Q2 2022 Fortinet AAA

Overview

We performed an independent test of the Fortinet Cloud Network Firewall (CNFW) v7.0.5 Build 0304(GA) on Amazon Web Services. The Instance used for this test was c5.9xlarge (36 vCPU, 72 GB memory, and 10+ Gbps network Bandwidth). The product was excellent overall and should be on everyone's short list.

Threat Protection was excellent; Fortinet blocked 35 out of 35 evasion techniques, and 977 out of 977 exploits. The device passed all stability and reliability tests. The HTTP Rated Throughput was 1,000 Mbps; TLS/HTTPS Rated Throughput was 892 Mbps, giving Fortinet an excellent combined Rated Throughput of 946 Mbps.

AAA | Management & Reporting Capabilities

AAA | Routing and Policy Enforcement

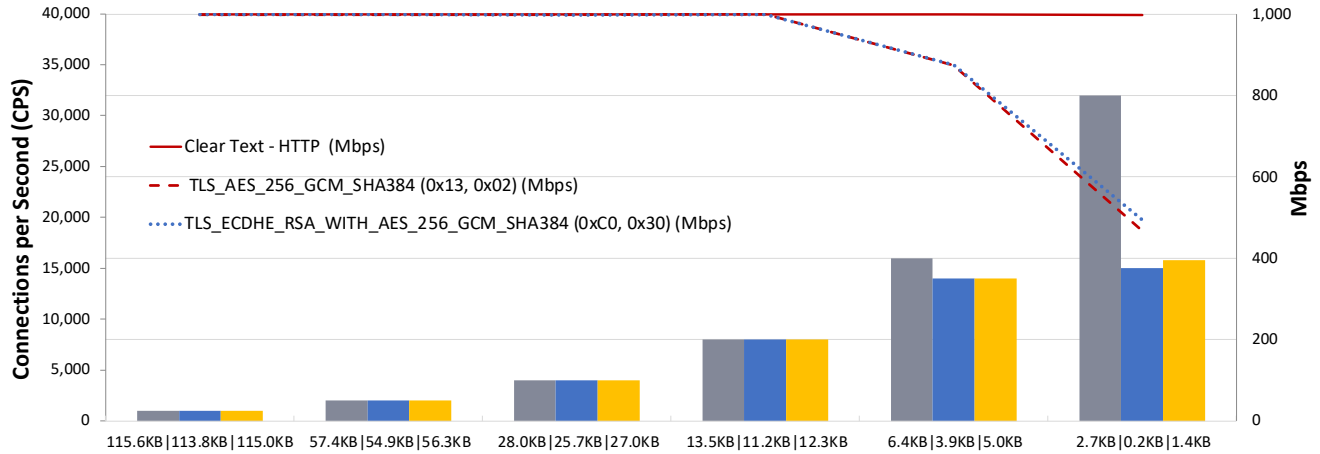
AAA | SSL/TLS Functionality

AAA | Threat Prevention

AAA | Performance

Summary of Results

Threat Prevention	Samples Tested	Samples Blocked	Blocked %
Exploit Block Rate (No Background Network Load)	977	977	100%
Exploit Block Rate (With Background Network Load)	977	977	100%
Evasion Techniques	35	35	100%
False Positive Testing	PASS		
Stability & Reliability	PASS		
SSL/TLS Functionality			
Current Cipher Suites			9/9
Insecure Cipher Suites			Reports Error
Decryption Validation			PASS
Decryption Bypass Exceptions			PASS
TLS Session Reuse			PASS



HTTP & HTTPS Performance Max Connections per Second (CPS)	Clear Text (HTTP)		TLS_AES_256_GCM_SHA384 (0x13, 0x02)		TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xC0, 0x30)	
	CPS	Mbps	CPS	Mbps	CPS	Mbps
115.6 KB | 113.8 KB | 115.0 KB | @ 1,000 CPS	1,000	1,000	999	999	1,000	1,000
57.4 KB | 54.9 KB | 56.3 KB | @ 2,000 CPS	2,000	1,000	1,998	999	2,000	1,000
28.0 KB | 25.7 KB | 27.0 KB | @ 4,000 CPS	4,000	1,000	3,997	999	3,992	998
13.5 KB | 11.2 KB | 12.3 KB | @ 8,000 CPS	8,000	1,000	7,998	1,000	8,001	1,000
6.4 KB | 3.9 KB | 5.0 KB | @ 16,000 CPS	16,000	1,000	14,000	875	14,000	875
2.7 KB | 0.2 KB | 1.4 KB | @ 32,000 CPS	31,990	1,000	15,000	469	15,800	494

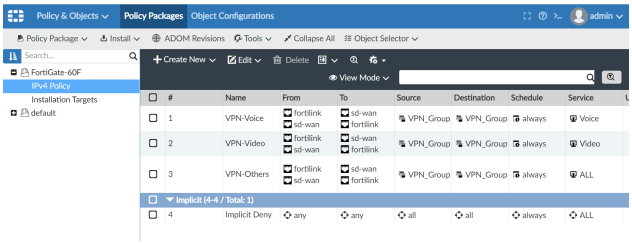
Management & Reporting Capabilities AAA

Authentication

FortiManager supports five role-based access control (RBAC) methods for the following users: Restricted User, Standard User, Super User, Package User, No Permission User, and custom user roles. Authentication is local or through a third-party authentication such as LDAP, RADIUS, TACACS+, PKI, Group, and SSO.

Policy

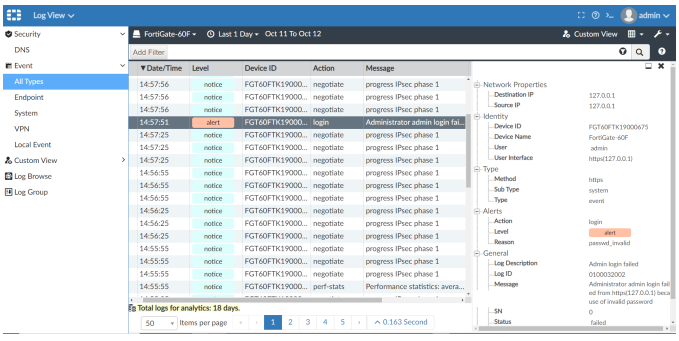
FortiManager supports using existing templates, creating new and saving multiple policies for general network settings, firewall, antivirus, web filter, app control, Intrusion prevention, SSL/TLS inspection, SD-WAN, and more. Administrators then create groups and apply policies easily configured from the GUI by either selecting or deselecting options, or using advanced options that allow for further customization.



Once policies have been defined, they can be associated with domains, specific sensors, groups of sensors, all sensors, individual ports, port groups, etc. In addition, policy checks, differentials, versioning, and rollback are supported natively in the system.

Inheritance (nested rules) is fully supported, including the creation of groups and sub-groups, such that sub-groups can inherit certain aspects of configuration and policy definition from parent groups.

