VULNERABILITY REPORT OF PRACTICE WEBSITE

TEAM – 2.6 , TOPIC – MALWARE DETECTION AND CLASSIFICATION

1. Vulnerability Name: CGI Generic SQL Injection(blind)

Risk Factor: High

CWE: 20, 77, 89, 91, 203, 643, 713, 722, 727, 751, 801, 810, 928, 929

OSWAP CATEGORY: A03:2021- Injection

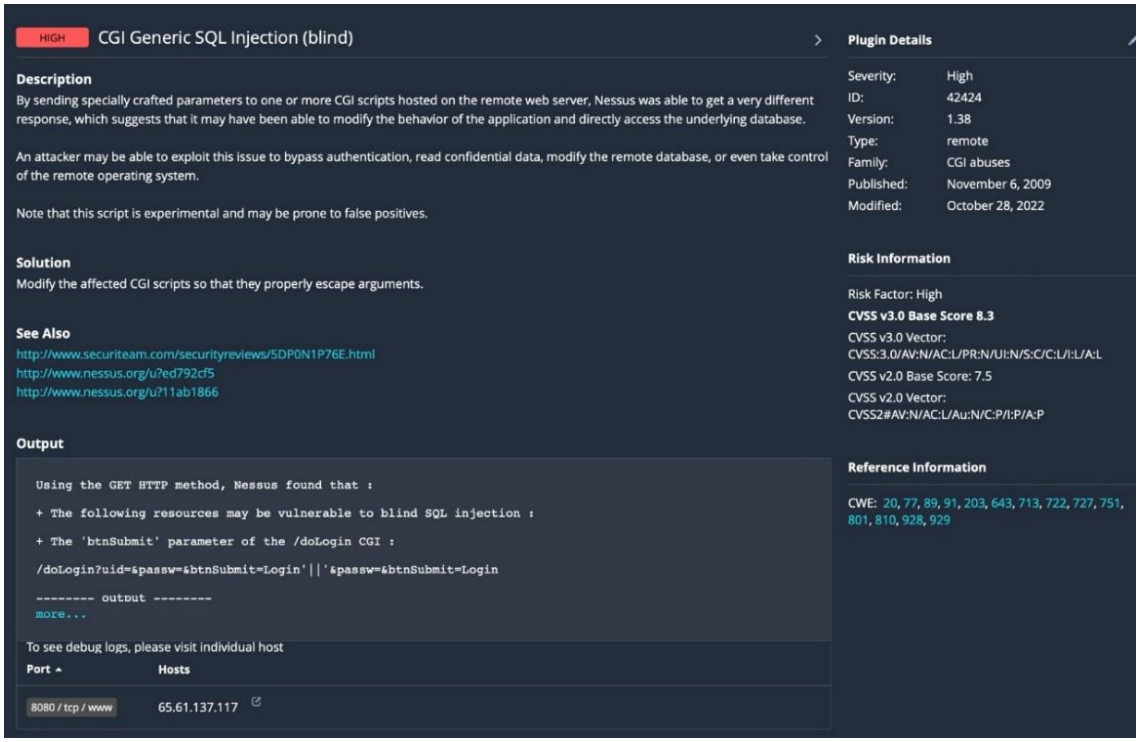
DESCRIPTION: By sending specially crafted parameters to one or more CGI scripts hosted on the remote web server, Nessus was able to get a very different response, which suggests that it may have been able to modify the behavior of the application and directly access the underlying database.
An attacker may be able to exploit this issue to bypass authentication, read confidential data, modify the remote database, or even take control of the remote operating system.
Note that this script is experimental and may be prone to false positives.

Vulnerability Path: http://testfire.net/

Business Impact: An attacker could exploit this vulnerability to:

- Bypass authentication and gain access to sensitive data, such as customer records, financial information, or trade secrets.
- Modify or delete data in the database, which could disrupt business operations or lead to financial losses.
- Take control of the web server or underlying operating system, which could allow them to launch further attacks or disrupt the business's online operations.

Recommendation: Modify the affected CGI scripts so that they properly escape arguments.



