps of background traffic.*

  *With genuine server responses to genuine browser sessions consisting of multiple **transactions** per session, mixed with FTP and UDP traffic, this is a typical "real world" background load. Although the Web server and the network are extremely busy at the higher traffic loads, the "normal" connection rates and packet sizes should enable most sensors to perform well at all load levels in this test.*

To gauge the effects of varying (smaller) packet sizes, connection rates and transaction delays, the results of tests 3.2 - 3.4 should be examined.

Test 3.6 - Maximum Open Connections

It is important that the device under test cannot only support a sufficiently high rate of connection set-up and tear-down, but that is can also support a sufficiently large number of simultaneous connections. This test determines its maximum capacity.

- ***Test 3.6.1 -** In this test the Spirent Avalanche is set to continually open HTTP connections with the Reflector. The Reflector is set to implement a delay on each transaction, thus ensuring that transactions remain open for a significant period of time (typical of a real world situation).*

  The Avalanche is monitored and the point where transactions begin to fail is noted. Once the exact point of failure has been determined, the test is run for several hours with that number of transactions held open throughout.

Section 4 – Latency & User Response Times

The aim of this section 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.

We also determine the effect of varying levels of normal HTTP traffic and varying levels of attack traffic on the average latency and user response times.

Test 4.1 - Latency

We use Spirent SmartFlow software and the Smartbits SMB6000 with Gigabit TeraMetrics cards to create multiple traffic flows through the appliance and measure the basic throughput, packet loss, and latency through the sensor. This test - whilst 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.

SmartFlow runs through several iterations of the test varying the traffic load from 250Mbps to 1Gbps bi-directionally (or up to the maximum rated throughput of the device should this be less than 1Gbps) in steps of 250Mbps. This is repeated for a range of packet sizes (256 bytes, 550 bytes and 1000 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.

- ***Test 4.1.1 - Latency With No Background Traffic:*** *SmartFlow traffic is passed across the infrastructure switches and through the device (the latency of the basic infrastructure is known and is constant throughout the tests). The packet loss and average latency are recorded at each packet size and each load level from 250Mbps to 1Gbps (in 250Mbps steps). Note that the* ***only*** *traffic passing through the device during this test is the UDP traffic used by SmartFlow to measure latency.*
- ***Test 4.1.2 - Latency With Background Traffic Load:*** *The Avalanche and Reflector are configured to generate varying loads of background HTTP traffic through the sensor (from 25 to 100 per cent of the maximum rated bandwidth of the device under test - maximum 1Gbps - 5,000 new connections per second - average packet size 540 bytes - 225,000 packets per second). A very small bi-directional load of SmartFlow traffic at various packet sizes (256 bytes, 540 bytes and 1000 bytes) is also passed across the infrastructure switches and through the device and the packet loss and average latency are recorded at each HTTP load level.*
- ***Test 4.1.3 - Latency When Under Attack:*** *The Spirent WebSuite software is used to generate varying loads of SYN flood traffic (from a single source IP) through the sensor (from 20 to 80 per cent of the maximum rated bandwidth of the device under test - maximum 800Mbps - 1,184,000 packets per second). A very small bi-directional load of SmartFlow traffic at various packet sizes (256 bytes, 540 bytes and 1000 bytes) is also passed across the infrastructure switches and through the device and the packet loss and average latency are recorded at each attack load level. The device should be configured to detect/mitigate the attack by the most efficient method available.*

Test 4.2 - User Response Times

Avalanche and Reflector devices are used to generate HTTP sessions through the device in order to gauge how any increases in latency will impact the user experience in terms of failed connections and increased Web response times.

