Scratchpad Template

Infrastructure Vulnerability Project

Designed to capture data and insights during the project work, which will not necessarily be included in the final client report.

Table of Contents

General Information

1. **Date: 9/5/2024**

2. **Cyber assessor (Student) Name: Christian Stevens**
3. **Client company name:** [redacted]
4. **Assessor's objectives: The objective of this project is to conduct an external, unauthenticated cybersecurity vulnerability assessment of My 1Health's Infrastructure.**
5. **Assessment Scope: Conduct a detailed vulnerability assessment of the** [redacted] **Infrastructure: 102.37.157.86; 41.30.245.67; 102.37.157.213**
6. **Tools used:** Nmap, OpenVAS, and Nessus

Client Interview - Summary Report Template

Designed to capture detailed information about an organization's network and systems infrastructure.

Organization Information

1. **Organization Name:** [redacted]
2. **Contact Person:** [redacted]

Scope and Drivers

1. **Target Systems:** 102.37.157.86; 41.60.245.67; 102.37.157.213
2. **Testing Drivers:** None
3. **Business Drivers:** Find vulnerabilities that could cause potential damage.

Scheduling and Logistics

1. **Start Date:** 8/21/2024
2. **End Date:** 9/7/2024
3. **Expected Duration:** Several hours
4. **Outside Working Hours:** None but preferred in the mornings

System and Infrastructure

1. **Critical Systems on Target IP Addresses:** (Yes/No, if yes, list systems)
2. **Public IP Addresses:** (List all public IP addresses or ranges)
3. **Cloud Environment:** (AWS, Microsoft Azure, IBM, GCP, Other Webhost, Private Cloud, On-Premise)
4. **Private IP Addresses:** (Optional, only if conducting an Internal Vulnerability Assessment)
5. **Company Domain Names:** (List all domain names associated with target systems)

Additional Information

1. **Known Vulnerabilities:** (List any known vulnerabilities in target systems)
2. **Security Documentation:** (Links to relevant security policies or architecture diagrams)

Testing Exclusions

1. **Specific systems or functionalities:** (Explain why these are excluded)
2. **Data or environments:** (Explain why these are excluded)

Nessus, NMAP and OpenVAS scan results

Nmap Findings

- Hostname: https://insurance[redacted]
- IP1: 41.60.245.67
- Ports Open: 80 SSH/53 DOMAIN/80 HTTP
    - TCP: 22/53/80
    - UDP:
- Nmap command used: nmap –scrpit vuln 41.60.245.67
- Observations: CVE-2011-3192/CVE-2007-6750
- Screenshots of scan results:

```
┌──(christian㉿kali)-[~]
└─$ nmap --script vuln 41.60.245.67
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-05 17:50 PDT
Nmap scan report for 41.60.245.67
Host is up (0.20s latency).
Not shown: 993 filtered tcp ports (no-response)
PORT     STATE SERVICE
22/tcp   open  ssh
53/tcp   open  domain
80/tcp   open  http
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
| http-vuln-cve2011-3192:
|   VULNERABLE:
|   Apache byterange filter DoS
|     State: VULNERABLE
|     IDs:  BID:49303  CVE:CVE-2011-3192
|       The Apache web server is vulnerable to a denial of service attack when numerous
|       overlapping byte ranges are requested.
|     Disclosure date: 2011-08-19
|     References:
|       https://www.tenable.com/plugins/nessus/55976
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-3192
|       https://www.securityfocus.com/bid/49303
|_      https://seclists.org/fulldisclosure/2011/Aug/175
|_http-csrf: Couldn't find any CSRF vulnerabilities.
443/tcp  open  https
|_http-vuln-cve2017-1001000: ERROR: Script execution failed (use -d to debug)
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
| http-vuln-cve2010-0738:
|_  /jmx-console/: Authentication was not required
| http-slowloris-check:
|   VULNERABLE:
|   Slowloris DOS attack
|     State: LIKELY VULNERABLE
|     IDs:  CVE:CVE-2007-6750
|       Slowloris tries to keep many connections to the target web server open and hold
|       them open as long as possible.  It accomplishes this by opening connections to
|       the target web server and sending a partial request. By doing so, it starves
|       the http server's resources causing Denial Of Service.
|
|     Disclosure date: 2009-09-17
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-6750
|_      http://ha.ckers.org/slowloris/
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-majordomo2-dir-traversal: ERROR: Script execution failed (use -d to debug)
| http-enum:
|   /blog/: Blog
|   /weblog/: Blog
|   /weblogs/: Blog
|   /wordpress/: Blog
|   /wiki/: Wiki
|   /mediawiki/: Wiki
|   /wiki/Main_Page: Wiki
|   /tikiwiki/: Tikiwiki
|   /cgi-bin/mj_wwwusr: Majordomo2 Mailing List
|   /majordomo/mj_wwwusr: Majordomo2 Mailing List
|   /j2ee/examples/servlets/: Oracle j2ee examples
|   /j2ee/examples/jsp/: Oracle j2ee examples
|   /dsc/: Trend Micro Data Loss Prevention Virtual Appliance
|   /reg_1.htm: Polycom IP phone
|   /adr.htm: Snom IP Phone
```

- Hostname:https:/[redacted] https://ims.[redacted] https://referral[redacted] https://patient-account.[redacted]
- IP2…: 102.37.157.86
- Ports Open: 53 DOMAIN/80 HTTP
    - TCP:53/80
    - UDP:
- Nmap command used: nmap –script vuln 102.37.157.86
- Observations: None
- Screenshots of scan results:

```
┌──(christian㉿kali)-[~]
└─$ nmap --script vuln 102.37.157.86
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-05 17:35 PDT
Nmap scan report for 102.37.157.86
Host is up (0.19s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT     STATE SERVICE
53/tcp   open  domain
80/tcp   open  http
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
443/tcp  open  https
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
5060/tcp open  sip
8080/tcp open  http-proxy
|_http-vuln-cve2014-3704: ERROR: Script execution failed (use -d to debug)
|_http-aspnet-debug: ERROR: Script execution failed (use -d to debug)

Nmap done: 1 IP address (1 host up) scanned in 260.22 seconds
```

