CYBER RATINGS.ORG

TEST METHODOLOGY: SPOT CHECK

Security Service Edge (SSE)

January 22, 2024

v1.0

Table of Contents

1 Security Service Edge (SSE)

Security Service Edge (SSE) solutions leverage the cloud's scalability, flexibility, and operational benefits to deliver security – Access Control, Authentication and Identity, Data Loss Prevention (DLP), DNS Protection, Encryption (SSL/TLS), Exploit Detection and Prevention, Malware and Phishing protection (including via Browser Isolation), Cloud Access / Application Control (CASB), and the ability to implement Zero Trust Network Access (ZTNA).

This test methodology is designed to assess the capabilities for Security Service Edge (SSE): **Threat Prevention**: Exploit Prevention, Malware Prevention, and Resistance to Evasions.

2 TLS/SSL Support

To address the growing threat of focused attacks using the most common web protocols and applications, CyberRatings tests the capabilities of SSE offerings to support a range of cipher suites and provide visibility into the encrypted payloads to detect attacks concealed by encryption as well as attacks against the encryption protocols themselves.

2.1 Cipher Suite Support

To provide visibility into potential threats that are encrypted using TLS/SSL, the SSE is expected to support a wide range of commonly used cipher suites. Cipher suites are selected based on the published current frequency of use[1] and security status[2].

Table 1 – Selected Cipher Suites[3]

Version	(Value)	Cipher Suites	Prevalence
TLS 1.3	(0x13, 0x02)	TLS_AES_256_GCM_SHA384	66.51%
TLS 1.2	(0xC0, 0x30)	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	11.85%
TLS 1.2	(0xC0, 0x2F)	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	9.26%
TLS 1.3	(0x13, 0x01)	TLS_AES_128_GCM_SHA256	8.07%
TLS 1.2	(0xCC, 0xA8)	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	1.72%
TLS 1.2	(0xC0, 0x28)	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	0.68%
TLS 1.3	(0x13, 0x03)	TLS_CHACHA20_POLY1305_SHA256	0.55%
TLS 1.2	(0xC0, 0x2C)	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	0.42%
TLS 1.2	(0xCC, 0xA9)	TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256	0.27%
TLS 1.2	(0xC0, 0x2B)	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	0.20%

2.1.1 2.1.1 Current Cipher Suites

Test Objective: To what extent does the SSE support current cipher suites?

Test Approach: Testing will determine which cipher suites are supported. Tested cipher suites are selected based on frequency of use and security recommendations from reputable sources. The cipher suites available for this test include those listed in Table 1. Since this test is minimally invasive, it will be randomly repeated during the test period with multiple SSE instances to confirm standard availability.

[1] Published international daily cipher suite usage can be found at https://crawler.ninja/files/ciphers.txt
[2] A list of cipher suites and associated attributes including security ratings can be found at https://ciphersuite.info/cs/
[3] Cipher suite descriptions and associated value codes for testing are from https://www.iana.org/assignments/tls-parameters/tls-parameters.xhtml.

3 Threat Protection

CyberRatings' threat prevention tests assess how accurately the SSE blocks and logs threats while remaining resistant to false positives. To accomplish this goal, the SSE will be deployed using vendor-recommended settings. Protection being tested must be available to customers at the time of testing.

3.1 False Positives

False positives are any legitimate, non-malicious traffic that the SSE perceives as malicious and blocks. The ability to correctly identify and allow legitimate traffic while maintaining protection against attacks is a key to effective protection. CyberRatings false positive tests examine the ability of the SSE to block attacks while permitting legitimate traffic. The rate at which SSEs block legitimate traffic will be recorded.