Logging

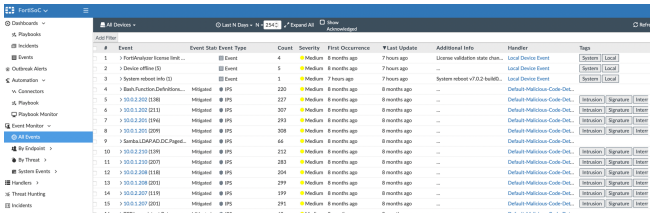


Logging is robust and includes everything from policy changes, policy deployed, unsuccessful logins, successful logins, hardware failure, malicious traffic, etc.

Logfile maintenance is included, such as archiving, rotation of log files, and restoring from the archive. Furthermore, Fortinet offers integration with third parties for log handling. Logs can be forwarded to any third party in Syslog and IPFIX format, but Fortinet also natively supports Splunk, ArcSight, and their own FortiSiEM.

Alert Handling

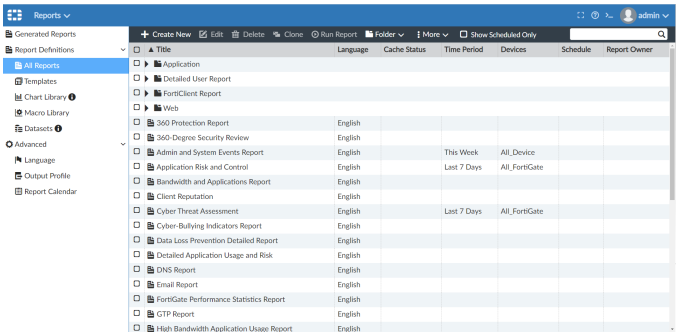
Alerts can be viewed in FortiSOC; alerts are found in Incidents & Events if not enabled. Furthermore, custom alert handlers can be created for certain thresholds, such as security, hardware, etc. Auto-remediation can be enabled for events to help automate tasks. Alerts can be acknowledged, commented on, assigned, created as new, or added to existing incidents.



The administrator views alerts through a detailed list of alerts, which can be filtered, and each alert offers details about the device, configuration, details of what occurred, etc. It is also possible to search for an alert.

Reporting

The system provides summary reporting on all alerts in FortiManager. In addition, it offers over 60 built-in reports covering typical requirements, such as a list of bandwidths used, users, top applications, SLA violations, PCI DSS 3.2.1, and Wireless PCI Compliance. Support for reporting format standards such as Syslog and JSON are also included.



The system includes a report generator that can construct complex data filters and summarize alerts on the specified criteria to customize a report. Reports are exportable as HTML, XML, CSV, or PDF formats and can be generated on demand, scheduled for delivery, or saved for subsequent use.

Change Control

Change control, rollback, and revision history are available.

Routing and Policy Enforcement AAA

Access control is the primary responsibility of a firewall. Firewalls have undergone several stages of development, from early packet filtering and circuit relay firewalls to application-layer (proxy-based), dynamic packet filtering firewalls, and user/application-aware "next-generation" firewalls. Throughout its history, the goal has been to enforce an access control policy between two networks. Rules were configured to permit or deny traffic from one network resource to another based on identifying criteria such as source IP, destination IP, source port, destination port, and protocols.

This test validates that the firewall enforces security policies over a range of policy environments, from simple to complex. The tests incrementally build on a baseline consisting of a simple configuration with no policy restrictions and no content inspection – to a complex multiple-zone configuration that supports many users, networks, policies, and applications. At each level of complexity, traffic was tested to ensure specified policies were enforced.

Network Segmentation	AAA
Unrestricted Traffic Test	✓
Segmented Traffic Test	✓
Access Control	**AAA**
Simple Policies	✓
Complex Multi-Zone Policies	✓

SSL/TLS Functionality AAA

As of May 13, 2022, data collected by W3Techs[1] showed that over 79.3% of web traffic is encrypted (HTTPS). To confirm that the firewall was correctly decrypting SSL/TLS traffic, we conducted a functional validation test prior to performance testing. Cipher suites are selected based on the published current frequency of use[2] and security status[3].

Cipher Suite Support

Supported Cipher suites accounted for ~98% of all HTTPS websites.

Version	Prevalence	Cipher Suites	AAA
TLS 1.3	60.5%	TLS_AES_256_GCM_SHA384 (0x13, 0x02)	✓
TLS 1.2	16.3%	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xC0, 0x30)	✓
TLS 1.2	11.7%	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xC0, 0x2F)	✓
TLS 1.3	6.7%	TLS_AES_128_GCM_SHA256 (0x13, 0x01)	✓
TLS 1.2	1.5%	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xC0, 0x28)	✓
TLS 1.2	1.3%	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xCC, 0xA8)	✓
TLS 1.3	0.5%	TLS_CHACHA20_POLY1305_SHA256 (0x13, 0x03)	✓
TLS 1.2	0.4%	TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xCC, 0xA9)	✓
TLS 1.2	0.3%	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xC0, 0x2C)	✓

Decryption Validation

Prevention of Insecure Ciphers	AAA
Null ciphers (no encryption of data)	✓
Anonymous Ciphers (no authorization)	Reports Error
Additional SSL/TLS Functionality	**AAA**
Decryption Validation	✓
Decryption Bypass Exceptions	✓
TLS Session Reuse	✓

First, we tested how the firewall handled cipher suites known to be insecure, using null ciphers (no encryption of data) and anonymous ciphers (no authorization). Then we validated the ability to correctly decrypt and inspect SSL/TLS traffic by using prohibited content previously blocked during testing. The content was then transmitted using encryption and verified that it was still blocked. We then tested to see if we could permit conditional bypass of decryption. This might be required to preserve privacy, either for regulatory or other reasons. Lastly, we tested TLS session reuse; to improve performance and reduce the overhead associated with conducting the full handshake for each session. The TLS protocol allows for abbreviated handshakes, which reuse previously established sessions.

[1] Usage Statistics of Default Protocol HTTPS for Websites as of May 13, 2022, https://w3techs.com/technologies/details/ce-httpsdefault

[2] Published international daily cipher suite usage can be found at https://crawler.ninja/files/ciphers.txt

[3] A list of cipher suites and associated attributes including security ratings can be found at https://ciphersuite.info/cs/

Threat Prevention

AAA

A cloud network firewall is a mechanism used to protect a trusted network from an untrusted network while allowing authorized communications to pass from one side to the other, thus facilitating secure business use of the Internet.

False Positives

A key to effective protection is the ability to correctly identify and allow legitimate traffic while maintaining protection against malware, exploits, and phishing attacks. False positives are any legitimate, non-malicious content/traffic that are perceived as malicious. False positive tests flex the ability of the firewall to block attacks while permitting legitimate traffic. If a device experienced false positive events, it was tuned until no further false positive events were encountered.

The CyberRatings exploit repository contains exploits that demonstrate a wide range of protocols and applications. Exploit sets for individual tests are selected based on CVSS score (how widely used is an application + what can an attacker do?), use case, and relevance to customers. This has implications for the age of exploits since some applications in industrial environments are deployed and then left untouched for years while other applications within office environments are refreshed every 5-7 years. First, we tested the firewall with no background network load to see how effective the firewall was at blocking exploits when protection was not resource constrained. Then we tested it with background network load.

Exploits

	Blocked
	977/977(100%)

To be eligible for security effectiveness testing, the firewall must perform all the tests included in the methodology with its protection against network-delivered exploitation features enabled.