- Hostname:
- IP1: 102.37.157.213
- Ports Open: 80 SSH/53 DOMAIN/80 HTTP
    - TCP: 22/53/80
    - UDP:
- Nmap command used: nmap –script vuln 102.37.157.213
- Observations: None
- Screenshots of scan results:

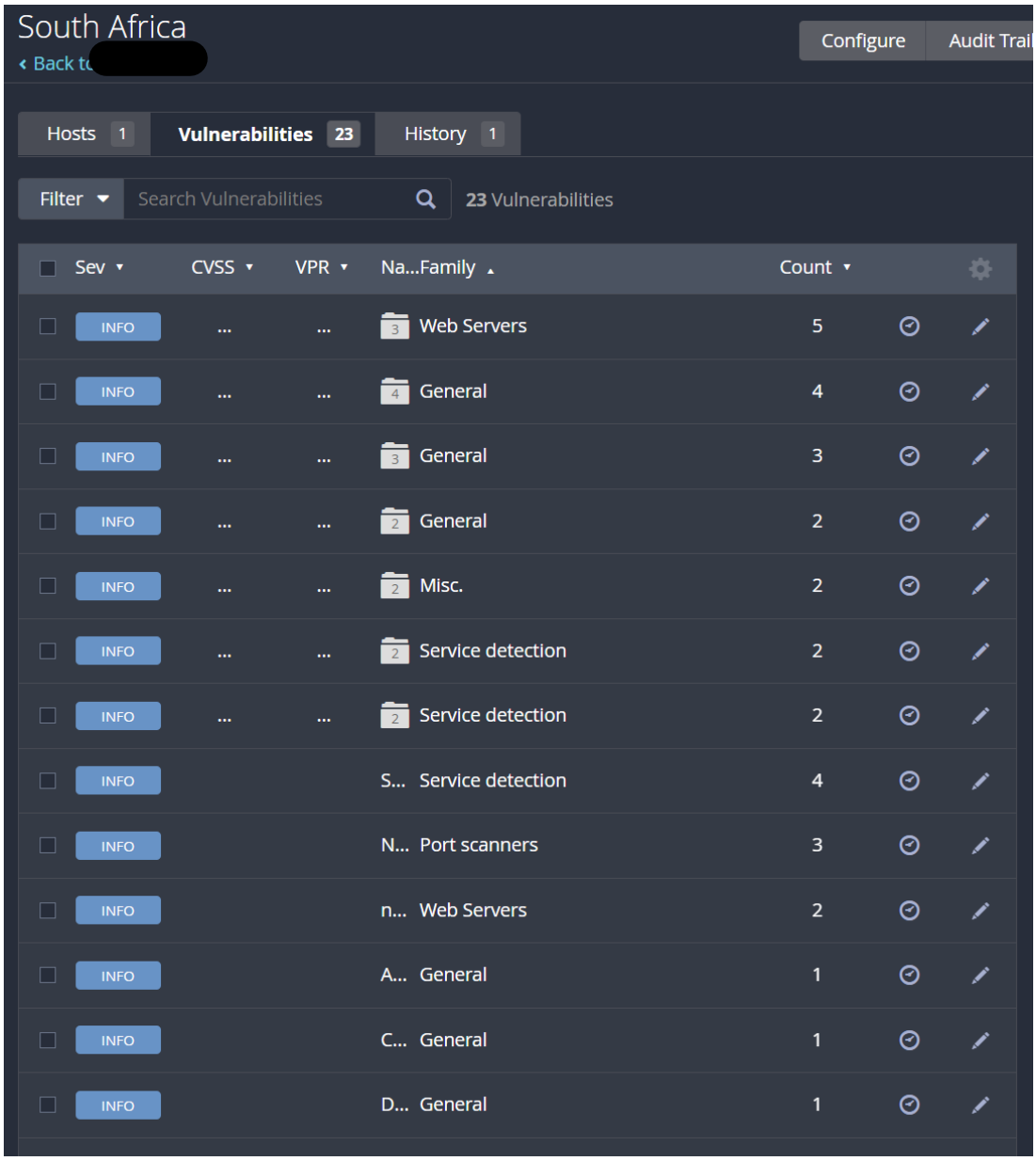
```
┌──(christian㉿kali)-[~]
└─$ nmap --script vuln 102.37.157.213
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-06 21:09 PDT
Nmap scan report for 102.37.157.213
Host is up (0.21s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT    STATE SERVICE
22/tcp  open  ssh
80/tcp  open  http
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
443/tcp open  https
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.

Nmap done: 1 IP address (1 host up) scanned in 1010.52 seconds
```

Nessus Findings

All vulnerabilities (including those classified as Low):

- IP1: 102.37.157.86
- Nessus Plugin ID:

- Vulnerability name: N/A
- Description: N/A
- CVE: N/A
- CVSS Score: N/A
- Likelihood: None
- Consequence: N/A
- Impact: N/A
- Recommendation: N/A
- Screenshots of scan results:



INFO	N...	Settings	1	
INFO	O...	Misc.	1	
INFO	O...	General	1	
INFO	O...	Settings	1	
INFO	S...	Misc.	1	
INFO	S...	General	1	
INFO	T...	Settings	1	
INFO	T...	General	1	
INFO	T...	General	1	
INFO	W...	Web Servers	1	