3.1.1 Ongoing check – legitimate traffic, documents, and files

Since SSE is a cloud offering which in some cases utilizes machine learning to modify/tune settings in real-time, testing for false positives requires legitimate traffic and documents to be included when testing the SSE offering's ability to block attacks. CyberRatings will introduce legitimate traffic, documents, and files into tests in sections 3.2 and Error! Reference source not found.. Testing may include but is not limited to the following file formats: HTML, .exe, .jar, .xlsm, .css, .pdf, .ppt, .pptx, .doc, .docx, .zip, 7zip, gzip, .DLL, .js, .xls, .xlsx, .chm, .rar, .lnk, .cur, .tar, .xrc.

3.2 Exploits

An exploit is an attack that takes advantage of a vulnerability in a protocol, product, operating system, or application. CyberRatings verifies that the SSE is capable of detecting and blocking exploits while remaining resistant to false positives by attempting to send exploits through the SSE and verifying that the malicious traffic is blocked.

The CyberRatings exploit repository contains thousands of exploits over 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.

Spot Check verifies that an SSE can block exploits while remaining resistant to false positives.

3.2.1 Exploits Delivered Over HTTP

Testing will determine if exploits delivered over HTTP are blocked by the SSE. All tests are performed with false positive traffic to ensure the SSE is not blocking all traffic. (*see section 3.1**Error! Reference source not found.***)

3.2.2 Exploits Delivered over HTTPS

Testing will determine if exploits delivered over HTTPS are blocked by the SSE. All tests are performed with false positive traffic to ensure the SSE is not blocking all traffic. (*see section 3.1*). Testing will be performed using the following cipher suites

Version	(Value)	Cipher Suites	Prevalence
TLS 1.3	(0x13, 0x02)	TLS_AES_256_GCM_SHA384	66.51%
TLS 1.2	(0xC0, 0x30)	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	11.85%
TLS 1.2	(0xC0, 0x2F)	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	9.26%
TLS 1.3	(0x13, 0x01)	TLS_AES_128_GCM_SHA256	8.07%

3.3 Malware Protection

Users may be deceived into clicking on a malicious link, such as a web page or a banner advertisement, that will download and execute malware. In cases where an attacker is aiming for a large number of victims, the attacker may hijack widely used reputable websites to distribute the malware.

The SSE model is one of shared responsibility. The vendor manages the version control of all hardware, software, firmware, and security/protection updates and provides vendor-recommended settings, while the customer manages access control and certain aspects of the security policy.

Spot Check verifies that an SSE can block malware while remaining resistant to false positives.

3.3.1 Malware Downloads Delivered Over HTTP

Testing will determine if malware downloads delivered over HTTP are blocked by the SSE. All tests are performed with false positive traffic to ensure the SSE is not blocking all traffic. (*see section 3.1*)

3.3.2 Malware Downloads Delivered over HTTPS

Testing will determine if malware downloads delivered over HTTPS are blocked by the SSE. All tests are performed with false positive traffic to ensure the SSE is not blocking all traffic. (*see section 3.1*). Testing will be performed using the following cipher suites:

Version	(Value)	Cipher Suites	Prevalence
TLS 1.3	**(**0x13, 0x02**)**	TLS_AES_256_GCM_SHA384	66.51%
TLS 1.2	**(**0xC0, 0x30**)**	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	11.85%
TLS 1.2	**(**0xC0, 0x2F**)**	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	9.26%
TLS 1.3	**(**0x13, 0x01**)**	TLS_AES_128_GCM_SHA256	8.07%

3.4 Evasions

Threat actors use evasion techniques to disguise and modify attacks at the point of delivery to avoid detection by security products. It is imperative that a firewall correctly handles evasions since just one successful evasion technique can enable an attacker to compromise systems undetected.

CyberRatings first verifies that the firewall detects and blocks a collection of baseline exploits and malware. Next, CyberRatings applies evasion techniques to the baseline exploits and malware and validates the execution of the exploit's payload or the malware's delivery. Wherever possible, the firewall is expected to successfully normalize the evaded traffic to provide an accurate alert relating to the original attack, rather than alerting purely on anomalous traffic detected because of the evasion technique itself.