2. Vulnerability Name: CGI Generic Cookie Injection Scripting

Risk Factor: Medium

CWE: 477, 642, 715, 722

OSWAP CATEGORY: A03:2021- Injection

DESCRIPTION: The remote web server hosts at least one CGI script that fails to adequately sanitize request strings with malicious JavaScript.
By leveraging this issue, an attacker may be able to inject arbitrary cookies. Depending on the structure of the web application, it may be possible to launch a 'session fixation' attack using this mechanism.
Please note that:

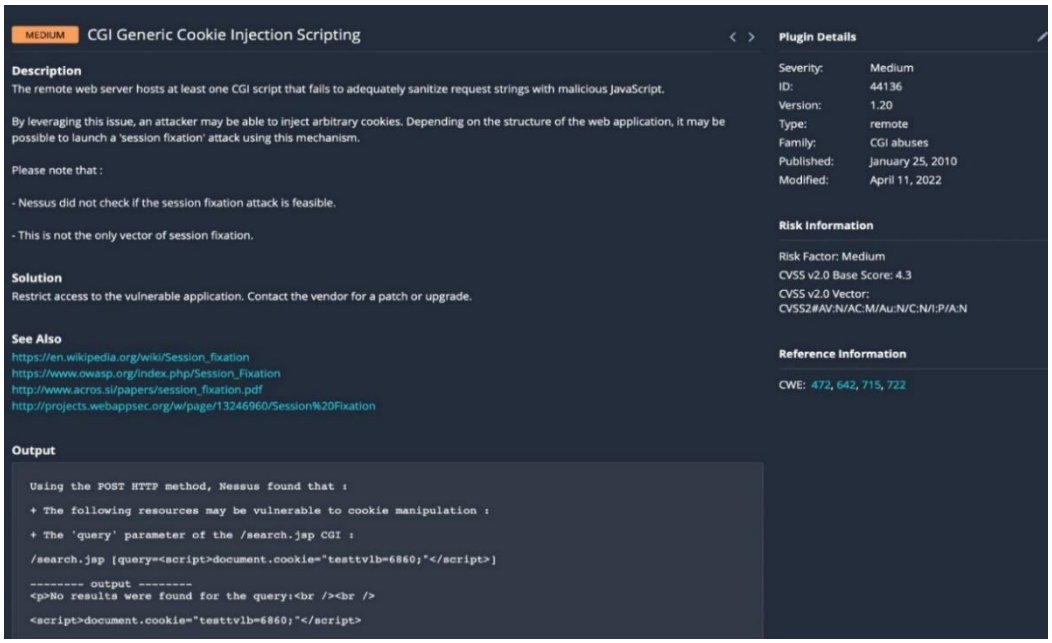
-Nessus did not check if the session fixation attack is feasible.

-This is not the only vector of session fixation.

Vulnerability Path: http://testfire.net/

Business Impact: An attacker could exploit this vulnerability to:

- **Steal session cookies and impersonate users:-** This could allow the attacker to access users' accounts, steal sensitive data, or make unauthorized transactions.
- **Launch session fixation attacks**:- This could allow the attacker to force a victim to use a predetermined session cookie, which could allow the attacker to impersonate the victim or steal their data.
- **Inject malicious code into cookies:-** This could allow the attacker to execute arbitrary code on the victim's browser, which could lead to data theft, account takeover, or malware infection.
- **Recommendation:** Restrict access to the vulnerable application. Contact the vendor for a patch or upgrade.



3. Vulnerability Name: CGI Generic HTML Injection(quick test)

Risk Factor: Medium

CWE: 80, 86

OSWAP CATEGORY: A03:2021- Injection

DESCRIPTION: The remote web server hosts CGI scripts that fail to adequately sanitize request strings with malicious JavaScript. By leveraging this issue, an attacker may be able to cause arbitrary HTML to be executed in a user's browser within the security context of the affected site.
The remote web server may be vulnerable to IFRAME injections or cross-site scripting attacks:
-IFRAME injections allow 'virtual defacement' that might scare or anger gullible users. Such injections are sometimes implemented for 'phishing attacks.
-XSS are extensively tested by four other scripts.