An exploit is an attack that takes advantage of a vulnerability in a protocol, product, operating system, or application. CyberRatings verified that the firewall could detect and block exploits while remaining resistant to false positives by attempting to send exploits through the product under test; and verified that the malicious traffic was blocked, and all appropriate logging and notifications were performed.

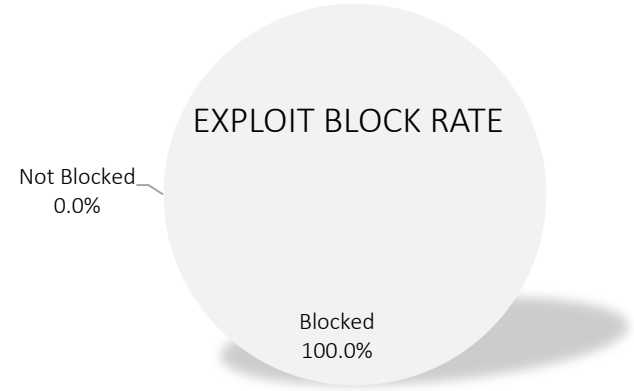


Figure 1 – Exploit Block Rate

Resistance to Evasions

	Blocked
	35/35 (100%)

Threat actors apply evasion techniques to disguise and modify attacks to avoid detection by security products. Therefore, it is imperative that a firewall correctly handles evasions. An attacker can bypass protection if a firewall fails to detect a single form of evasion.

Our engineers verified that the firewall could block exploits when subjected to numerous evasion techniques. To develop a baseline, we took several previously blocked attacks. We then applied evasion techniques to those baseline samples and tested them. This ensured that any misses were due to the evasions and not the baseline samples.

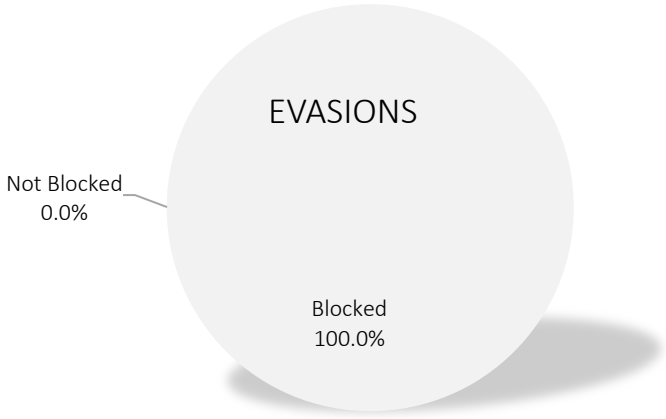


Figure 2 – Resistance to Evasions

Performance AAA

Cloud security architects are tasked with designing environments that scale. Making an informed decision regarding the performance of a cloud network firewall in an environment requires an understanding of the impact an appliance will have on traffic passing through it under various load conditions. The tests in this section measured the firewall's performance using traffic scenarios selected to provide realistic performance expectations. Individual implementations will vary based on the use case; however, these quantitative metrics inform whether a particular device is appropriate for a given environment. Each test was performed concurrently without the firewall to provide a baseline control. The test was then repeated with the firewall in the exact same configuration as used for exploits and evasions. Results are reported both as measured, relative to the baseline, and in context with other measurable attributes and confounding variables.

Rated Throughput

While the cloud firewall can burst to higher than 1,000 Mbps, depending on the file sizes, connections per second, time of day, and so on; our aim for this test was to determine if the cloud firewall could sustain 1,000 Mbps traffic over time, for a range of packet sizes and connections per second. We measured performance with different packet sizes and payloads in order to capture the firewall's performance curves for UDP, HTTP, and HTTPS.

The "Plain Text Rated Throughput," "HTTPS Rated Throughput," and the combined "Rated Throughput" are good benchmarks for what an enterprise can expect the firewall instance to achieve consistently [over time] when deployed on AWS.

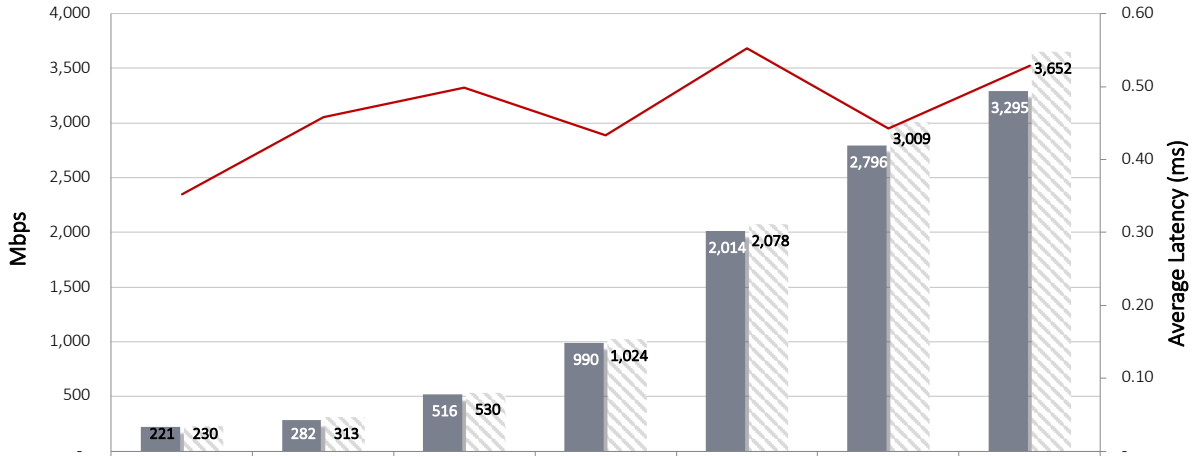
Performance		AAA
Plain Text Rated Throughput (Average of HTTP capacity —without delays)	1,000 Mbps	Rated Throughput **946 Mbps**
HTTPS (SSL/TLS) Rated Throughput (Average of all HTTPS Capacity tests)	892 Mbps	

Figure 3 – Rated Throughput (Mbps)

Raw Packet Processing Performance (UDP Throughput)

This test used UDP packets of varying sizes generated by traffic generation tools. A constant stream of the appropriate packet size — with variable source and destination IP addresses transmitting from a fixed source port to a fixed destination port — was transmitted bi-directionally through the firewall. Each packet contained dummy data and was targeted at a valid port on a valid IP address on the target subnet. Testing determined the maximum rate the firewall could process raw packets of various sizes, the associated latency, and the number of dropped packets.

This traffic did not attempt to simulate any form of real-world network condition. No TCP sessions were created during this test, and there was very little for the detection engine to do. However, each vendor was required to write a signature to detect the test packets to ensure that they were being passed through the detection engine and not "fast-tracked" from the inbound port to the outbound port.

