YourIT!
Your Logo Goes Here

Technical Review

External Vulnerability Scan Results

Prepared for: Client Company

Prepared by: YourIT Company

YourIT!
Your Logo Goes Here

Table of Contents

1 - Summary

This report gives details on hosts that were tested and issues that were found during the External Vulnerability Scan. The findings are grouped by category.

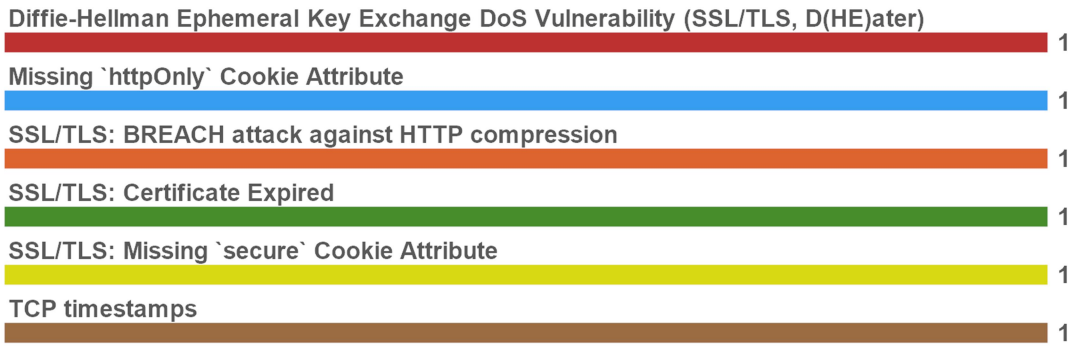
Appliances Used:
1. EVS-TFCZ29

Host Issue Summary

HOST	OPEN PORTS	HIGH	MED	LOW	FALSE	HIGHEST CVSS
96.68.119.142(96-68-119-142-static.atl.earthlink.net)[1]	3	1	4	1	0	7.5
Total: 1	3	1	4	1	0	7.5

Issues by Severity

Issues by NVT



ISSUE	COUNT
TCP timestamps	1
Missing `httpOnly` Cookie Attribute	1

ISSUE	COUNT
SSL/TLS: Certificate Expired	1
SSL/TLS: BREACH attack against HTTP compression	1
SSL/TLS: Missing `secure` Cookie Attribute	1
Diffie-Hellman Ephemeral Key Exchange DoS Vulnerability (SSL/TLS, D(HE)ater)	1

2 - Scan Details

This section details the issues discovered in order of severity. For each issue, the affected nodes are also listed.

Issues by Severity

2.1 - Diffie-Hellman Ephemeral Key Exchange DoS Vulnerability (SSL/TLS, D(HE)ater)

H

HIGH: (CVSS: 7.5)

OID: 1.3.6.1.4.1.25623.1.0.117840

443/TCP (HTTPS)

Summary
The remote SSL/TLS server is supporting Diffie-Hellman ephemeral (DHE) Key Exchange algorithms and thus could be prone to a denial of service (DoS) vulnerability.

Affected Nodes
96.68.119.142(96-68-119-142-static.atl.earthlink.net)[1]

Vulnerability Detection Result
'DHE' cipher suites accepted by this service via the TLSv1.2 protocol: TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 TLS_DHE_RSA_WITH_AES_256_CBC_SHA TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 TLS_DHE_RSA_WITH_AES_256_CCM TLS_DHE_RSA_WITH_AES_256_CCM_8 TLS_DHE_RSA_WITH_AES_256_GCM_SHA384

Solution
'- DHE key exchange should be disabled if no other mitigation mechanism can be used and either elliptic-curve variant of Diffie-Hellman (ECDHE) or RSA key exchange is supported by the clients. The fact that RSA key exchange is not forward secret should be considered. - Limit the maximum number of concurrent connections in e.g. the configuration of the remote server. For Postfix this limit can be configured via 'smtpd_client_new_tls_session_rate_limit' option, for other products please refer to the manual of the product in question on configuration possibilities.

Vulnerability Insight
The Diffie-Hellman Key Agreement Protocol allows remote attackers (from the client side) to send arbitrary numbers that are actually not public keys, and trigger expensive server-side DHE modular-exponentiation calculations, aka a D(HE)ater attack. The client needs very little CPU resources and network bandwidth. The attack may be more disruptive in cases where a client can require a server to select its largest supported key size. The basic attack scenario is that the client must claim that it can only communicate with DHE, and the server must be configured to allow DHE.