- ***Test 4.2.1 - Web Response With No Background Traffic:*** *The Avalanche and Reflector are configured to generate HTTP traffic through the sensor (up to 50 per cent of the maximum rated bandwidth of the device under test - maximum 500Mbps - maximum 2,500 new connections per second - average packet size 540 bytes - maximum 112,500 packets per second). The minimum, maximum and average page response times and number of failed connections are recorded by Avalanche to provide an indication of the expected response times under normal traffic conditions.*
- ***Test 4.2.2 - Web Response When Under Attack (10% load):*** *The Avalanche and Reflector are configured to generate HTTP traffic through the sensor as for Test 4.2.1. The Spirent WebSuite software 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 (maximum 100Mbps - 148,000 packets per second). Note that with the background traffic, this rest will result in a maximum load of 70 per cent of the rated bandwidth of the device under test. The minimum, maximum and average page response times and number of failed connections are recorded by Avalanche to provide an indication of the expected response times when the device is under attack.*
- ***Test 4.2.3 - Web Response When Under Attack (20% load):*** *As for Test 4.2.2, but with a 20 per cent load of SYN flood traffic (single source IP, maximum 200Mbps - 296,000 packets per second). Note that with the background traffic, this rest will result in a maximum load of 70 per cent of the rated bandwidth of the device under test. The minimum, maximum and average page response times and number of failed connections are recorded by Avalanche to provide an indication of the expected response times when the device is under attack.*
- ***Test 4.2.4 - Web Response When Under Attack (40% load):*** *As for Test 4.2.2, but with a 40 per cent load of SYN flood traffic (single source IP, maximum 400Mbps - 592,000 packets per second). Note that with the background traffic, this rest will result in a maximum load of 90 per cent of the rated bandwidth of the device under test. The minimum, maximum and average page response times and number of failed connections are recorded by Avalanche to provide an indication of the expected response times when the device is under attack.*

Section 5 – Stability & Reliability

These tests attempt to verify the stability of the device under test under various extreme conditions. Long term stability is particularly important for an in-line device, where failure can produce network outages.

- ***Test 5.1.1 - Blocking Under Extended Attack:*** *For this test, we expose the external interface of the device to a constant stream of genuine traffic interspersed with occasional attack traffic for a total of 8 hours. This is not intended as a stress test in terms of traffic load - 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 attacks, raising an alert for each. Results are presented as a percentage of attacks mitigated out of the total generated through the device. If any recognisable attacks are not mitigated for any reason - caused by either the volume of traffic ***or*** *the sensor failing open - this will result in a FAIL.*

- ***Test 5.1.2 - Passing Legitimate Traffic Under Extended Attack:*** *This test is identical to 5.1.1, where we expose the external interface of the device to a constant stream of genuine traffic over an extended period of time, interspersed with occasional attack traffic. The device is expected to remain operational and stable throughout this test, and to pass 100 per cent of legitimate traffic.*

  Results are presented as a percentage of legitimate traffic passed out of the total generated through the device. If an excessive level (>0.09%) of legitimate traffic is blocked - caused by either the volume of traffic ***or*** *the sensor failing closed for any reason - this will result in a FAIL.*

- ***Test 5.1.3 - Resistance to ISIC/ESIC/TCPSIC/UDPSIC/ICMPSIC traffic:*** *This test attempts to stress the protocol stack of the device under test by exposing it to traffic from the ISIC test tool. The ISIC test tool host is connected directly to the external interface of the sensor, and the ISIC target directly to the internal interface. ISIC traffic is transmitted through the sensor (without passing through any other network equipment) and the effects noted. Traffic load is a maximum of 350Mbps and 60,000 packets a*

  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 to attain a PASS.

- ***Test 5.1.4 - Mitigation of ISIC/ESIC/TCPSIC/UDPSIC/ICMPSIC mitigation traffic:*** *As for Test 5.1.3. Results are presented as a simple PASS/FAIL - the device will receive a PASS should it prove capable of total mitigation of all the ISIC traffic.*

Section 6 – Management and Configuration

The aim of this section is to determine the features of the management system, together with the ability of the management port on the device under test to resist attack.

Test 6.1 - Management Port

Clearly the ability to manage the data collected by the sensor is a critical part of any 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 monitoring and tuning the mitigation device during an attack.

- ***Test 6.1.1 - Open ports:*** *We will scan the open ports and active services on the management interface and report on known vulnerabilities.*

- ***Test 6.1.2 - Resistance to ISIC/ESIC/TCPSIC/UDPSIC/ICMPSIC traffic:*** *This test attempts to stress the protocol stack of the management interface of the device under test by exposing it to traffic from the ISIC test tool. The ISIC test tool host is connected directly to the management interface of the IPS sensor, and that interface is also the target.*

ISIC traffic is transmitted to the management interface of the IPS device (without passing through any other network equipment) and the effects noted. 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 to attain a PASS.