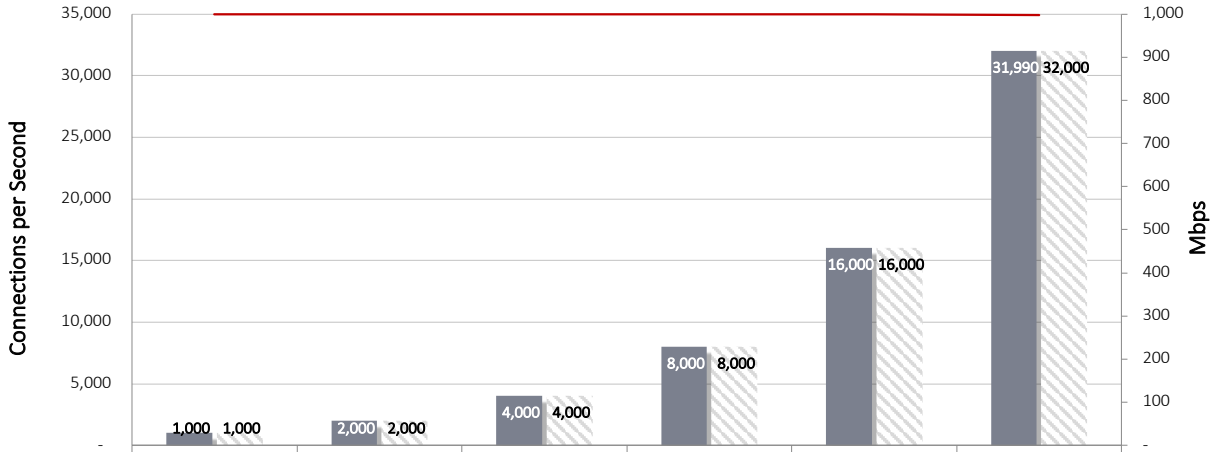


	64 -byte	128 -byte	256 -byte	512 -byte	1024 -byte	1280 -byte	1518 -byte
Fortinet Tested Throughput (Mbps)	221	282	516	990	2,014	2,796	3,295
AWS Maximum Throughput (Mbps)	230	313	530	1,024	2,078	3,009	3,652
Fortinet Average Latency (ms)	0.35	0.46	0.50	0.43	0.55	0.44	0.53

Figure 4 - Raw Packet Processing Performance (UDP Traffic)

HTTP Capacity

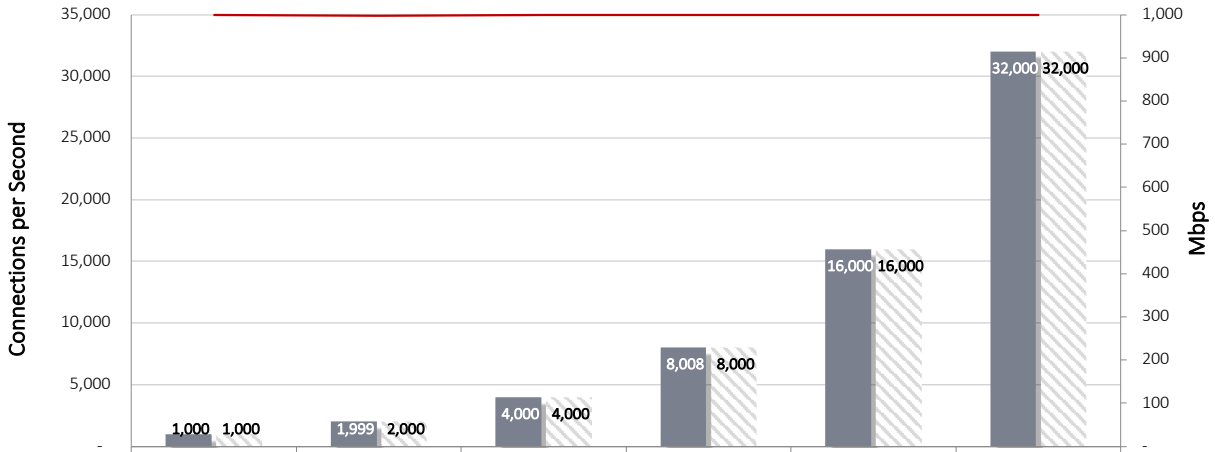
The goal was to stress the HTTP detection engine and determine how the device copes with network loads of varying average packet sizes and varying connections per second. By creating genuine session-based traffic with varying session lengths, the device was forced to track valid TCP sessions, thus ensuring a higher workload than simple packet-based background traffic. This provided a test environment as close to real-world conditions as possible in a lab environment while ensuring absolute accuracy and repeatability.



	115.6 KB Response	57.4 KB Response	28.0 KB Response	13.5 KB Response	6.4 KB Response	2.7 KB Response
Fortinet Tested Throughput (CPS)	1,000	2,000	4,000	8,000	16,000	31,990
AWS Maximum Throughput (CPS)	1,000	2,000	4,000	8,000	16,000	32,000
Fortinet Tested Throughput (Mbps)	1,000	1,000	1,000	1,000	1,000	1,000

Figure 5 - HTTP Capacity (with delays)

Each transaction consisted of a single HTTP GET request with delays (i.e., the web browser/client waits ten seconds to "read" the content provided by the web server. The web server then responds immediately, after the web browser/client clicks to the next page thus maintaining each connection for ten seconds). All packets contained valid payloads.



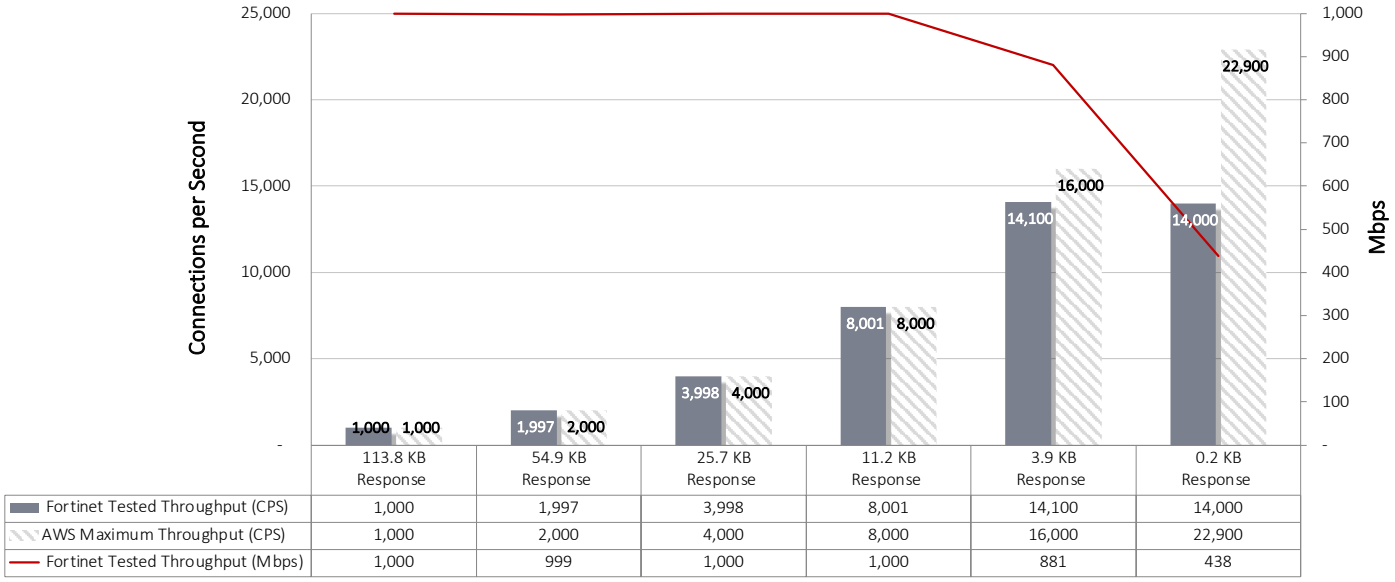
	115.6 KB Response	57.4 KB Response	28.0 KB Response	13.5 KB Response	6.4 KB Response	2.7 KB Response
Fortinet Tested Throughput (CPS)	1,000	1,999	4,000	8,008	16,000	32,000
AWS Maximum Throughput (CPS)	1,000	2,000	4,000	8,000	16,000	32,000
Fortinet Tested Throughput (Mbps)	1,000	1,000	1,000	1,000	1,000	1,000