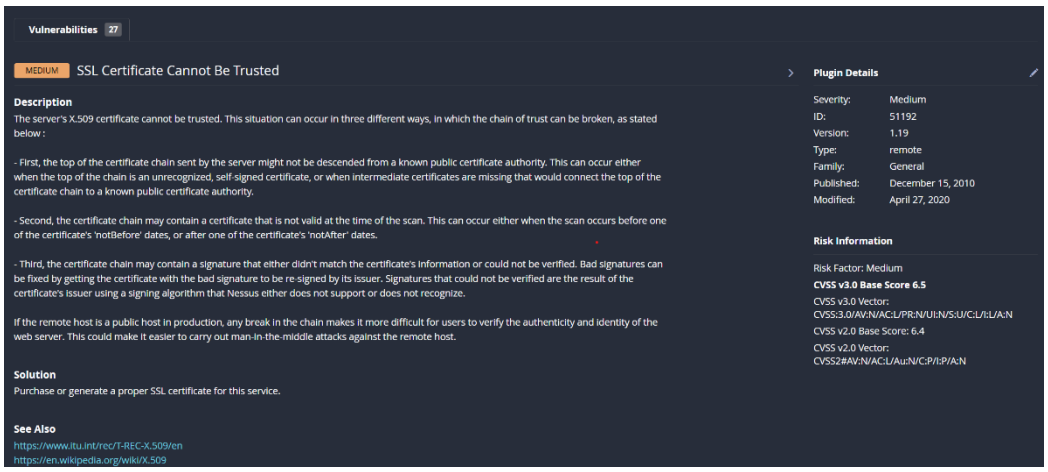
- IP2 41.60.245.67
- Nessus Plugin ID:
- Vulnerability name: SSL Certificate Cannot Be Trusted
- Description: The server's X.509 certificate cannot be trusted. This situation can occur in three different ways, in which the chain of trust can be broken, as stated below:

  - First, the top of the certificate chain sent by the server might not be descended from a known public certificate authority. This can occur either when the top of the chain is an unrecognized, self-signed certificate, or when intermediate certificates are missing that would connect the top of the certificate chain to a known public certificate authority.

  - Second, the certificate chain may contain a certificate that is not valid at the time of the scan. This can occur either when the scan occurs before one of the certificate's 'notBefore' dates, or after one of the certificate's 'notAfter' dates.

  - Third, the certificate chain may contain a signature that either didn't match the certificate's information or could not be verified. Bad signatures can be fixed by getting the certificate with the bad signature to be re-signed by its issuer. Signatures that could not be verified are the result of the certificate's issuer using a signing algorithm that Nessus either does not support or does not recognize.
- CVE: N/A
- CVSS Score:6.5
- Likelihood: Medium

- Consequence: If the remote host is a public host in production, any break in the chain makes it more difficult for users to verify the authenticity and identity of the web server. This could make it easier to carry out man-in-the-middle attacks against the remote host.
- Impact: Users are vulnerable to data interception, man-in-the-middle attacks, and malicious activities.
- Recommendation: Purchase or generate a proper SSL certificate for this service
- Screenshots of scan results:



Vulnerabilities 27

MEDIUM SSL Certificate Cannot Be Trusted

Description

The server's X.509 certificate cannot be trusted. This situation can occur in three different ways, in which the chain of trust can be broken, as stated below :

- First, the top of the certificate chain sent by the server might not be descended from a known public certificate authority. This can occur either when the top of the chain is an unrecognized, self-signed certificate, or when intermediate certificates are missing that would connect the top of the certificate chain to a known public certificate authority.

- Second, the certificate chain may contain a certificate that is not valid at the time of the scan. This can occur either when the scan occurs before one of the certificate's 'notBefore' dates, or after one of the certificate's 'notAfter' dates.

- Third, the certificate chain may contain a signature that either didn't match the certificate's information or could not be verified. Bad signatures can be fixed by getting the certificate with the bad signature to be re-signed by its issuer. Signatures that could not be verified are the result of the certificate's issuer using a signing algorithm that Nessus either does not support or does not recognize.

If the remote host is a public host in production, any break in the chain makes it more difficult for users to verify the authenticity and identity of the web server. This could make it easier to carry out man-in-the-middle attacks against the remote host.

Solution

Purchase or generate a proper SSL certificate for this service.

See Also

https://www.itu.int/rec/T-REC-X.509/en
https://en.wikipedia.org/wiki/X.509

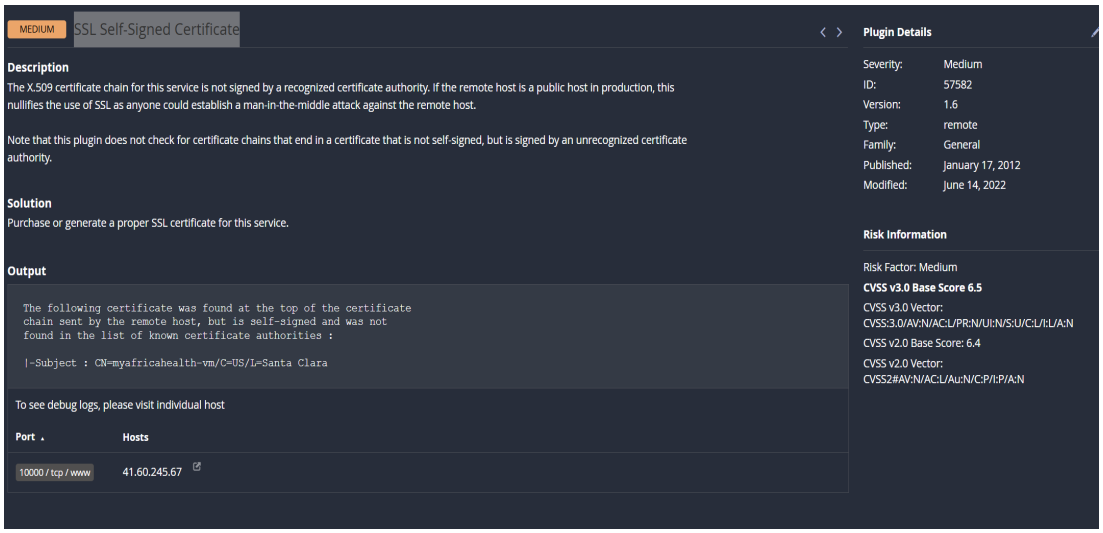
Plugin Details

Severity:	Medium
ID:	51192
Version:	1.19
Type:	remote
Family:	General
Published:	December 15, 2010
Modified:	April 27, 2020