3.4.1 HTTP Evasions

3.4.1.1 HTTP Headers

An attacker can send bogus, duplicate, or malformed headers to alter the parsing of HTTP messages by the receiving application.

Examples of header evasions include but are not limited to:

- Declare HTTP/0.9 and send headers
- Send two Transfer-Encoding headers where one is given a junk value

3.4.1.2 HTTP Chunked Encoding

Chunked encoding allows a client or server to break an HTTP message body into small chunks and transmit them individually. The sender needs only to specify the size of each chunk before it is transmitted and then indicate when the last chunk has been transmitted. Since chunked encoding intersperses arbitrary numbers (chunk sizes) with the elements of the original message body, it can significantly change the content's appearance as observed

"on the wire" during transmission. In addition, the sender can break the message into chunks at arbitrary points. This makes it difficult to reliably identify the original content from the raw data on the network.

Examples of chunked encoding include but are not limited to:

- Chunked stream with small chunk sizes
- Chunked stream with malformed trailer sections

3.4.1.3 HTTP Compression

Per RFC 2616, the HTTP protocol allows clients and servers to use several compression methods. These compression methods not only improve performance in many circumstances, but they also completely change the characteristic size and appearance of served content.

Examples of compression include but are not limited to:

- br – Brotli format defined in RFC 7932 and implemented in all major modern browsers
- deflate – Deflate format defined in RFC 1951 and implemented in all major modern browsers
- gzip – GNU zip format defined in RFC 1952 and implemented in all major modern browsers

3.4.2 HTML

HTML is the standard markup used for creating web pages and web applications. HTML's feature rich markup allows attackers to encode malicious web pages in various formats.

Examples of HTML evasions include but are not limited to:

- Unicode encoding (UTF-8, UTF-16, UTF-16LE, UTF-16BE, UTF-7)
- Byte Order Mark (BOM)
- Padding
- EICAR string included at top of the HTML

3.4.3 XML

XML is a format that can be used for data exchange or for creating web pages and web applications.

Examples of XML evasions include but are not limited to:

- Unicode encoding (UTF-8, UTF-16, UTF-16LE, UTF-16BE, UTF-7)
- Byte Order Mark (BOM)
- Padding

3.4.4 JSON

JSON is a popular format for data exchange in REST APIs. It is used to structure and transmit data between clients and servers in a lightweight and human-readable manner. JSON supports multiple Unicode encodings and character encoding escape sequences. An attacker can use the encodings to hide attacks in a format vastly different from the original data.

Examples of JSON evasions include but are not limited to:

- Unicode encoding (UTF-8, UTF-16, UTF-16LE, UTF-16BE, UTF-7)
- Byte Order Mark (BOM)
- Padding
- Unicode escape of characters

3.4.5 Multipart/form-data

Multipart/form-data is a content type typically used when sending web form data to the server for submission. Ambiguities in the format may allow an attacker to obfuscate transferred data.

Examples of form-data evasions include but are not limited to:

- Padding
- Spaces between attribute value and '='

3.4.6 Portable Executable

Portable Executable (PE) is the primary file format for Windows programs. The flexibility and capabilities of the PE format make it a critical file type for attackers to package as malware.

Examples of PE evasions include but are not limited to:

- Packed (VMProtect, UPX, ASPack, etc.)
- Archived (7z, zip, rar, etc.)
- Malformed PE header
- Section name modifications
- Code-signed

3.4.7 Layered Evasions

These tests determine the effectiveness of the firewall when subjected to combinations of evasion techniques, both within a single protocol and across multiple protocols.

Examples of layered evasions include but are not limited to:

- Combination of multiple TCP evasions
- Combination of IP and TCP evasions
- Combination of IP, TCP, TLS, HTTP, HTML evasions

Contact Information

CyberRatings.org

2303 Ranch Road 620 South

Suite 160, #501

Austin, TX 78734

info@cyberratings.org

www.cyberratings.org