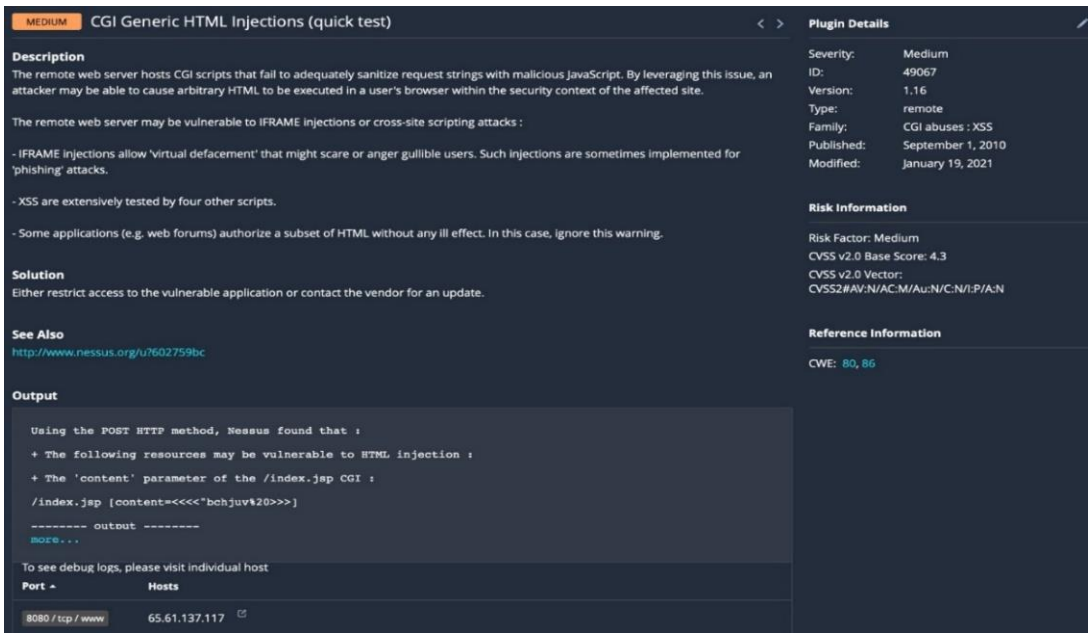
-Some applications (e.g. web forums) authorize a subset of HTML without any ill effect. In this case, ignore this warning.

Vulnerability Path: http://testfire.net/

Business Impact: An attacker could exploit this vulnerability to:

- **Inject malicious HTML into web pages:-** This could allow the attacker to deface the website, redirect users to malicious websites, or steal their data.
- **Launch cross-site scripting (XSS) attacks**:- This could allow the attacker to steal session cookies, hijack user accounts, or execute arbitrary code on the victim's browser.
- **Phish users:-** The attacker could inject malicious HTML into web pages to create phishing forms that look like legitimate login forms. If a user enters their credentials into a phishing form, the attacker can steal them.

Recommendation: Either restrict access to the vulnerable application or contact the vendor for an update.



4. Vulnerability Name: CGI Generic XSS(quick test)

Risk Factor: Medium

CWE: 20, 74, 79, 80, 81, 83, 86, 116, 442, 692, 712, 722, 725, 751, 801, 811, 928, 931

OSWAP CATEGORY: A03:2021- Injection and A08:2021- Software and Data Integrity Failures

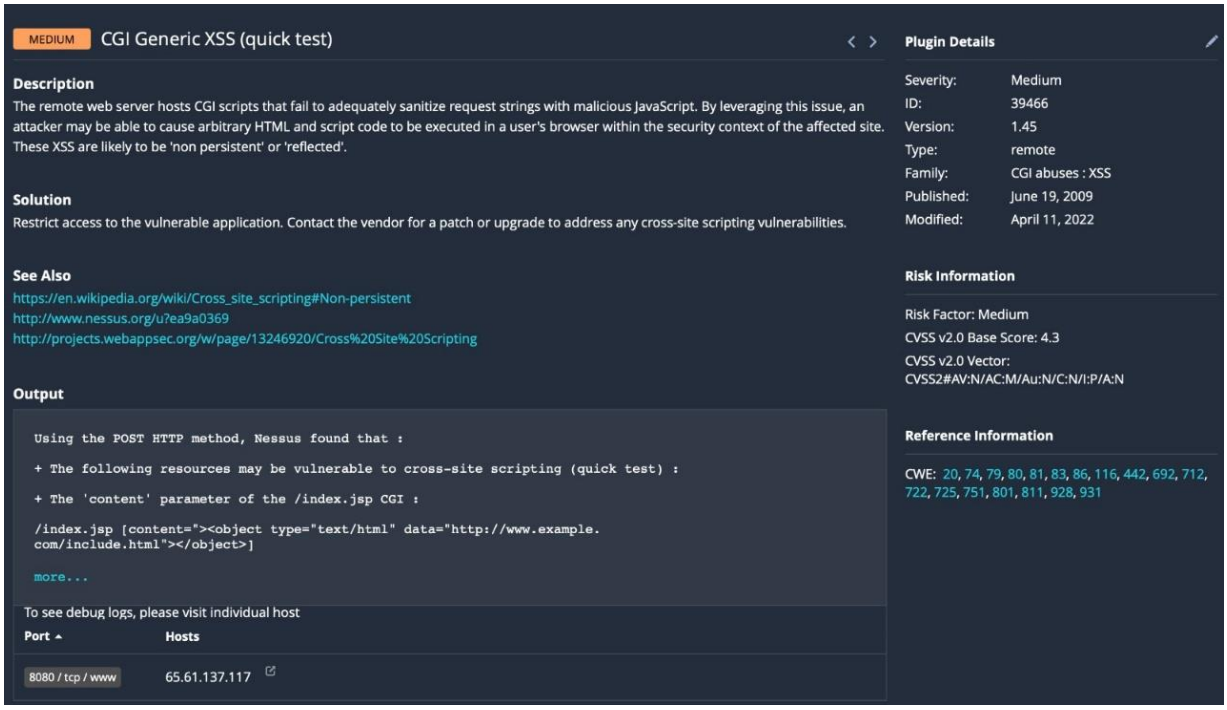
DESCRIPTION: By sending specially crafted parameters to one or more CGI scripts hosted on the remote web server, Nessus was able to get a very different response, which suggests that it may have been able to modify the behavior of the application and directly access the underlying database.