Figure 6 - HTTP Capacity (without delays)

Each transaction consisted of a single HTTP GET request, and there were no transaction delays (i.e., the webserver responded immediately to all requests). All packets contained valid payload (a mix of binary and ASCII objects) and address data. This test provided an excellent representation of a live network (albeit one biased towards HTTP traffic) at various network loads. Testing determined the maximum rate the firewall was able to process HTTP packets of multiple sizes and its efficiency at forwarding packets quickly to provide the highest level of network performance with the lowest latency. The results were recorded at each response size at a load level of 95% of the maximum throughput with zero packet loss.

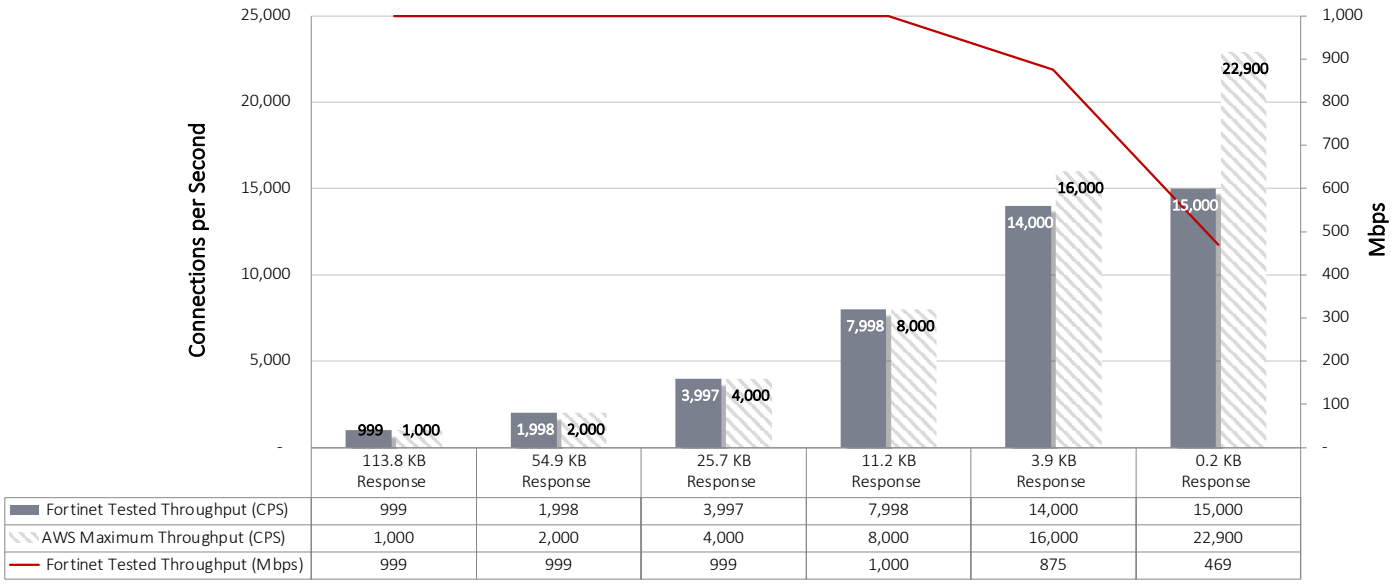
SSL/TLS Capacity

The goal was to stress the HTTPS engine and determine how the device coped with network loads of varying average packet sizes and varying connections per second. By creating session-based traffic with varying session lengths, the device was forced to track valid TCP sessions, thus ensuring a higher workload than simple packet-based background traffic. Encrypting the traffic using SSL/TLS with varying algorithms forced the device to decrypt traffic before inspection, increasing the workload further. This provided a test environment that is as close to real-world conditions as possible to achieve in a lab environment (albeit biased towards HTTPS traffic) while ensuring accuracy and repeatability. Tests were conducted with one transaction per connection; a single (1) HTTP(S) GET request. There were no transaction delays (i.e., the webserver responded immediately to all requests), and all packets contained valid payloads (a mix of binary and ASCII objects) and address data. Testing determined the maximum rate the firewall was able to process HTTPS packets of various sizes and its efficiency at forwarding packets quickly to provide the highest level of network performance with the lowest latency. The results were recorded at each response size at a load level of 95% of the maximum throughput with zero packet loss.



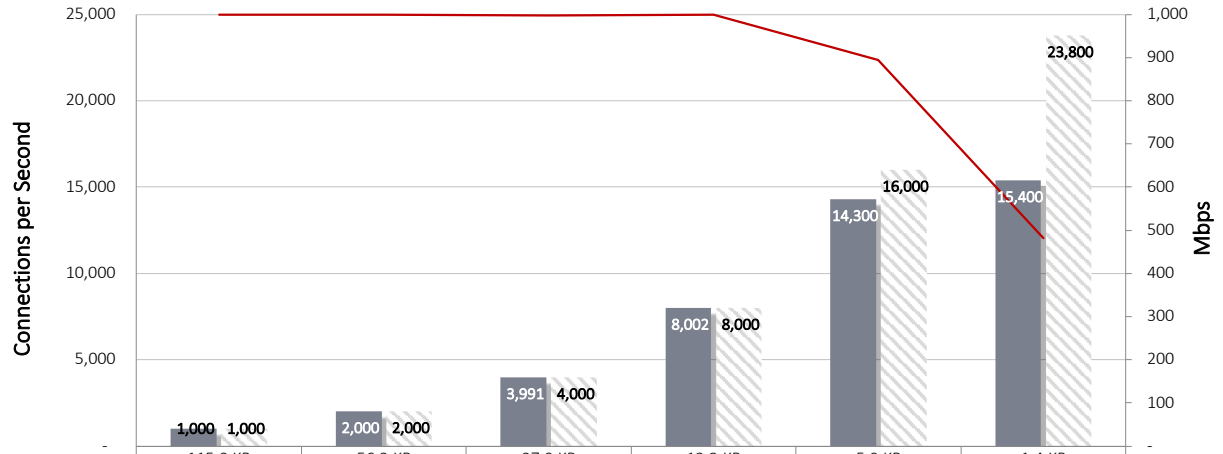
	113.8 KB Response	54.9 KB Response	25.7 KB Response	11.2 KB Response	3.9 KB Response	0.2 KB Response
Fortinet Tested Throughput (CPS)	1,000	1,997	3,998	8,001	14,100	14,000
AWS Maximum Throughput (CPS)	1,000	2,000	4,000	8,000	16,000	22,900
Fortinet Tested Throughput (Mbps)	1,000	999	1,000	1,000	881	438