- ***Test 6.1.3 - Detection of ISIC/ESIC/TCPSIC/UDPSIC/ICMPSIC traffic:*** *As for Test 6.1.2. We note whether the ISIC attacks themselves are detected by the sensor even though targeted at the management port.*

Top Layer IPS 5500 V3.3 Test Results (Rate-based)

Section 1 - Detection Engine

Test 1.1 – Attack Detection/Mitigation	Detected?	Mitigated?
Test 1.1.1 - SYN Flood	YES	YES
Test 1.1.2 - TCP SYN Attack (low-rate SYN flood)	YES	YES
Test 1.1.3 - ICMP Flood	NO	YES
Test 1.1.4 - Distributed Denial Of Service (DDOS) Attack	YES	YES
Test 1.1.5 - UDP Flood	YES	YES
Test 1.1.6 - IGMP Flood	YES	YES
Test 1.1.7 - Connection Flood (fast)	YES	YES
Test 1.1.8 - Connection Flood (slow)	YES	YES
Test 1.1.9 - Random protocol violations (invalid packets)	NO	YES
Test 1.1.10 - Trojan response (external host receives responses from Trojan)	NO	YES[1]
Test 1.1.11 - ICMP Sweep (inbound)	NO	YES[1]
Test 1.1.12 - ICMP Sweep (outbound)	NO	YES[1]
Test 1.1.13 - SQL Slammer	YES	YES
Test 1.1.14 - Spoofed IP Attack	YES	YES
Test 1.1.15 - Web Vulnerability Scan	YES	YES
Test 1.1.16 - TCP Port Scan (full connect)	YES	YES[2]
Test 1.1.17 - Stealth Port Scan	YES	YES
Test 1.1.18 - FIN Port Scan	YES	YES
Test 1.1.19 - UDP Port Scan	YES	YES[2]
Test 1.1.20 - NULL Port Scan	YES	YES
Test 1.1.21 - Xmas Port Scan	YES	YES
Test 1.1.22 - IP Protocol Port Scan	YES	NO
Test 1.1.23 - ACK Port Scan	YES	YES
Test 1.1.24 - Window Port Scan	YES	YES
Total	**19 / 24**	**23 / 24**

Test 1.2 – High Volume Attack Detection/Mitigation		100Mbps	200Mbps	400Mbps	800Mbps
Test 1.2.1 - SYN Flood DOS (single source IP)	Mitigated	PASS	PASS	PASS	PASS
	Pass legit	PASS	PASS	PASS	PASS
Test 1.2.2 - SYN Flood DDOS (multiple source IPs)	Mitigated	PASS	PASS	PASS	PASS
	Pass legit	PASS	PASS	PASS	FAIL[3]
Test 1.2.3 - Smurf	Mitigated	PASS	PASS	PASS	PASS
	Pass legit	PASS	PASS	PASS	FAIL[3]
Test 1.2.4 - Teardrop	Mitigated	PASS	PASS	PASS	PASS
	Pass legit	PASS	PASS	PASS	PASS
Test 1.2.5 - ICMP Flood	Mitigated	PASS	PASS	PASS	PASS
	Pass legit	PASS	PASS	PASS	FAIL[3]
Test 1.2.6 - UDP Flood	Mitigated	PASS	PASS	PASS	PASS
	Pass legit	PASS	PASS	PASS[4]	FAIL[3]

Test 1.3 – Resistance to False Positives	Pass/Fail
Test 1.3.1 - False positives	PASS[5]

Section 2 - Evasion Techniques

Test 2.1 – Evasion Baselines	Detected?	Mitigated?
Test 2.1.1 - UDP Flood	YES	YES
Test 2.1.2 - TCP Port Scan	YES	YES
Test 2.1.3 - Stealth Port Scan	YES	YES
Test 2.1.4 - Connection Flood	YES	YES
Total	**4 / 4**	**4 / 4**

Test 2.2 – Fragmentation and Timing	Detected?	Mitigated?
Test 2.2.1 - Fragmented UDP Flood (Teardrop)	YES	YES
Test 2.2.2 - Fragmented Stealth Port Scan	YES	YES
Test 2.2.3 - Slow Stealth Port Scan (0.4 secs between packets)	NO[6]	NO[6]
Test 2.2.4 - Very Slow Stealth Port Scan (15 secs between packets)	NO[6]	NO[6]
Test 2.2.5 - Slow Connection Flood (1 second between packets)	YES	YES
Test 2.2.6 - Slow Connection Flood (3 seconds between packets)	YES	YES
Total	**4 / 6**	**4 / 6**