An attacker may be able to exploit this issue to bypass authentication, read confidential data, modify the remote database, or even take control of the remote operating system.
Note that this script is experimental and may be prone to false positives.
Vulnerability Path: http://testfire.net/

Business Impact: An attacker could exploit this vulnerability to:

- **Inject malicious HTML into web pages:-** This could allow the attacker to deface the website, redirect users to malicious websites, or steal their data.
- **Launch cross-site scripting (XSS) attacks**:- This could allow the attacker to steal session cookies, hijack user accounts, or execute arbitrary code on the victim's browser.
- **Phish users:-** The attacker could inject malicious HTML into web pages to create phishing forms that look like legitimate login forms. If a user enters their credentials into a phishing form, the attacker can steal them.

Recommendation: Restrict access to the vulnerable application. Contact the vendor for a patch or upgrade to address any cross-site scripting vulnerabilities.



5. Vulnerability Name: Web Application Potentially Vulnerable to Clickjacking

Risk Factor: Medium

CWE: 693

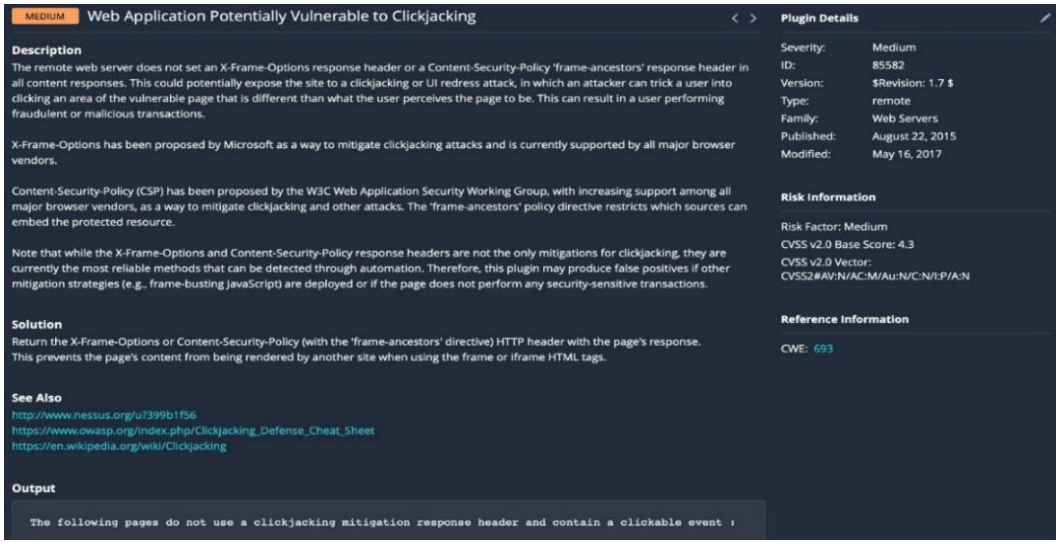
OSWAP CATEGORY: A05:2021- Security Misconfiguration

DESCRIPTION: The remote web server does not set an X-Frame-Options response header or a Content-Security-Policy 'frame-ancestors' response header in all content responses. This could potentially expose the site to a clickjacking or Ul redress attack, in which an attacker can trick a user into clicking an area of the vulnerable page that is different than what the user perceives the page to be. This can result in a user performing fraudulent or malicious transactions.
X-Frame-Options has been proposed by Microsoft as a way to mitigate clickjacking attacks and is currently supported by all major browser vendors.
Content-Security-Policy (CSP) has been proposed by the W3C Web Application Security Working Group, with increasing support among all major browser vendors, as a way to mitigate clickjacking and other attacks. The frame-ancestors' policy directive restricts which sources can embed the protected resource.
Note that while the X-Frame-Options and Content-Security-Policy response headers are not the only mitigations for clickjacking, they are currently the most reliable methods that can be detected through automation. Therefore, this plugin may produce false positives if other mitigation strategies (e.g., frame-busting JavaScript) are deployed or if the page does not perform any security-sensitive transactions.
Vulnerability Path: http://testfire.net/

Business Impact: An attacker could exploit this vulnerability to:

- Trick users into clicking on malicious buttons or links, such as fake login buttons or phishing links.
- Steal sensitive data, such as credit card information or login credentials.
- Make unauthorized transactions on the user's account.
- Launch denial-of-service attacks by flooding the victim's browser with requests.