Figure 7 - HTTPS Capacity (TLS_AES_128_GCM_SHA256 (0x13, 0x01))



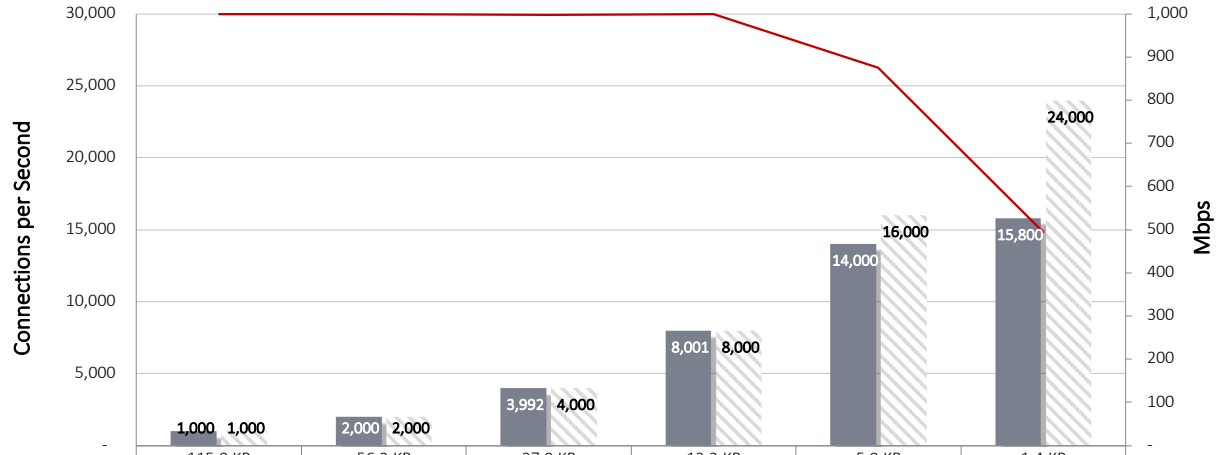
	113.8 KB Response	54.9 KB Response	25.7 KB Response	11.2 KB Response	3.9 KB Response	0.2 KB Response
Fortinet Tested Throughput (CPS)	999	1,998	3,997	7,998	14,000	15,000
AWS Maximum Throughput (CPS)	1,000	2,000	4,000	8,000	16,000	22,900
Fortinet Tested Throughput (Mbps)	999	999	999	1,000	875	469

Figure 8 - HTTPS Capacity (TLS_AES_256_GCM_SHA384 (0x13, 0x02))



	115.0 KB Response	56.3 KB Response	27.0 KB Response	12.3 KB Response	5.0 KB Response	1.4 KB Response
Fortinet Tested Throughput (CPS)	1,000	2,000	3,991	8,002	14,300	15,400
AWS Maximum Throughput (CPS)	1,000	2,000	4,000	8,000	16,000	23,800
Fortinet Tested Throughput (Mbps)	1,000	1,000	998	1,000	894	481

Figure 9 - HTTPS Capacity (TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xC0, 0x2F)



	115.0 KB Response	56.3 KB Response	27.0 KB Response	12.3 KB Response	5.0 KB Response	1.4 KB Response
Fortinet Tested Throughput (CPS)	1,000	2,000	3,992	8,001	14,000	15,800
AWS Maximum Throughput (CPS)	1,000	2,000	4,000	8,000	16,000	24,000
Fortinet Tested Throughput (Mbps)	1,000	1,000	998	1,000	875	494

Figure 10 - HTTPS Capacity (TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xC0, 0x30)

Stability and Reliability AAA

Long-term stability is essential for a firewall, where failure can produce network outages. These tests verified the firewall's stability and its ability to maintain security effectiveness while under normal load and while passing malicious traffic. Products that could not sustain legitimate traffic (or that crash) while under hostile attack did not pass.

The product was required to remain operational and stable throughout these tests and to block 100% of previously blocked traffic, raising an alert for each. If any policy-forbidden traffic passes, caused by either the volume of traffic or by the product failing open for any reason, this results in a fail.

Protocol Fuzzing & Mutation

Testing determined how the firewall responded (e.g., crashes, reboots, etc.) due to traffic generated from various protocol randomizers and mutation tools. The product was expected to remain operational and capable of detecting and blocking exploits throughout the test.

Blocking Under Extended Attack

This test provided an indication of the ability of the firewall to remain operational and stable (i.e., block violations and raise associated alerts) throughout an extended attack.

- Blocking with Minimal Load: A continuous stream of security policy violations mixed with legitimate traffic was transmitted through the firewall for an extended period of time with no additional background traffic. This was not intended as a stress test for traffic load (covered in the performance section); it was a reliability test for consistency of blocking
- **Blocking Under Load:** This was intended as a stress test. This test added legitimate traffic to the Blocking with Minimal Load test.

Behavior of the State Engine Under Load

This test determined whether the device could preserve state across a large number of open connections over an extended period of time. At various points throughout the test (including after the maximum had been reached), it was confirmed that the device was still capable of inspecting and blocking traffic that violated the currently applied security policy while ensuring that legitimate traffic was not blocked (perhaps as a result of exhaustion of the resources allocated to state tables).

- **Attack Detection/Blocking – Normal Load:** This test determined whether the device could enforce the policy as the number of concurrent open connections increased.
- **State Preservation – Normal Load:** A legitimate HTTP session is opened, and the first packet of a two-packet exploit is transmitted. (Both halves of the exploit are required to trigger an alert.) As the number of open connections approaches the maximum, the initial HTTP session is completed with the second half of the exploit, and the session is closed. If the cloud network firewall is still maintaining the state on the original session, the exploit will be recorded and blocked. If it is not, the exploit string will not be reconstructed properly and the attack will be successful.
- Pass Legitimate Traffic – Normal Load: This test ensured that the product continued to pass legitimate traffic as the number of open sessions reached 75% of the maximum determined previously in performance testing.
- **State Preservation – Maximum Exceeded:** This test determined whether the product maintained the state of pre-existing sessions as the number of open sessions exceeded the maximum determined previously in performance testing.
- Drop Traffic – Maximum Exceeded: This test ensured that the product continued to drop all traffic as the number of open sessions exceeded the maximum determined previously in performance testing.