Test 2.3 – URL Obfuscation	Detected?	Mitigated?
Test 2.3.1 - URL encoding	YES	YES
Test 2.3.2 - /./ directory insertion	YES	YES
Test 2.3.3 - Premature URL ending	YES	YES
Test 2.3.4 - Long URL	YES	YES
Test 2.3.5 - Fake parameter	YES	YES
Test 2.3.6 - TAB separation	YES	YES
Test 2.3.7 - Case sensitivity	YES	YES
Test 2.3.8 - Windows \ delimiter	YES	YES
Test 2.3.9 - Session splicing	YES	YES
Total	**9 / 9**	**9 / 9**

Section 3 - Detection/Mitigation Performance Under Load

Test 3.1 – UDP traffic to random valid ports		250Mbps	500Mbps	750Mbps	1Gbps	Max
Test 3.1.1 - 256 byte packet test - max 453,000pps	Mitigated	PASS	PASS	PASS	PASS	1Gbps
	Pass legit	PASS	PASS	PASS	PASS	
Test 3.1.2 - 550 byte packet test - max 220,000pps	Mitigated	PASS	PASS	PASS	PASS	1Gbps
	Pass legit	PASS	PASS	PASS	PASS	
Test 3.1.3 - 1514 byte packet test - max 122,000pps	Mitigated	PASS	PASS	PASS	PASS	1Gbps
	Pass legit	PASS	PASS	PASS	PASS	

Test 3.2 – HTTP "maximum stress" traffic with no transaction delays		250Mbps	500Mbps	750Mbps	1Gbps	Max
Test 3.2.1 - Max 2500 connections per second - ave packet size 1000 bytes - max 120,000 packets per second	Mitigated	PASS	PASS	PASS	PASS	1Gbps
	Pass legit	PASS	PASS	PASS	PASS	
Test 3.2.2 - Max 5000 connections per second - ave packet size 540 bytes - max 225,000 packets per second	Mitigated	PASS	PASS	PASS	PASS	1Gbps
	Pass legit	PASS	PASS	PASS	PASS	
Test 3.2.3 - Max 10000 connections per second - ave packet size 440 bytes - max 275,000 packets per second	Mitigated	PASS	PASS	PASS	PASS	1Gbps
	Pass legit	PASS	PASS	PASS	PASS	
Test 3.2.4 - Max 20000 connections per second - ave packet size 360 bytes - max 320,000 packets per second	Mitigated	PASS	PASS	PASS	PASS	1Gbps
	Pass legit	PASS	PASS	PASS	PASS	

Test 3.3 – HTTP "maximum stress" traffic with transaction delays		250Mbps	500Mbps	750Mbps	1Gbps	Max
Test 3.3.1 - Max 5000 connections per second - ave packet size 540 bytes - max 225,000 packets per second - 10 sec delay - max 5,000 open conns	Mitigated	PASS	PASS	PASS	PASS	1Gbps
	Pass legit	PASS	PASS	PASS	PASS	
Test 3.3.2 - Max 10000 connections per second - ave packet size 440 bytes - max 275,000 packets per second - 10 sec delay - max 100,000 open conns	Mitigated	PASS	PASS	PASS	PASS	1Gbps
	Pass legit	PASS	PASS	PASS	PASS	

Test 3.4 – Protocol mix		250Mbps	500Mbps	750Mbps	1Gbps	Max
Test 3.4.1 - 72% HTTP (540 byte packets) + 20% FTP + 6% UDP (256 byte packets). Max 4000 connections per second - ave packet size 540 bytes - max 215,000 packets per second - max 750 open connections	Mitigated	PASS	PASS	PASS	PASS	1Gbps
	Pass legit	PASS	PASS	PASS	PASS	

Test 3.5 – Real World traffic		250Mbps	500Mbps	750Mbps	1Gbps	Max
Test 3.5.1 - Pure HTTP (simulated browsing session on NSS Web site). Max 4700 connections per second - 2 new users per second - ave packet size 560 bytes - max 210,000 packets per second	Mitigated	PASS	PASS	PASS	PASS	1Gbps
	Pass legit	PASS	PASS	PASS	PASS	
Test 3.5.2 - Protocol mix - 72% HTTP (simulated browsing sessions as 2.5.1) + 20% FTP + 6% UDP (256 byte packets). Max 3700 connections per second - ave packet size 560 bytes - max 205,000 packets per second - max 1500 open connections	Mitigated	PASS	PASS	PASS	PASS	1Gbps
	Pass legit	PASS	PASS	PASS	PASS	

Test 3.6 - Stateful Operation	Result
Test 3.6.1 - Maximum simultaneous open TCP connections	1 million