Vulnerability Detection Method
Checks the supported cipher suites of the remote SSL/TLS server.Details:Diffie-Hellman Ephemeral Key Exchange DoS Vulnerability (SSL/TLS, D(HE)ater)(OID: 1.3.6.1.4.1.25623.1.0.117840)Version used: 2021-12-17T14:03:21Z

References
https://www.researchgate.net/profile/Anton-Stiglic-2/publication/2401745_Security_Issues_in_the_Diffie-Hellman_Key_Agreement_Protocol,https://github.com/Balasys/dheater

2.2 - SSL/TLS: Missing `secure` Cookie Attribute

M	MEDIUM: (CVSS: 6.4) OID: 1.3.6.1.4.1.25623.1.0.902661	443/TCP (HTTPS)

Summary
The host is running a server with SSL/TLS and is prone to information disclosure vulnerability.

Affected Nodes
96.68.119.142(96-68-119-142-static.atl.earthlink.net)[1]

Vulnerability Detection Result
The cookies: Set-Cookie: PHPSESSID=***replaced***; path=/ are missing the "secure" attribute.

Solution
Set the 'secure' attribute for any cookies that are sent over a SSL/TLS connection.

Vulnerability Insight
The flaw is due to cookie is not using 'secure' attribute, which allows cookie to be passed to the server by the client over non-secure channels (http) and allows attacker to conduct session hijacking attacks.

Vulnerability Detection Method
Details:SSL/TLS: Missing `secure` Cookie Attribute(OID: 1.3.6.1.4.1.25623.1.0.902661)Version used: 2021-08-06T11:34:45Z

References
https://www.owasp.org/index.php/SecureFlag,http://www.ietf.org/rfc/rfc2965.txt,https://www.owasp.org/index.php/Testing_for_cookies_attributes_(OWASP-SM-002)

2.3 - SSL/TLS: BREACH attack against HTTP compression

M	MEDIUM: (CVSS: 5.9) OID: 1.3.6.1.4.1.25623.1.0.117414	443/TCP (HTTPS)

Summary
SSL/TLS connections are vulnerable to the 'BREACH' (Browser Reconnaissance & Exfiltration via Adaptive Compression of Hypertext) attack.

Affected Nodes
96.68.119.142(96-68-119-142-static.atl.earthlink.net)[1]

Vulnerability Detection Result
Based on the following information it was determined that the remote web server has HTTP compression enabled: HTTP headers : Content-Encoding: gzip URL : https://96-67-119-196-static.hfc.comcastbusiness.net/

Impact
The flaw makes it easier for man-in-the-middle attackers to obtain plaintext secret values.

Solution
The following mitigation possibilities are available: 1. Disabling HTTP compression 2. Separating secrets from user input 3. Randomizing secrets per request 4. Masking secrets (effectively randomizing by XORing with a random secret per request) 5. Protecting vulnerable pages with CSRF 6. Length hiding (by adding random number of bytes to the responses) 7. Rate-limiting the requests Note: The mitigations are ordered by effectiveness (not by their practicality - as this may differ from one application to another).

Vulnerability Insight
Angelo Prado, Neal Harris and Yoel Gluck reported that SSL/TLS attacks are still viable via a 'BREACH' (Browser Reconnaissance & Exfiltration via Adaptive Compression of Hypertext) attack, which they describe as: While CRIME was mitigated by disabling TLS/SPDY compression (and by modifying gzip to allow for explicit separation of compression contexts in SPDY), BREACH attacks HTTP responses. These are compressed using the common HTTP compression, which is much more common than TLS-level compression. This allows essentially the same attack demonstrated by Duong and Rizzo, but without relying on TLS-level compression (as they anticipated). It is important to note that the attack is agnostic to the version of TLS/SSL, and does not require TLS-layer compression. Additionally, the attack works against any cipher suite. Against a stream cipher, the attack is simpler: The difference in sizes across response bodies is much more granular in this case. If a block cipher is used, additional work must be done to align the output to the cipher text blocks.

Vulnerability Detection Method
Checks if the remote web server has HTTP compression enabled. Note: Even with HTTP compression enabled the web application hosted on the web server might not be vulnerable. The low Quality of Detection (QoD) of this VT reflects this fact.Details:SSL/TLS: BREACH attack against HTTP compression(OID: 1.3.6.1.4.1.25623.1.0.117414)Version used: 2021-08-24T09:01:06Z

References
http://breachattack.com/,http://www.kb.cert.org/vuls/id/987798,http://www.openwall.com/lists/oss-security/2013/08/07/1,https://bugzilla.redhat.com/show_bug.cgi?id=995168,https://en.wikipedia.org/wiki/HTTP_compression

2.4 - Missing `httpOnly` Cookie Attribute

M	MEDIUM: (CVSS: 5) OID: 1.3.6.1.4.1.25623.1.0.105925	443/TCP (HTTPS)

Summary
The application is missing the 'httpOnly' cookie attribute

Affected Nodes
96.68.119.142(96-68-119-142-static.atl.earthlink.net)[1]

Vulnerability Detection Result
The cookies: Set-Cookie: PHPSESSID=***replaced***; path=/ are missing the "httpOnly" attribute.