Stability and Reliability	Result
Protocol Fuzzing & Mutation	✓
Blocking under Extended Attack	✓
Blocking with Minimal Load	✓
Blocking Under Load	✓
Behavior of the State Engine under Load	✓
Attack Detection/Blocking – Normal Load	✓
State Preservation – Normal Load	✓
Pass Legitimate Traffic – Normal Load	✓
State Preservation – Maximum Exceeded	✓
Drop Traffic – Maximum Exceeded	✓

Three-Year Total Cost of Ownership

When calculating TCO for a cloud firewall, there are several considerations to consider:

- **First**, there is the cost of the cloud provider and the specific price for the cloud firewall instance under consideration.
- **Secondly**, some instances offer a guaranteed level of throughput; others offer boosts up to a certain amount of throughput – but often fail to specify what type of traffic – for a given period.
- **Thirdly**, there is the ongoing cost of running the instance (cost per hour), which can be different for the region selected, etc.,

From a cloud firewall perspective, traditional licenses are offered, but so are bundles, which could be as long as three or five years. What should the bundle contain? An example could be IoT detection, Antivirus, Application Control, Web & Video Filtering, Antispam Service etc. There are pay-as-you-go options, which are charged hourly, daily, etc. Furthermore, enterprises should include labor costs for operational expenditures (OPEX) such as administration, policy and configuration handling, log handling, alert handling, monitoring, reporting, analysis, auditing and compliance, maintenance, software updates, and troubleshooting. To calculate the exact TCO for a given enterprise becomes very complex. To measure the value of the firewall, we can use the following formula:

Security Effectiveness **=** Exploit Block Rate* Evasions* Stability and Reliability
TCO per Protected Mbps = TCO / (*Security Effectiveness* * *Tested Throughput*)

Figure 11 – Security Effectiveness and TCO per Protected Mbps Formulas

This formula incorporates the cost of the cloud firewall, the instance costs, and how effective the firewall is in delivering both security and performance over time. The *TCO per Protected Mbps* metric provides clear guidance on whether a product's price is higher or lower than its competitors.

Figure 12 contains the list price for a few of Fortinet's licensing options. It is not meant as an extensive list. The pricing is provided by Fortinet and assumes Unified Threat Protection (UTP) with IPS, Advanced Malware Protection, Application Control, Web & Video Filtering, Antispam Service, and 24/7 FortiCare Premium.

vCPU	Fortinet Model	License Cost 1 year	License Cost 3 year	License Cost 5 year
1 vCPU	FGT-VM 01	$1,753	$5,259	$8,765
2 vCPU	FGT-VM 02	$1,957	$5,871	$9,785
4 vCPU	FGT-VM 04	$4,012	$12,036	$20,060
8 vCPU	FGT-VM 08	$11,285	$33,855	$56,425
16 vCPU	FGT-VM 16	$23,895	$71,685	$119,475
32 vCPU	FGT-VM 32	$30,680	$92,040	$153,400
72 vCPU	FGT-VMUL	$49,088	$147,264	$245,440

Figure 12 – Fortinet list price

If a customer picks the FGT-VMUL (unlimited) annual subscription as the license, and picks the c5.9xlarge in AWS (North-Virginia), with hours cost of $1.53, then the calculation would be as follows:

	Cost c5.9xlarge	Daily Cost	Annual Cost	AWS Cloud + Fortinet Cost
AWS	$1.53	$36.72	$13,402.80	$62,490.80
Fortinet			$49,088.00	

Now, we can calculate the TCO:

AWS Cloud + Fortinet Cost	Exploit Block Rate	Evasions	Stability & Reliability	Tested Throughput	TCO per Protected Mbps
$62,490.80	100%	100%	100%	946	$66.06

Figure 13 – TCO Calculation

Appendix A – Scorecard

Test Configuration			
Vendor		Fortinet	
Cloud Service Provider		AWS	
AWS Instance		c5.9xlarge	
Device		v7.0.5 Build 0304(GA)	
vCPU		36	
Memory		72 GB	
Routing Functionality		Result	
Unrestricted Traffic Test		PASS	
Segmented Traffic Test		PASS	
Access Control		Result	
Simple Policies		PASS	
Complex Multi-Zone Policies		PASS	
SSL/TLS Support			
Cipher Suites	Prevalence	Version	Result
TLS_AES_256_GCM_SHA384 (0x13, 0x02)	60.5%	TLS 1.3	100%
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xC0, 0x30)	16.3%	TLS 1.2	100%
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xC0, 0x2F)	11.7%	TLS 1.2	100%
TLS_AES_128_GCM_SHA256 (0x13, 0x01)	6.7%	TLS 1.3	100%
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xC0, 0x28)	1.5%	TLS 1.2	100%
TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xCC, 0xA8)	1.3%	TLS 1.2	100%
TLS_CHACHA20_POLY1305_SHA256 (0x13, 0x03)	0.5%	TLS 1.3	100%
TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xCC, 0xA9)	0.4%	TLS 1.2	100%
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xC0, 0x2C)	0.3%	TLS 1.2	100%
Null ciphers (no encryption of data)		PASS	
Anonymous Ciphers (no authorization)		Reports Error	
Decryption Validation		PASS	
Decryption Bypass Exceptions		PASS	
TLS Session Reuse		PASS	
Threat Prevention		Tested	Blocked
False Positives		55	0
No Background Network Load		977	977
With Background Network Load		977	977
Evasions		35	35
IP Packet Fragmentation		**11**	**11**
8 Byte IP Fragments; reverse order			100%
8 Byte IP Fragments; random order			100%
8 Byte IP Fragments; overlapping duplicate fragments with garbage payloads			100%
8 Byte IP Fragments; interleave chaff sandwich (invalid IP options)			100%
8 Byte IP Fragments; reverse order; delay last fragment			100%
8 Byte IP Fragments; reverse order; overlapping			100%
8 Byte IP Fragments; random order; overlapping			100%
8 Byte IP Fragments; reverse order; overlapping; type-of-service bits value			100%
8 Byte IP Fragments; random order; overlapping; type-of-service bits value			100%
16 Byte IP Fragments; overlapping; random order			100%

16 Byte IP Fragments; overlapping duplicate fragments with garbage payloads		100%
TCP Steam Segmentation	**13**	**13**
3 Byte TCP Segments; reverse order		100%
3 Byte TCP Segments; random order		100%
3 Byte TCP Segments; delay first segment		100%
3 Byte TCP Segments; reverse order; delay last segment		100%
Overlapping 3 Byte TCP Segments		100%
Overlapping 3 Byte TCP Segments; reverse order		100%
Overlapping 3 Byte TCP Segments; random order		100%
Overlapping 3 Byte TCP Segments; duplicate Segments with garbage payloads		100%
3 Byte TCP Segments; interleave chaff after (invalid TCP checksums); delay first segment		100%
3 Byte TCP Segments; random order; interleave chaff before (invalid TCP checksums); delay random segment		100%
3 Byte TCP Segments; random order; interleave chaff sandwich (out-of-window sequence numbers); TCP MSS option		100%
3 Byte TCP Segments; random order; interleave chaff after (requests to resynch sequence numbers mid-stream); TCP window scale option		100%
3 Byte TCP Segments; random order; interleave chaff sandwich (requests to resynch sequence numbers mid-stream); TCP window scale option; delay first segment		100%
Layered Evasions	**8**	**8**
3 Byte TCP Segments; 8 Byte IP Fragments;		100%
3 Byte TCP Segments; 8 Byte IP Fragments; in reverse order		100%
3 Byte TCP Segments; random order; 8 Byte IP Fragments;		100%
3 Byte TCP Segments; 8 Byte IP Fragments; random order		100%
3 Byte TCP Segments; random order; 8 Byte IP Fragments; in reverse order		100%
3 Byte TCP Segments; random order; interleave chaff before (out-of-window sequence numbers); TCP MSS option; Overlapping 8 Byte IP Fragments; reverse order; interleave chaff after (invalid IP options); delay random fragment		100%
3 Byte TCP Segments; interleave chaff sandwich (requests to resynch sequence numbers mid-stream); TCP window scale option; delay first segment; 8 Byte IP Fragments; interleave chaff before (invalid IP options)		100%
Overlapping 3 Byte TCP Segments; overlapping 8 Byte IP Fragments;		100%
3 Byte TCP Segments; 8 Byte IP Fragments; interleave chaff before (invalid IP options)		100%
IP Address Spoofing	**1**	**1**
TCP Split Handshake Spoofing	**1**	**1**
Other Evasions	**1**	**1**
Open TCP session; send small pieces of application protocol headers; pause between pieces		100%
Performance		
Raw Packet Processing Performance (UDP Traffic)	**Mbps**	**Latency (ms)**
64 Byte Frames	221	0.35
128 Byte Frames	282	0.46
256 Byte Frames	516	0.50
512 Byte Frames	990	0.43
1024 Byte Frames	2,014	0.55
1280 Byte Frames	2,796	0.44
1518 Byte Frames	3,295	0.53
HTTP Capacity without delays	**CPS**	**Mbps**
1,000 Connections Per Second - 115.6 KB Response	1,000	1,000
2,000 Connections Per Second - 57.4 KB Response	2,000	1,000