Recommendation: Return the X-Frame-Options or Content-Security-Policy (with the 'frame-ancestors' directive) HTTP header with the page's response. This prevents the page's content from being rendered by another site when using the frame or iframe HTML tags.



6. Vulnerability Name: TLS Version 1.0 Protocol Detection

Risk Factor: Medium

CWE: 327

OSWAP CATEGORY: A02:2021-Cryptographic Failures

DESCRIPTION: The remote service accepts connections encrypted using TLS 1.0. TLS 1.0 has a number of cryptographic design flaws. Modern implementations of TLS 1.0 mitigate these problems, but newer versions of TLS like 1.2 and 1.3 are designed against these flaws and should be used whenever possible.

As of March 31, 2020, Endpoints that aren't enabled for TLS 1.2 and higher will no longer function properly with major web browsers and major vendors.

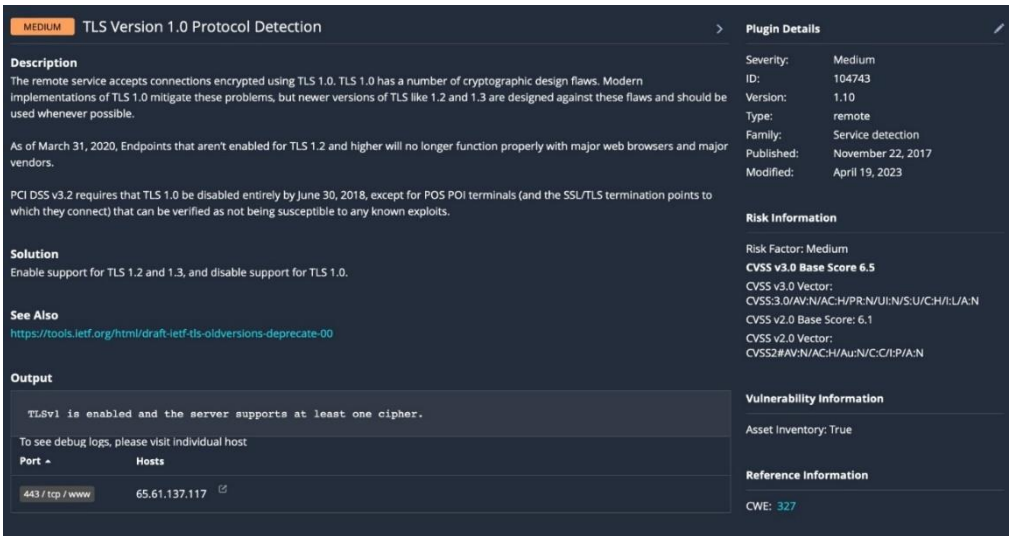
PCI DSS v3.2 requires that TLS 1.0 be disabled entirely by June 30, 2018, except for POS POI terminals (and the SSL/TLS termination points to which they connect) that can be verified as not being susceptible to any known exploits.

Vulnerability Path: http://testfire.net/

Business Impact: The business impact of TLS Version 1.0 Protocol Detection can be significant, depending on the nature of the business and the types of data that are being transmitted.
Potential business impacts:

- Data breaches: Attackers can exploit vulnerabilities in TLS 1.0 to intercept and decrypt traffic, including sensitive data such as passwords, credit card numbers, and personal information. This could lead to data breaches that could damage a company's reputation and result in financial losses.
- Compliance violations: Many industries, such as healthcare and finance, have regulations that require companies to use secure encryption protocols. Failure to comply with these regulations could result in fines, penalties, and other legal consequences.
- Loss of customer trust: Customers are increasingly aware of the importance of cybersecurity. If they learn that a company is using outdated and insecure encryption protocols, they may lose trust in that company and take their business elsewhere.

Recommendation: Enable support for TLS 1.2 and 1.3, and disable support for TLS 1.0.