Section 4 - Latency & User Response Times

Test 4.1 – Latency	Packet Size	25% load	50% load	75% load	100% load
Test 4.1.1 Average latency (µs) with no background traffic (max load 1Gbps pure UDP latency measurement traffic)	256	19.83	20.04	20.27	20.89
	550	27.29	27.65	27.64	27.93
	1000	40.11	40.15	40.46	40.63
Test 4.1.2 Average latency (µs) with pure legitimate HTTP traffic (max load 1Gbps - 2500 connections per second - ave packet size 540 bytes – 225,000 packets per second)	256	22.85	36.86	71.29	107.71
	550	30.52	42.70	77.52	111.10
	1000	42.55	57.05	90.48	129.76
Test 4.1.3 Average latency (µs) when under pure SYN Flood DOS attack (max load 800Mbps SYN packets – 1,184,000pps from one source IP)	256	20.06	20.10	20.22	22.76
	550	27.48	27.70	27.76	34.53
	1000	41.96	41.96	41.97	41.97

Test 4.2 – User Response Times	Attempted Trans	Failed Trans	Min Page Response	Max Page Response	Ave Page Response
Test 4.2.1 - Web page response (ms) with pure HTTP background traffic (500Mbps HTTP traffic - max 2500 connections per sec - ave packet size 540 bytes - max 112,500 packets per sec)	1543997	0	200	210	205
Test 4.2.2 - Web page response (ms) when under **10%** attack load (500Mbps HTTP traffic - max 2500 conns per sec - ave packet size 540 bytes - max 112,500pps **PLUS** 100Mbps SYN flood DOS (148,000pps from one IP)	1540730	0	200	211	205
Test 4.2.3 - Web page response (ms) when under **20%** attack load (500Mbps HTTP traffic, max 2500 conns per sec - ave packet size 540 bytes - max 112,500pps **PLUS** 200Mbps SYN flood DOS (296,000pps from one IP)	1543987	0	200	212	205
Test 4.2.4 - Web page response (ms) when under **40%** attack load (500Mbps HTTP traffic, max 2500 conns per sec - ave packet size 540 bytes - max 112,500pps **PLUS** 400Mbps SYN flood DOS (592,000pps from one IP)	1329242	0	200	212	205

Section 5 - Stability & Reliability

Test ID	Result
Test 5.1.1 - Mitigation under extended attack	100%
Test 5.1.2 - Passing legitimate traffic under extended attack	100%
Test 5.1.3 - Resistance to ISIC/ESIC/TCPSIC/UDPSIC/ICMPSIC traffic	PASS
Test 5.1.4 - Mitigation of ISIC/ESIC/TCPSIC/UDPSIC/ICMPSIC traffic	YES

Section 7 - Management Interface

Test ID	Result
Test 7.1.1 - Open ports	PASS
Test 7.1.2 - Resistance to ISIC/ESIC/TCPSIC/UDPSIC/ICMPSIC traffic	PASS
Test 7.1.3 - ISIC attacks detected against management interface?	NO

Notes:

1. Required the use of firewall rules to mitigate effectively
2. Partial mitigation
3. Latency through device too high - CPU = 100% - some packets dropped
4. Very high latency through the device – CPU almost 100% - HTTP response was slow, but Web transactions completed eventually
5. Careful configuration is required when imposing client/application rate/connection limiting to avoid false positives/accidental mitigation
6. Optional CMS is required to detect “low and slow” attacks

Section 1: Detection Engine

With no auto-learning mechanism built in to the Top Layer product, there was some significant configuration to perform before we could run our tests. We needed to set limits on ICMP and UDP traffic, set outbound client connection limits, and adjust the mid-flow and SYN flood mitigation mechanisms. All of these required some trial and error adjustment to ensure that attack traffic was mitigated successfully whilst legitimate traffic was allowed to pass unimpeded. Apart from the rate-based adjustments, an identical configuration was used to the content-based testing.

Attack detection/mitigation was excellent, with the IPS 5500 detecting most of our attacks, and successfully mitigating all but one of them. Partial mitigation was effected for both the TCP (full connect) and UDP port scans, where partial scan results were returned to the attacker.

It should also be noted that it was necessary to use the IPS 5500 firewall filtering in order to successfully block the Back Orifice Trojan response, and both the inbound and outbound ICMP sweeps. With a strong firewall capability, the IPS 5500 does tend to rely on its firewall filtering to block many of these types of attacks rather than utilising detection and mitigation mechanisms.

Performance in the high volume detection/mitigation test was almost impeccable across the board, with perfect detection and mitigation at all load levels.