4,000 Connections Per Second - 28.0 KB Response	4,000	1,000
8,000 Connections Per Second - 13.5 KB Response	8,000	1,000
16,000 Connections Per Second - 6.4 KB Response	16,000	1,000
32,000 Connections Per Second - 2.7 KB Response	31,990	1,000
HTTP Capacity with delays	**CPS**	**Mbps**
1,000 Connections Per Second - 115.6 KB Response	1,000	1,000
2,000 Connections Per Second - 57.4 KB Response	1,999	1,000
4,000 Connections Per Second - 28.0 KB Response	4,000	1,000
8,000 Connections Per Second - 13.5 KB Response	8,008	1,000
16,000 Connections Per Second - 6.4 KB Response	16,000	1,000
32,000 Connections Per Second - 2.7 KB Response	32,000	1,000
HTTPS Capacity (0x13, 0x01)	**CPS**	**Mbps**
1,000 Connections Per Second - 113.8 KB Response	1,000	1,000
2,000 Connections Per Second - 54.9 KB Response	1,997	999
4,000 Connections Per Second - 25.7 KB Response	3,998	1,000
8,000 Connections Per Second - 11.2 KB Response	8,001	1,000
16,000 Connections Per Second - 3.9 KB Response	14,100	881
32,000 Connections Per Second - 0.2 KB Response	14,000	438
HTTPS Capacity (0x13, 0x02)	**CPS**	**Mbps**
1,000 Connections Per Second - 113.8 KB Response	999	999
2,000 Connections Per Second - 54.9 KB Response	1,998	999
4,000 Connections Per Second - 25.7 KB Response	3,997	999
8,000 Connections Per Second - 11.2 KB Response	7,998	1,000
16,000 Connections Per Second - 3.9 KB Response	14,000	875
32,000 Connections Per Second - 0.2 KB Response	15,000	469
HTTPS Capacity (0xC0, 0x2F)	**CPS**	**Mbps**
1,000 Connections Per Second - 115.0 KB Response	1,000	1,000
2,000 Connections Per Second - 56.3 KB Response	2,000	1,000
4,000 Connections Per Second - 27.0 KB Response	3,991	998
8,000 Connections Per Second - 12.3 KB Response	8,002	1,000
16,000 Connections Per Second - 5.0 KB Response	14,300	894
32,000 Connections Per Second - 1.4 KB Response	15,400	481
HTTPS Capacity (0xC0, 0x30)	**CPS**	**Mbps**
1,000 Connections Per Second - 115.0 KB Response	1,000	1,000
2,000 Connections Per Second - 56.3 KB Response	2,000	1,000
4,000 Connections Per Second - 27.0 KB Response	3,992	998
8,000 Connections Per Second - 12.3 KB Response	8,001	1,000
16,000 Connections Per Second - 5.0 KB Response	14,000	875
32,000 Connections Per Second - 1.4 KB Response	15,800	494

Stability and Reliability	Result
Protocol Fuzzing & Mutation	PASS
Blocking with Minimal Load	PASS
Blocking Under Load	PASS
Attack Detection/Blocking – Normal Load	PASS
State Preservation – Normal Load	PASS
Pass Legitimate Traffic – Normal Load	PASS
State Preservation – Maximum Exceeded	PASS
Drop Traffic – Maximum Exceeded	PASS

Appendix B

CyberRatings Classification Matrix	
RATING	DEFINITION
AAA	A product rated 'AAA' has the highest rating assigned by CyberRatings. The product's capacity to meet its commitments to consumers is extremely strong.
AA	A product rated 'AA' differs from the highest-rated products only to a small degree. The product's capacity to meet its commitments to consumers is very strong.
A	A product rated 'A' is somewhat less capable than higher-rated categories. However, the product's capacity to meet its commitments to consumers is still strong.
BBB	A product rated 'BBB' exhibits adequate stability and reliability. However, previously unseen events and use cases are more likely to negatively impact the product's capacity to meet its commitments to consumers.
	A product rated 'BB,' 'B,' 'CCC,' 'CC,' and 'C' is regarded as having significant risk characteristics. 'BB' indicates the least degree of risk and 'C' the highest. While such products will likely have some specialized capability and features, these may be outweighed by large uncertainties or major exposure to adverse conditions.
BB	A product rated 'BB' is more susceptible to failures than products that have received higher ratings. The product has the capacity to meet its commitments to consumers. However, it faces minor technical limitations that have a potential to be exposed to risks.
B	A product rated 'B' is more susceptible to failures than products rated 'BB'; however, it has the minimum capacity. Adverse conditions will likely expose the product's technical limitations that lead to an inability to meet its commitments to consumers.
CCC	A product rated 'CCC' is susceptible to failures and is dependent upon favorable conditions to perform expected functions. In the event of adverse conditions, the product is not likely to have the capacity to meet its commitments to consumers.
CC	A product rated 'CC' is highly susceptible to failures. The 'CC' rating is used when a failure has not yet occurred, but CyberRatings considers it a virtual certainty.
C	A product rated 'C' is highly susceptible to failures. The product is expected to fail under any abnormal operating conditions and does not offer a useful management systems and logging information compared with products that are rated higher.
D	A product rated 'D' is actively underperforming and failing and does not meet the use-case. The 'D' rating is used when the product is not operational without a major technical overhaul. Unless CyberRatings believes that such technical fixes will be made within a stated grace period (typically 30-90 calendar days), the 'D' rating also is an indicator that existing customers using the product have already experienced a failure and should take immediate action.

Authors

Thomas Skybakmoen, Vikram Phatak, Ahmed Basheer

Contact Information

CyberRatings.org
2303 Ranch Road 620 South
Suite 160, #501
Austin, TX 78734

info@cyberratings.org
www.cyberratings.org