Solution
Set the 'httpOnly' attribute for any session cookie.

Vulnerability Insight
The flaw is due to a cookie is not using the 'httpOnly' attribute. This allows a cookie to be accessed by JavaScript which could lead to session hijacking attacks.

Vulnerability Detection Method
Check all cookies sent by the application for a missing 'httpOnly' attributeDetails:Missing `httpOnly` Cookie Attribute(OID: 1.3.6.1.4.1.25623.1.0.105925)Version used: 2020-08-24T15:18:35Z

References
https://www.owasp.org/index.php/HttpOnly,https://www.owasp.org/index.php/Testing_for_cookies_attributes_(OTG-SESS-002)

2.5 - SSL/TLS: Certificate Expired

MEDIUM: (CVSS: 5)
OID: 1.3.6.1.4.1.25623.1.0.103955
443/TCP (HTTPS)

Summary
The remote server's SSL/TLS certificate has already expired.

Affected Nodes
96.68.119.142(96-68-119-142-static.atl.earthlink.net)[1]

Vulnerability Detection Result
The certificate of the remote service expired on 2017-06-23 00:29:25. Certificate details: fingerprint (SHA-1) | 83ED1CC41839631FD88B3D4B52FC75E5421C5646 fingerprint (SHA-256) | 88A06C2D7AA791ECCF2C20E14396A389F29FB9670584B4DC44DC35BA00C06B3B issued by | CN=pfSense-4effa8e5cdaa8,O=pfSense webConfigurator Self-Signed Certificate public key size (bits) | 2048 serial | 00 signature algorithm | sha256WithRSAEncryption subject | CN=pfSense-4effa8e5cdaa8,O=pfSense webConfigurator Self-Signed Certificate subject alternative names (SAN) | pfSense-4effa8e5cdaa8 valid from | 2012-01-01 00:29:25 UTC valid until | 2017-06-23 00:29:25 UTC

Solution
Replace the SSL/TLS certificate by a new one.

Vulnerability Insight
This script checks expiry dates of certificates associated with SSL/TLS-enabled services on the target and reports whether any have already expired.

Vulnerability Detection Method
Details:SSL/TLS: Certificate Expired(OID: 1.3.6.1.4.1.25623.1.0.103955)Version used: 2021-11-22T15:32:39Z

2.6 - TCP timestamps

LOW: (CVSS: 2.6)
OID: 1.3.6.1.4.1.25623.1.0.80091

Summary
The remote host implements TCP timestamps and therefore allows to compute the uptime.

Affected Nodes
96.68.119.142(96-68-119-142-static.atl.earthlink.net)[1]

Vulnerability Detection Result
It was detected that the host implements RFC1323/RFC7323. The following timestamps were retrieved with a delay of 1 seconds in-between: Packet 1: 1785064370 Packet 2: 1271867096

Impact
A side effect of this feature is that the uptime of the remote host can sometimes be computed.

Solution
To disable TCP timestamps on linux add the line 'net.ipv4.tcp_timestamps = 0' to /etc/sysctl.conf. Execute 'sysctl -p' to apply the settings at runtime. To disable TCP timestamps on Windows execute 'netsh int tcp set global timestamps=disabled' Starting with Windows Server 2008 and Vista, the timestamp can not be completely disabled. The default behavior of the TCP/IP stack on this Systems is to not use the Timestamp options when initiating TCP connections, but use them if the TCP peer that is initiating communication includes them in their synchronize (SYN) segment. See the references for more information.

Vulnerability Insight
The remote host implements TCP timestamps, as defined by RFC1323/RFC7323.

Vulnerability Detection Method
Special IP packets are forged and sent with a little delay in between to the target IP. The responses are searched for a timestamps. If found, the timestamps are reported.Details:TCP timestamps(OID: 1.3.6.1.4.1.25623.1.0.80091)Version used: 2020-08-24T08:40:10Z

References
http://www.ietf.org/rfc/rfc1323.txt,http://www.ietf.org/rfc/rfc7323.txt,https://web.archive.org/web/20151213072445/http://www.microsoft.com/en-us/download/details.aspx?id=9152