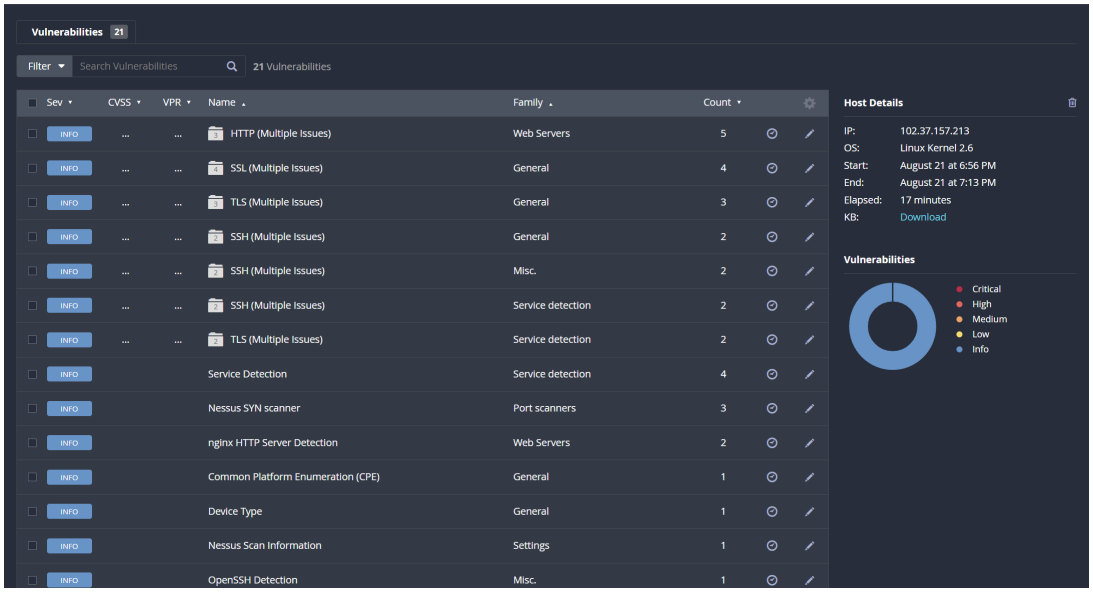
Risk Information

Risk Factor: Medium
CVSS v3.0 Base Score 6.5
CVSS v3.0 Vector:
CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N
CVSS v2.0 Base Score: 6.4
CVSS v2.0 Vector:
CVSS2#AV:N/AC:L/Au:N/C:P/I:P/A:N

- IP2: 41.60.245.67
- Nessus Plugin ID:
- Vulnerability Name: SSL Self-Guided Certificate
- Description: The X.509 certificate chain for this service is not signed by a recognized certificate authority. If the remote host is a public host in production, this nullifies the use of SSL as anyone could establish a man-in-the-middle attack against the remote host.
- CVE: None
- CVSS Score: 6.5
- Likelihood: Medium
- Consequence: Note that this plugin does not check for certificate chains that end in a certificate that is not self-signed, but is signed by an unrecognized certificate authority.
- Impact: lack of trust and authentication
- Recommendation: Purchase or generate a proper SSL certificate for this service.
- Screenshots of scan results:



- IP3: 102.37.157.213
- Nessus Plugin ID:
- Vulnerability name: N/A
- Description: N/A
- CVE: N/A
- CVSS Score: N/A
- Likelihood: N/A
- Consequence: None
- Impact: None
- Recommendation: None
- Screenshots of scan results:



INFO	OS Identification	General	1	
INFO	OS Security Patch Assessment Not Available	Settings	1	
INFO	SSL Root Certification Authority Certificate Information	General	1	
INFO	Target Credential Status by Authentication Protocol - No Crede...	Settings	1	
INFO	TCP/IP Timestamps Supported	General	1	
INFO	Traceroute Information	General	1	
INFO	Web Server No 404 Error Code Check	Web Servers	1	

"Informational" findings (list)

- HTTP ServerType and Version
- HyperText Transfer Protocol (HTTP) Information
- HSTS Missing from HTTPS Server
- SSL Certificatre Expiry – Future Expiry (10/19/24)
- SSL Certificate Information
- SSL Cipher Suites Supported
- SSL Perfect Forward Secrecy Cipher Suites Supported
- SSL/TLS Versions Supported
- SSL/TLS Recommended Cipher Suites
- TLS Next Protocol Supported
- Backported Security Patch Detection (SSH)
- SSH Protocol Versions Supported
- SSH Algorithms and Language Supported
- SSH SHA-1 HMAC Algorithms Enabled
- SSH Password Authentication Accepted
- SSH Server Type and Version Information
- TLS Version 1.2 Protocol Detection
- TLS Version 1.3 Protocol Detection
- Common Platform Enumeration (CPE)

OpenVAS findings

All vulnerabilities (including those classified as Low):

- IP1: 102.37.157.86
- OpenVAS Plugin ID:
- Vulnerability name: Missing Secure Cookie Attribute (HTTP)
- Description: The remote HTTP web server / application is missing to set 'Secure' cookie attribute for one or more sent HTTP cookie

- CVE: N/A
- OpenVAS Risk Rating: 5.0
- Likelihood: Medium
- Consequence: Man-in-the-middle attacks, session hijacking, compromised confidentiality, increased risk in shared or public networks, cross-site scripting (XSS) Exploitation, and failing security best practices.
- Impact: Loss of confidentiality, Phishing and Social Engineering, and Downgrade Attacks.
- Recommendation: Set the 'Secure' cookie attribute for any cookies that are sent over a SSL/TLS connection
- Screenshots of scan results:

NVT: Missing 'Secure' Cookie Attribute (HTTP)

ID: 1.3.6.1.4.1.25623.1.0.902661 Created: Thu, Mar 1, 2012 11:40 AM UTC Modified: Fri, Jan 12, 2024 4:12 PM UTC Owner: *(Global Object)*

Information | Preferences (0) | User Tags (0)

Summary

The remote HTTP web server / application is missing to set the 'Secure' cookie attribute for one or more sent HTTP cookie.

Scoring

CVSS Base	5.0 (Medium)
CVSS Base Vector	AV:N/AC:L/Au:N/C:P/I:N/A:N
CVSS Origin	N/A
CVSS Date	Thu, Mar 1, 2012 11:40 AM UTC

Insight

The flaw exists if a cookie is not using the 'Secure' cookie attribute and is sent over a SSL/TLS connection.