7. Vulnerability Name: TLS Version 1.1 Protocol Deprecated

Risk Factor: Medium

CWE: 327

OSWAP CATEGORY: A02:2021-Cryptographic Failures

DESCRIPTION: The remote service accepts connections encrypted using TLS 1.1. TLS 1.1 lacks support for current and recommended cipher suites. Ciphers that support encryption before MAC computation, and authenticated encryption modes such as GCM cannot be used with TLS 1.1

As of March 31, 2020, Endpoints that are not enabled for TLS 1.2 and higher will no longer function properly with major web browsers and major vendors.

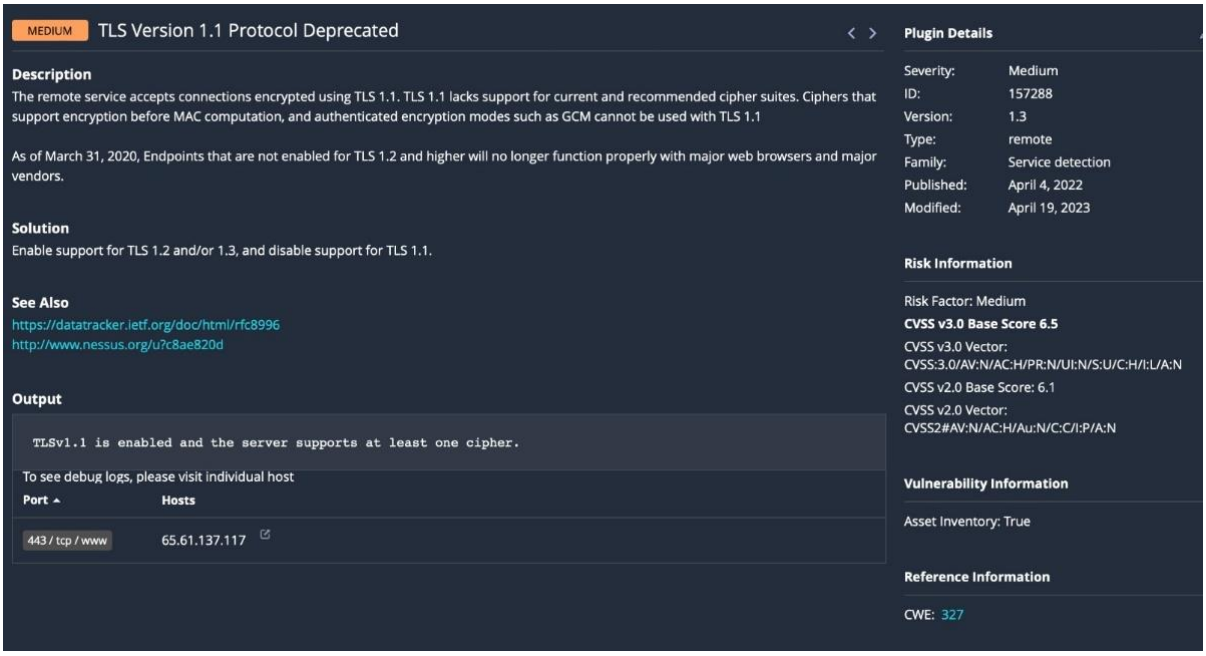
Vulnerability Path: http://testfire.net/

Business Impact: The business impact of TLS Version 1.1 Protocol Deprecated can be similar to the business impact of TLS Version 1.0 Protocol Detection, as described in my previous response.

Potential business impacts:

- Data breaches: Attackers can exploit vulnerabilities in TLS 1.0 to intercept and decrypt traffic, including sensitive data such as passwords, credit card numbers, and personal information. This could lead to data breaches that could damage a company's reputation and result in financial losses.
- Compliance violations: Many industries, such as healthcare and finance, have regulations that require companies to use secure encryption protocols. Failure to comply with these regulations could result in fines, penalties, and other legal consequences.
- Loss of customer trust: Customers are increasingly aware of the importance of cybersecurity. If they learn that a company is using outdated and insecure encryption protocols, they may lose trust in that company and take their business elsewhere.

Recommendation: Enable support for TLS 1.2 and 1.3, and disable support for TLS 1.1.