However, some problems were noted in passing legitimate traffic at the highest load levels of the SYN Flood DDOS, Smurf DDOS, ICMP Flood DDOS and UDP Flood DDOS attacks. At the 800Mbps load levels we noted the 5500's CPU utilisation was pegged at 100 per cent, causing some packet loss and inevitable impediment to legitimate traffic.

However, these load levels can be considered excessive, and the device performed almost impeccably up to the 600Mbps level of attack traffic.

A major concern in deploying an in-line device is the blocking of legitimate traffic. Careful configuration is required when imposing client/application rate/connection limiting to avoid accidental mitigation, and a self-imposed DoS condition.

Section 2: Evasion Techniques

Resistance to our evasion attempts proved very good, with the IPS 5500 successfully detecting all of the fragmented and obfuscated attacks which we ran. However, because the device relies on a "brute force" connections/SYNs per second type of approach to detect many port scans, it was trivial to evade it by running these at a much slower rate than normal.

This does not work with all rate-based attacks – the connection floods were still detected (eventually), no matter how slow we ran them, for example.

However, we feel that detecting and mitigating reconnaissance probes such as port scans is not the strongest feature of the IPS 5500, which is much better suited to protecting against high-volume DOS/DDOS attacks.

Section 3: Detection/Mitigation Performance Under Load

Performance at all levels of our load tests was impeccable, with 100 per cent of all attacks being detected and mitigated under all load conditions, and no interruptions to legitimate sessions. We would happily confirm Top Layer's 1Gbps rating for this device as a rate-based IPS appliance.

Out of the box, the IPS 5500 handled 1 million simultaneously open connections without tuning - this is not configurable.

Section 4: Latency & User Response Times

Basic latency figures were excellent at all traffic loads and with all packet sizes, ranging from 20µs with 250Mbps of 256 byte packets, to 41µs with 1Gbps of 1000 byte packets. Behaviour throughout the tests with no background traffic was very even and predictable at all load levels and with all packet sizes.

Placing the device under an increasing load of HTTP traffic (ranging from 250Mbps to 1Gbps) had some effect on the latency figures, which ranged from 23µs with 250Mbps of 256 byte packets, to 130µs with 1Gbps of 1000 byte packets.

All of these figures are well inside the limits we would expect to see for a 1Gbps device, meaning the IPS 5500 could be situated anywhere on a Gigabit network, either internally or at the perimeter.

Naturally, performance when under attack is critical for an attack mitigation device, and the IPS 5500 excelled. Mitigation was handled well at all levels in our high-volume SYN flood DOS test, with latencies ranging from just 20µs at 200Mbps of SYN flood traffic to 42µs at 800Mbps.

While the attack mitigator could defend against the entire 800Mbps (and beyond) of single-source IP SYN flood (DOS) that we fired at it, when the attack was distributed amongst a larger number of source addresses (DDOS), its ability to defend against the attack was exhausted at a lower rate due to high CPU utilisation. However this is a very challenging attack, and we have not yet seen any device that has performed even this well in our tests.

HTTP response times were remarkably consistent even when under heavy attack. Maximum transaction response times barely increased as we subjected our 500Mbps of background HTTP traffic to increasing levels of SYN flood traffic ranging from 100Mbps to 400Mbps, and average response times remained identical across all tests.

Section 5: Stability & Reliability

The IPS 5500 performed consistently and completely reliably throughout our tests. Under eight hours of extended attack (comprising millions of sessions of legitimate traffic interspersed with some attacks) it continued to mitigate 100 per cent of attack traffic, whilst passing 100 per cent of legitimate traffic.

Exposing the sensor interface to ISIC-generated traffic had no adverse effects, and the device continued to detect and block all other exploits throughout and following the ISIC attack.

The IPS mitigates the ISIC attack partially at first (detecting a wide range of alerts), but eventually sees it as DDOS attack and provides almost total mitigation.

There were no residual stability problems once the attack had been terminated.

Section 6: Management Interface

Open ports on the management interface are restricted to HTTP, HTTPS, NTP and SNMP. HTTP can be disabled once HTTPS has been configured.

The extended ISIC attack against the management interface had virtually no effect on the appliance and its ability to detect and block attacks, though there was a slight delay in sensor to console communications throughout. No alerts were raised during the attack.

The sensor continued to detect and block malicious traffic whilst passing legitimate traffic throughout and following the ISIC attack, and there were no residual stability problems.