This allows a cookie to be passed to the server by the client over non-secure channels (HTTP) and subsequently allows an attacker to e.g. conduct session hijacking attacks.

Detection Method

Checks all cookies sent by the remote HTTP web server / application over a SSL/TLS connection for a missing 'Secure' cookie attribute.
Quality of Detection: remote_analysis (70%)

Affected Software/OS

Any web application accessible via a SSL/TLS connection (HTTPS) and at the same time also accessible over a cleartext connection (HTTP).

Solution

Solution Type: Mitigation
- Set the 'Secure' cookie attribute for any cookies that are sent over a SSL/TLS connection

- Evaluate / do an own assessment of the security impact on the web server / application and create an override for this result if there is none (this can't be checked automatically by this VT)

Family

Web application abuses

References

Other https://www.rfc-editor.org/rfc/rfc6265#section-5.2.5
https://owasp.org/www-community/controls/SecureCookieAttribute
https://wiki.owasp.org/index.php/Testing_for_cookies_attributes_(OTG-SESS-002)

- IP1: 102.37.157.86
- OpenVAS Plugin ID:
- Vulnerability name: Missing 'HTTPOnly' Cookie Attribute
- Description: The remote HTTP web server / application is missing to set 'HTTPOnly' cookie attribute for one or more sent HTTP cookie
- CVE: N/A
- OpenVAS Risk Rating: 5.0
- Likelihood: Medium

- Consequence: Man-in-the-middle attacks, session hijacking, compromised confidentiality, increased risk in shared or public networks, cross-site scripting (XSS) Exploitation, and failing security best practices.
- Impact: Loss of confidentiality, Phishing and Social Engineering, and Downgrade Attacks.
- Recommendation: Set the 'Secure' cookie attribute for any cookies that are sent over a SSL/TLD connection
- Screenshots of scan results:

NVT: Missing 'HttpOnly' Cookie Attribute (HTTP)

ID: 1.3.6.1.4.1.25623.1.0.105925 Created: Mon, Sep 1, 2014 3:00 PM UTC Modified: Fri, Jan 12, 2024 4:12 PM UTC Owner: *(Global Object)*

Information | Preferences *(0)* | User Tags *(0)*

Summary

The remote HTTP web server / application is missing to set the 'HttpOnly' cookie attribute for one or more sent HTTP cookie.

Scoring

CVSS Base	5.0 (Medium)
CVSS Base Vector	AV:N/AC:L/Au:N/C:P/I:N/A:N
CVSS Origin	N/A
CVSS Date	Mon, Sep 1, 2014 3:00 PM UTC

Insight

The flaw exists if a session cookie is not using the 'HttpOnly' cookie attribute.

This allows a cookie to be accessed by JavaScript which could lead to session hijacking attacks.

Detection Method

Checks all cookies sent by the remote HTTP web server / application for a missing 'HttpOnly' cookie attribute.
Quality of Detection: remote_analysis (70%)

Affected Software/OS

Any web application with session handling in cookies.

Solution

Solution Type: Mitigation
- Set the 'HttpOnly' cookie attribute for any session cookie

- Evaluate / do an own assessment of the security impact on the web server / application and create an override for this result if there is none (this can't be checked automatically by this VT)

Family

Web application abuses

References

Other https://www.rfc-editor.org/rfc/rfc6265#section-5.2.6
https://owasp.org/www-community/HttpOnly
https://wiki.owasp.org/index.php/Testing_for_cookies_attributes_(OTG-SESS-002)

- IP1: 102.37.157.86
- OpenVAS Plugin ID:
- Vulnerability Name: SSL/TLS: Renegotiation DOS Vulnerability
- Description: The remote SSL/TLS service is prone to denial of service (DOS) vulnerability
- CVE: 2011-1473; 2011-5094
- OpenVAS Risk Rating: 5.0
- Likelihood: Medium
- Consequence: Denial of Service (DoS) Attacks, Resource Exhaustion, and Service Disruption

- Impact: Service unavailability, financial loss, reputation damage, operational lost, and compliance and legal risks.
- Recommendation: Remove/disable renegotiations capabilities altogether from/in the affected SSL/TLS service.
- Screenshots of scan results:

NVT: SSL/TLS: Renegotiation DoS Vulnerability (CVE-2011-1473, CVE-2011-5094)

ID: 1.3.6.1.4.1.25623.1.0.117761 | Created: Fri, Oct 29, 2021 8:24 AM UTC | Modified: Wed, Jul 24, 2024 5:06 AM UTC | Owner: (Global Object)

Information | Preferences (0) | User Tags (0)

Summary

The remote SSL/TLS service is prone to a denial of service (DoS) vulnerability.

Scoring

CVSS Base	5.0 (Medium)
CVSS Base Vector	AV:N/AC:L/Au:N/C:N/I:N/A:P
CVSS Origin	N/A
CVSS Date	Fri, Oct 29, 2021 8:24 AM UTC

Insight

The flaw exists because the remote SSL/TLS service does not properly restrict client-initiated renegotiation within the SSL and TLS protocols.

Note: The referenced CVEs are affecting OpenSSL and Mozilla Network Security Services (NSS) but both are in a DISPUTED state with the following rationale:

> It can also be argued that it is the responsibility of server deployments, not a security library, to prevent or limit renegotiation when it is inappropriate within a specific environment.

Both CVEs are still kept in this VT as a reference to the origin of this flaw.

Detection Method

Checks if the remote service allows to re-do the same SSL/TLS handshake (Renegotiation) over an existing / already established SSL/TLS connection.
Quality of Detection: remote_analysis (70%)

Affected Software/OS

Every SSL/TLS service which does not properly restrict client-initiated renegotiation.

Impact

The flaw might make it easier for remote attackers to cause a DoS (CPU consumption) by performing many renegotiations within a single connection.

Solution

Solution Type: Vendorfix
Users should contact their vendors for specific patch information.

A general solution is to remove/disable renegotiation capabilities altogether from/in the affected SSL/TLS service.