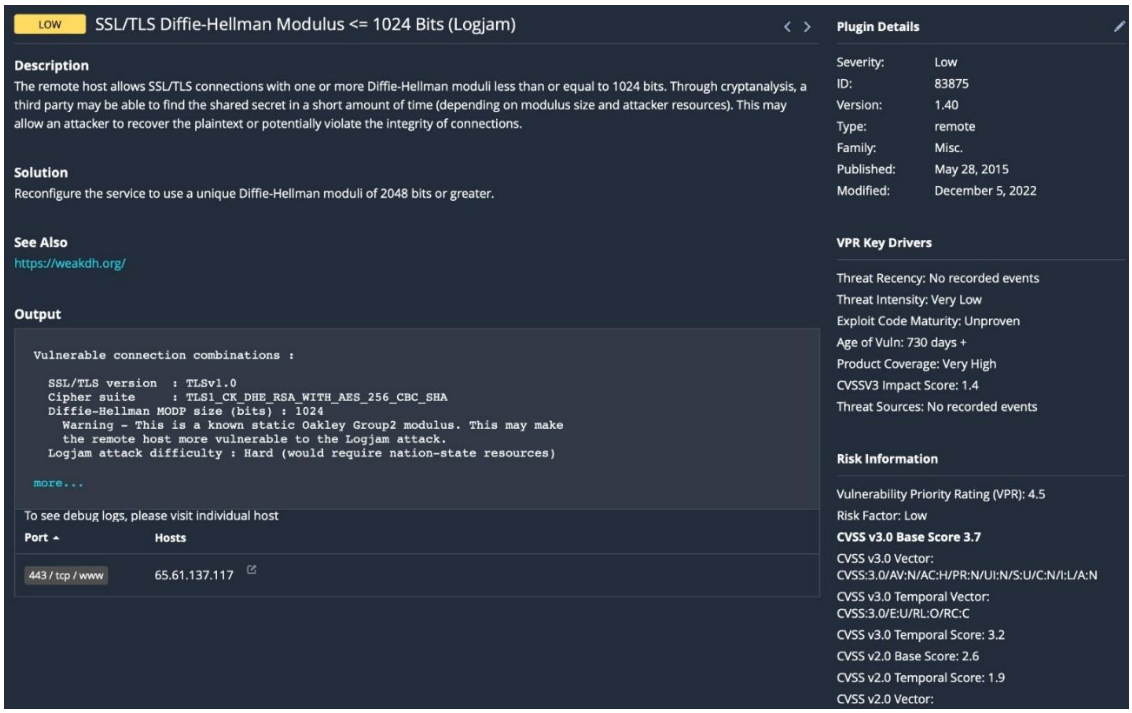
8. Vulnerability Name: SSL/TLS Diffie-Hellman Modulus<=1024 Bits(Logjam)

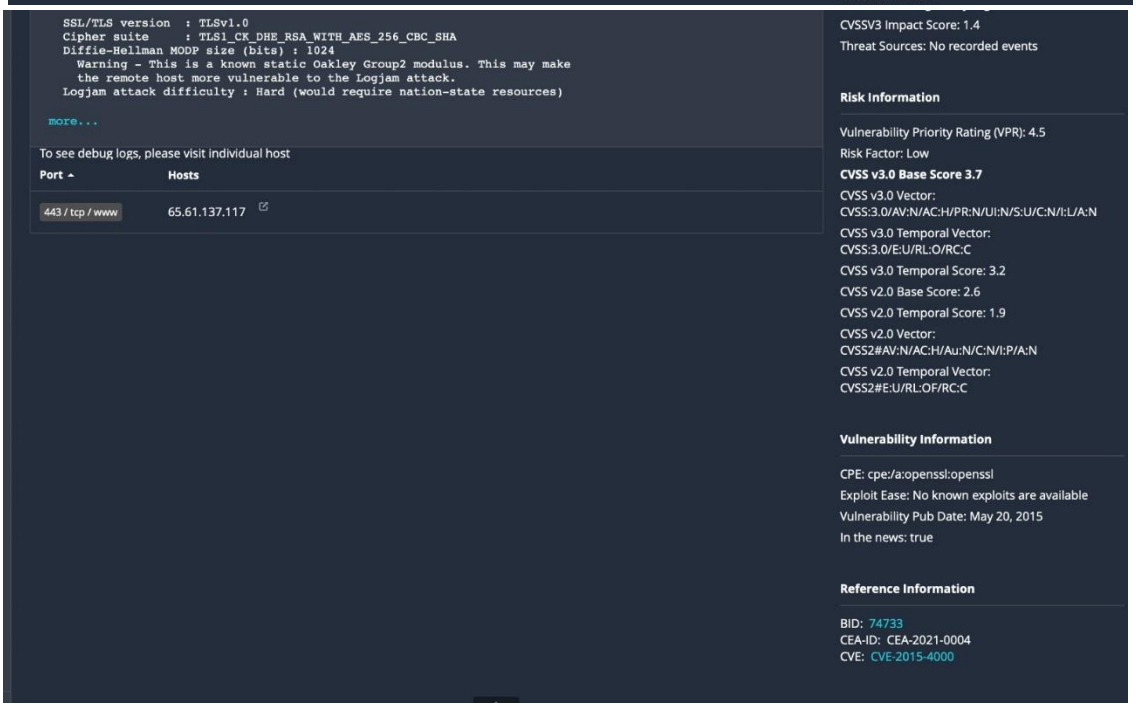
Risk Factor: Low

CVE: CVE-2015-4000

DESCRIPTION: The remote host allows SSL/TLS connections with one or more Diffie-Hellman moduli less than or equal to 1024 bits. Through cryptanalysis, a third party may be able to find the shared secret in a short amount of time (depending on modulus size and attacker resources). This may allow an attacker to recover the plaintext or potentially violate the integrity of connections.