Family

SSL and TLS

References

- IP1: 102.37.157.86
- OpenVAS Plugin ID:
- Vulnerability name: Weak MAC Algorithm (s) Supported (SSH)
- Description: The remote server is configured to allow / support weak MAC algorithms
- CVE: N/A
- OpenVAS Risk Rating: 2.6
- Likelihood: Low
- Consequence: Data Integrity Compromise, Authentication Bypass, Replay Attacks, Man-In-The-Middle Attacks, Cryptographic Attacks
- Impact: Financial Loss, Reputation Damage, Impact on Encrypted Communications, and Non-Compliance with Security Standards
- Recommendation: Disable the reported weak MAC algorithm
- Screenshots of scan results:

Summary

The remote SSH server is configured to allow / support weak MAC algorithm(s).

Detection Result

```
The remote SSH server supports the following weak client-to-server MAC algorithm(s):

umac-64-etm@openssh.com
umac-64@openssh.com


The remote SSH server supports the following weak server-to-client MAC algorithm(s):

umac-64-etm@openssh.com
umac-64@openssh.com
```

Product Detection Result

Product cpe:/a:ietf:secure_shell_protocol

Method SSH Protocol Algorithms Supported (OID: 1.3.6.1.4.1.25623.1.0.105565)

Log View details of product detection

Detection Method

Checks the supported MAC algorithms (client-to-server and server-to-client) of the remote SSH server.

Currently weak MAC algorithms are defined as the following:

- MD5 based algorithms

- 96-bit based algorithms

- 64-bit based algorithms

- 'none' algorithm

Details: Weak MAC Algorithm(s) Supported (SSH) OID: 1.3.6.1.4.1.25623.1.0.105610

Version used: 2024-06-14T05:05:48Z

Solution

Solution Type: Mitigation
Disable the reported weak MAC algorithm(s).

References

Other https://www.rfc-editor.org/rfc/rfc6668
https://www.rfc-editor.org/rfc/rfc4253#section-6.4

(Applied filter: apply_overrides=0 levels=hml rows=100 min_qod=70 first=1 sort-reverse=severity) 1 - 3 of 3

- IP1: 41.60.245.67
- OpenVAS Plugin ID:
- Vulnerability name: SSL/TLS: Renegotiation DOS Vulnerability
- Description: The remote SSL/TLS service is prone to denial of service (DOS) vulnerability
- CVE: 2011-1473; 2011-5094
- OpenVAS Risk Rating: 5.0
- Likelihood: Medium
- Consequence: Denial of Service (DoS) Attacks, Resource Exhaustion, and Service Disruption
- Impact: Service unavailability, financial loss, reputation damage, operational lost, and compliance and legal risks.
- Recommendation: Remove/disable renegotiations capabilities altogether from/in the affected SSL/TLS service.
- Screenshots of scan results:

Report: Sat, Sep 7, 2024 11:09 AM UTC Done ID: e349582a-bbe7-43a0-8731-74da9b97f088 Created: Sat, Sep 7, 2024 11:09 AM UTC Modified: Sat, Sep 7, 2024 12:24 PM UTC Owner: adm

Information | Results (4 of 86) | Hosts (1 of 1) | Ports (3 of 5) | Applications (9 of 9) | Operating Systems (1 of 1) | CVEs (1 of 1) | Closed CVEs (0 of 0) | TLS Certificates (2 of 2) | Error Messages (3 of 3) | User Tags (0)

1 - 4 of 4

Vulnerability	Severity ▼	QoD	Host IP	Host Name	Location	Created
SSL/TLS: Renegotiation DoS Vulnerability (CVE-2011-1473, CVE-2011-5094)	5.0 (Medium)	70 %	41.60.245.67	insurance-old.my1health.com	10000/tcp	Sat, Sep 7, 2024 11:51 AM UTC

Summary

The remote SSL/TLS service is prone to a denial of service (DoS) vulnerability.

Detection Result

```
The following indicates that the remote SSL/TLS service is affected:

Protocol Version | Successful re-done SSL/TLS handshakes (Renegotiation) over an existing / already established SSL/TLS connection
----------------------------------------------------------------------------------------------------------------------------
TLSv1.2          | 10
```

Insight

The flaw exists because the remote SSL/TLS service does not properly restrict client-initiated renegotiation within the SSL and TLS protocols.

Note: The referenced CVEs are affecting OpenSSL and Mozilla Network Security Services (NSS) but both are in a DISPUTED state with the following rationale:

> It can also be argued that it is the responsibility of server deployments, not a security library, to prevent or limit renegotiation when it is inappropriate within a specific environment.

Both CVEs are still kept in this VT as a reference to the origin of this flaw.

Detection Method

Checks if the remote service allows to re-do the same SSL/TLS handshake (Renegotiation) over an existing / already established SSL/TLS connection.

Details: SSL/TLS: Renegotiation DoS Vulnerability (CVE-2011-1473, CVE-2011-5094)..OID: 1.3.6.1.4.1.25623.1.0.117761

Version used: 2024-07-24T05:06:37Z

Affected Software/OS

Every SSL/TLS service which does not properly restrict client-initiated renegotiation.

Impact

The flaw might make it easier for remote attackers to cause a DoS (CPU consumption) by performing many renegotiations within a single connection.

Solution

Solution Type: Vendorfix
Users should contact their vendors for specific patch information.

- IP2: 41.60.245.67
- OpenVAS Plugin ID:
- Vulnerability name: Missing ‘HTTPOnly’ Cookie Attribute
- Description: The remote HTTP web server / application is missing to set ‘HTTPOnly’ cookie attribute for one or more sent HTTP cookie
- CVE: N/A
- OpenVAS Risk Rating: 5.0
- Likelihood: Medium
- Consequence: Man-in-the-middle attacks, session hijacking, compromised confidentiality, increased risk in shared or public networks, cross-site scripting (XSS) Exploitation, and failing security best practices.
- Impact: Loss of confidentiality, Phishing and Social Engineering, and Downgrade Attacks.
- Recommendation: Set the ‘Secure’ cookie attribute for any cookies that are sent over a SSL/TLD connection
- Screenshots of scan results:

Summary

The remote HTTP web server / application is missing to set the 'HttpOnly' cookie attribute for one or more sent HTTP cookie.

Detection Result

```
The cookie(s):

Set-Cookie: XSRF-
TOKEN=eyJpdiI6IlpKRmRtbDlQYWswN01naHEyaDRYSXc9PSIsInZhbHVlIjoiWERlZ1Bhc2hwek1VKzk3SjlVVXlSUUxmV2NpTEpnNEtKTnNhTnZQTER5Z1QrejI0Y0s4c29aUkx6NGRzN0c2TUF
VUXd2M2FUbFR2SkFacXZiMng2RGFkZGhyOXU0anJyWksvZUVuY25ya0pxZTFJNHFad2NrZG00dnVmTStEZjgiLCJtYWMiOiJhMDNiODFmZTUyNDdkYjYzMDdhZDEzNjAyYzAyYzE2MGM0OTA3YTIz
M2VkODJmZmE1OTJkYmUwY2EyNTk2N2U0IiwidGFnIjoiIn0%3D; expires=Sat, 07 Sep 2024 13:15:08 GMT; Max-Age=***replaced***; path=/; secure; samesite=lax

is/are missing the "HttpOnly" cookie attribute.
```

Insight

The flaw exists if a session cookie is not using the 'HttpOnly' cookie attribute.

This allows a cookie to be accessed by JavaScript which could lead to session hijacking attacks.

Detection Method

Checks all cookies sent by the remote HTTP web server / application for a missing 'HttpOnly' cookie attribute.

Details: Missing 'HttpOnly' Cookie Attribute (HTTP) OID: 1.3.6.1.4.1.25623.1.0.105925

Version used: 2024-01-12T16:12:12Z

Affected Software/OS

Any web application with session handling in cookies.

Solution

Solution Type: Mitigation
- Set the 'HttpOnly' cookie attribute for any session cookie

- Evaluate / do an own assessment of the security impact on the web server / application and create an override for this result if there is none (this can't be checked automatically by this VT)

References

Other https://www.rfc-editor.org/rfc/rfc6265#section-5.2.6
https://owasp.org/www-community/HttpOnly
https://wiki.owasp.org/index.php/Testing_for_cookies_attributes_(OTG-SESS-002)

- IP2: 41.60.245.67
- OpenVAS Plugin ID:
- Vulnerability name: Missing Secure Cookie Attribute (HTTP)
- Description: The remote HTTP web server / application is missing to set ‘Secure’ cookie attribute for one or more sent HTTP cookie
- CVE: N/A
- OpenVAS Risk Rating: 5.0
- Likelihood: Medium
- Consequence: Man-in-the-middle attacks, session hijacking, compromised confidentiality, increased risk in shared or public networks, cross-site scripting (XSS) Exploitation, and failing security best practices.
- Impact: Loss of confidentiality, Phishing and Social Engineering, and Downgrade Attacks.
- Recommendation: Set the ‘Secure’ cookie attribute for any cookies that are sent over a SSL/TLD connection
- Screenshots of scan results:

Summary

The remote HTTP web server / application is missing to set the
'Secure' cookie attribute for one or more sent HTTP cookie.

Detection Result

```
The cookie(s):

Set-Cookie:
insurance_session=eyJpdiI6IlBFS2ZOaEQ0SFc2d2IyRmJ0RWdERFE9PSIsInZhbHVlIjoiR1RBZjNqZ1d5YUtCQm5jWjlrWWYyMStRSDVCYU9CV1hlOFZuNTZEc0VSR1pIUTkzUTVFTEtxTzl
3OTdYU202cVFtZ0pLQTVNT05sYnhEakZYZ0lUWnBLc0cySjd0bjZYLzJNR3VpSnNtc2xTclJqcmlESWhWcHFoWjdQYUNuVjQiLCJtYWMiOiI5NWMwYTQ3ZmViYTIxOTNiMWViMTI3M2I1N2Q5YzM2
NTNmNGI0NDFjMzNkOGQ0YWNjZjRhZjBmMjUyYTQ2MWNhIiwidGFnIjoiIn0%3D; expires=Sat, 07 Sep 2024 13:15:08 GMT; Max-Age=***replaced***; path=/; httponly;
samesite=lax

is/are missing the "Secure" cookie attribute.
```

Insight

The flaw exists if a cookie is not using the 'Secure' cookie
attribute and is sent over a SSL/TLS connection.

This allows a cookie to be passed to the server by the client over non-secure channels (HTTP) and
subsequently allows an attacker to e.g. conduct session hijacking attacks.

Detection Method

Checks all cookies sent by the remote HTTP web server /
application over a SSL/TLS connection for a missing 'Secure' cookie attribute.

Details: Missing 'Secure' Cookie Attribute (HTTP) OID: 1.3.6.1.4.1.25623.1.0.902661

Version used: 2024-01-12T16:12:12Z

Affected Software/OS

Any web application accessible via a SSL/TLS connection (HTTPS)
and at the same time also accessible over a cleartext connection (HTTP).

Solution

Solution Type: Mitigation
- Set the 'Secure' cookie attribute for any cookies that are
sent over a SSL/TLS connection

- Evaluate / do an own assessment of the security impact on the web server / application and create
an override for this result if there is none (this can't be checked automatically by this VT)

References

Other https://www.rfc-editor.org/rfc/rfc6265#section-5.2.5
https://owasp.org/www-community/controls/SecureCookieAttribute
https://wiki.owasp.org/index.php/Testing_for_cookies_attributes_(OTG-SESS-002)

- IP2: 41.60.245.67
- OpenVAS Plugin ID:
- Vulnerability name: Weak MAC Algorithm (s) Supported (SSH)
- Description: The remote server is configured to allow / support weak MAC algorithms
- CVE: N/A
- OpenVAS Risk Rating: 2.6
- Likelihood: Low
- Consequence: Data Integrity Compromise, Authentication Bypass, Replay Attacks, Man-In-The-Middle Attacks, Cryptographic Attacks
- Impact: Financial Loss, Reputation Damage, Impact on Encrypted Communications, and Non-Compliance with Security Standards
- Recommendation: Disable the reported weak MAC algorithm
- Screenshots of scan results:

Summary

The remote SSH server is configured to allow / support weak MAC algorithm(s).