Vulnerability Path: http://testfire.net/





9. Vulnerability Name: Web Server Transmits Cleartext Credentials

Risk Factor: Low

CWE: 522, 523, 718, 724, 928, 930

OSWAP CATEGORY: A02:2021- Cryptographic Failures

DESCRIPTION: The remote web server contains several HTML form fields containing an input of type 'password' which transmit their information to a remote web server in cleartext.

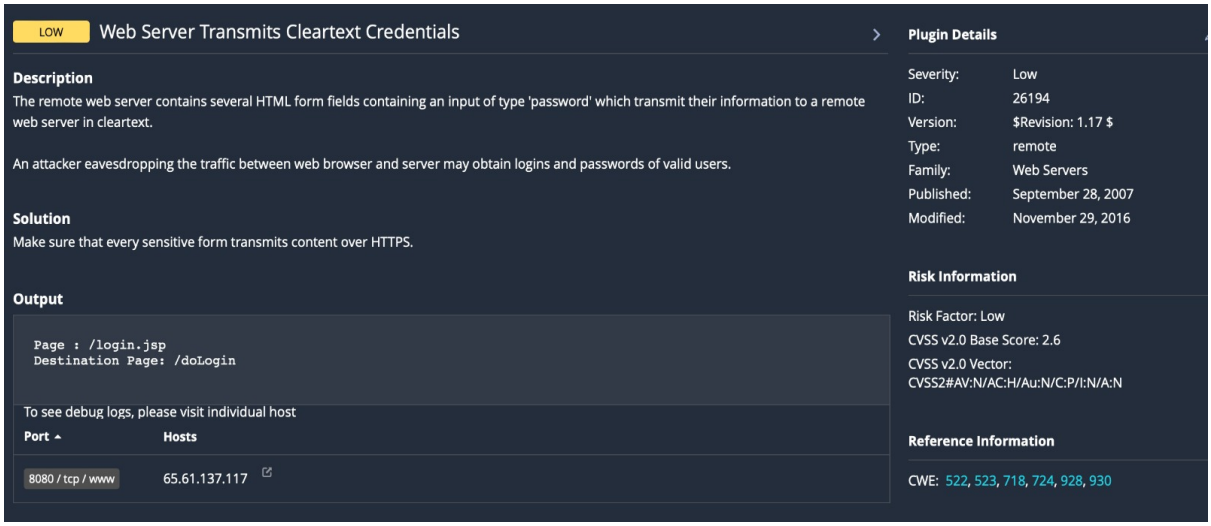
An attacker eavesdropping the traffic between web browser and server may obtain logins and passwords of valid users.

Vulnerability Path: http://testfire.net/

Business Impact: An attacker eavesdropping on the traffic between a web browser and a server could obtain the logins and passwords of valid users. This could allow the attacker to:

- Gain access to user accounts, including sensitive data such as credit card information, personal information, or financial data.
- Launch denial-of-service attacks by flooding the victim's account with requests.
- Impersonate users to commit fraud or other malicious activities.

Recommendation: Make sure that every sensitive form transmits content over HTTPS.



10. Vulnerability Name: Web Server Allows Password Auto-Completion

Risk Factor: Low

OSWAP CATEGORY: A07:2021- Identification and Authentication Failures

DESCRIPTION: The remote web server contains at least one HTML form field that has an input of type 'password' where 'autocomplete' is not set to 'off'.

While this does not represent a risk to this web server per se, it does mean that users who use the affected forms may have their credentials saved in their browsers, which could in turn lead to a loss of confidentiality if any of them use a shared host or if their machine is compromised at some point.

Vulnerability Path: http://testfire.net/

Business Impact: The business impact of a web server allowing password auto-completion is low, but it is still a good practice to mitigate this risk. If a user's browser is compromised, an attacker could gain access to their saved credentials and use them to access the web server.

Recommendation: Add the attribute 'autocomplete=off' to these fields to prevent browsers from catching credentials.