Detection Result

```
The remote SSH server supports the following weak client-to-server MAC algorithm(s):

umac-64-etm@openssh.com
umac-64@openssh.com


The remote SSH server supports the following weak server-to-client MAC algorithm(s):

umac-64-etm@openssh.com
umac-64@openssh.com
```

Product Detection Result

Product cpe:/a:ietf:secure_shell_protocol

Method SSH Protocol Algorithms Supported (OID: 1.3.6.1.4.1.25623.1.0.105565)

Log View details of product detection

Detection Method

Checks the supported MAC algorithms (client-to-server and server-to-client) of the remote SSH server.

Currently weak MAC algorithms are defined as the following:

- MD5 based algorithms

- 96-bit based algorithms

- 64-bit based algorithms

- 'none' algorithm

Details: Weak MAC Algorithm(s) Supported (SSH) OID: 1.3.6.1.4.1.25623.1.0.105610

Version used: 2024-06-14T05:05:48Z

Solution

Solution Type: Mitigation
Disable the reported weak MAC algorithm(s).

References

Other https://www.rfc-editor.org/rfc/rfc6668
https://www.rfc-editor.org/rfc/rfc4253#section-6.4

(Applied filter: apply_overrides=0 levels=hml rows=100 min_qod=70 first=1 sort-reverse=severity) 1 - 4 of 4

- IP3: 102.37.157.213
- OpenVAS Plugin ID:
- Vulnerability name: Weak MAC Algorithm (s) Supported (SSH)
- Description: The remote server is configured to allow / support weak MAC algorithms
- CVE: N/A
- OpenVAS Risk Rating: 2.6
- Likelihood: Low
- Consequence: Data Integrity Compromise, Authentication Bypass, Replay Attacks, Man-In-The-Middle Attacks, Cryptographic Attacks
- Impact: Financial Loss, Reputation Damage, Impact on Encrypted Communications, and Non-Compliance with Security Standards
- Recommendation: Disable the reported weak MAC algorithm
- Screenshots of scan results:

1 - 1 of 1

/ulnerability		Severity ▼	QoD	Host IP	Host Name	Location	Created
Veak MAC Algorithm(s) Supported (SSH)		2.6 (Low)	80 %	102.37.157.213		22/tcp	Sat, Sep 7, 2024 11:37 AM UTC

Summary

The remote SSH server is configured to allow / support weak MAC algorithm(s).

Detection Result

```
The remote SSH server supports the following weak client-to-server MAC algorithm(s):

umac-64-etm@openssh.com
umac-64@openssh.com


The remote SSH server supports the following weak server-to-client MAC algorithm(s):

umac-64-etm@openssh.com
umac-64@openssh.com
```

Product Detection Result

Product cpe:/a:ietf:secure_shell_protocol

Method SSH Protocol Algorithms Supported (OID: 1.3.6.1.4.1.25623.1.0.105565)

Log View details of product detection

Detection Method

Checks the supported MAC algorithms (client-to-server and server-to-client) of the remote SSH server.

Currently weak MAC algorithms are defined as the following:

- MD5 based algorithms

- 96-bit based algorithms

- 64-bit based algorithms

- 'none' algorithm

Details: Weak MAC Algorithm(s) Supported (SSH) OID: 1.3.6.1.4.1.25623.1.0.105610

Version used: 2024-06-14T05:05:48Z

Solution

Solution Type: Mitigation
Disable the reported weak MAC algorithm(s).

References

"Informational" findings (list)

- HTTP Server type and version
- Traceroute
- SSL/TLS: Safe/Secure Renegotiation Support Status
- SSL/TLS: Untrusted Certificate Detection
- SSH Server type and version
- MariaDB/Oracle MySQL Detection

Additional Observations

- None

Preliminary Conclusions

- Nothing critical and can be solved easily.

False positives

- Identify scan results that do not indicate a vulnerability risk. Explain how you concluded that there is no risk.

Next Steps

- Report findings and compare notes with team
- Finalize the presentation to the client

Appendix: Documenting vulnerabilities - explanation

- **Vulnerability**: Enter the descriptive name assigned to the vulnerability by the testing tool (Nessus, OpenVas, etc.) This name should provide a clear understanding of the issue.
- **Description**: Provide a brief description of the vulnerability. This information can often be found in the details section of the testing tool's output. The description should explain what the vulnerability is and how it can be exploited.
- **Risk Rating:** Record the overall risk score assigned by the testing tool. This score typically combines the likelihood and consequence of the vulnerability
- **Likelihood:** In this section, students should **estimate** the likelihood of the vulnerability being exploited. They can consider factors such as:
    a. **Prevalence of exploit code:** Is there readily available exploit code for this vulnerability?
    b. **Exploit difficulty:** How technically challenging is it to exploit this vulnerability?
    c. **Value of the target:** Is your system or data a high-value target for attackers?
    d. **Patch availability:** Is a patch available to fix the vulnerability? Students should use their understanding of the vulnerability and the environment to make an informed judgment about the likelihood of exploitation.
- **Consequence:** In this section, students should analyze the **potential consequences** of exploiting the vulnerability. They can consider the same factors mentioned for Impact, but with a focus on the severity of the potential damage:
    a. **Confidentiality:** What sensitive data could be exposed if exploited?
    b. **Integrity:** How critical are the systems or data that could be compromised?
    c. **Availability:** What would be the impact of a system or service outage?

d. **Financial Loss:** Could the vulnerability lead to significant financial losses? By considering these factors, students can assess the potential severity of the consequences if the vulnerability is exploited.

- **CVE:** Enter the unique identifier for the specific vulnerability identified by the testing tool (Nessus, OpenVas, etc.). This will typically be a code like CVE-2023-XXXX
- **CVSS:** Enter the CVSS score assigned by the testing tool. The CVSS score (0.0-10.0) reflects the severity of the vulnerability
- **Recommendation:** Provide recommendations for fixing the vulnerability. This may involve patching the system, disabling the affected service, or taking other steps to mitigate the risk