A Basic Vulnerability Scan using OpenVAS

Introduction

Vulnerability scanning is a foundational process in cybersecurity that helps organizations detect potential weaknesses in their systems before they can be exploited by attackers. The assignment involves performing a vulnerability scan using OpenVAS (Greenbone Vulnerability Manager), an open source and widely used vulnerability assessment tool. The primary objective is to evaluate the security posture of a target machine on a local network by detecting known vulnerabilities, misconfigurations, and weak encryption practices. OpenVAS utilizes regularly updated vulnerability feeds and offers comprehensive reporting, making it suitable for academic, enterprise, and individual use. By conducting this scan, we aim to understand the nature of system vulnerabilities, analyze their severity levels, and recommend necessary mitigation techniques.

Environment Setup (Methodology used to Conduct Scan)

To conduct a successful vulnerability assessment using OpenVAS (Greenbone Vulnerability Manager), it is essential to establish a secure and functional laboratory environment. The setup ensures that all components of the scanning framework work efficiently and that the scan targets are configured properly.

Host System

- Host OS: Kali Linux 2025.2 (64-bit)
- System Requirements:
  - CPU: Dual-core or higher
  - RAM: Minimum 4 GB (8 GB recommended)
  - Disk Space: At least 20 GB free
  - Network: NAT / Bridged Adapter for internal scanning

Tool Installed

- OpenVAS (GVM - Greenbone Vulnerability Manager)
- Interface: Greenbone Security Assistant (Web UI)
- Installation Command (Kali Linux):

```
sudo apt update (figure 1)
sudo apt install openvas -y (figure 2)
sudo gvm-setup (figure 3)
sudo gvm-check-setup (figure 4)
sudo gvm-start (figure 5)
```

Screenshot of Terminal:

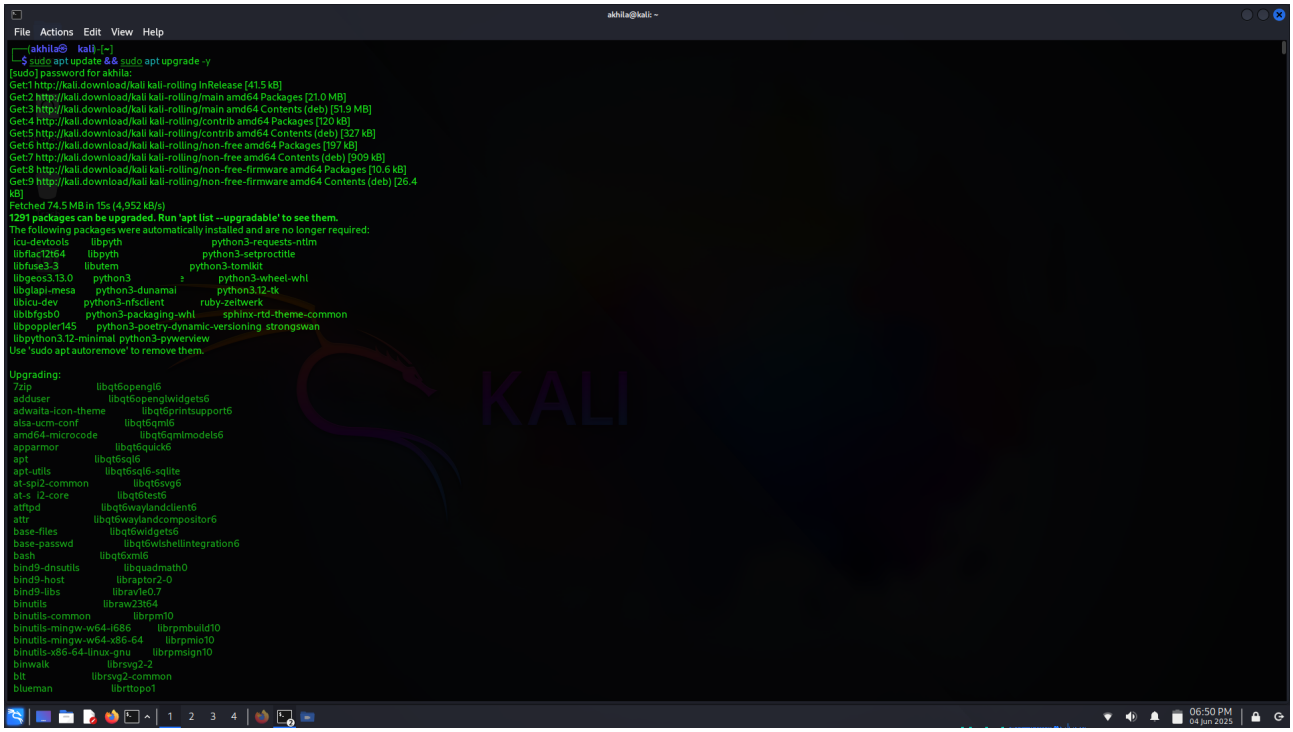

Figure 1: Updating the System

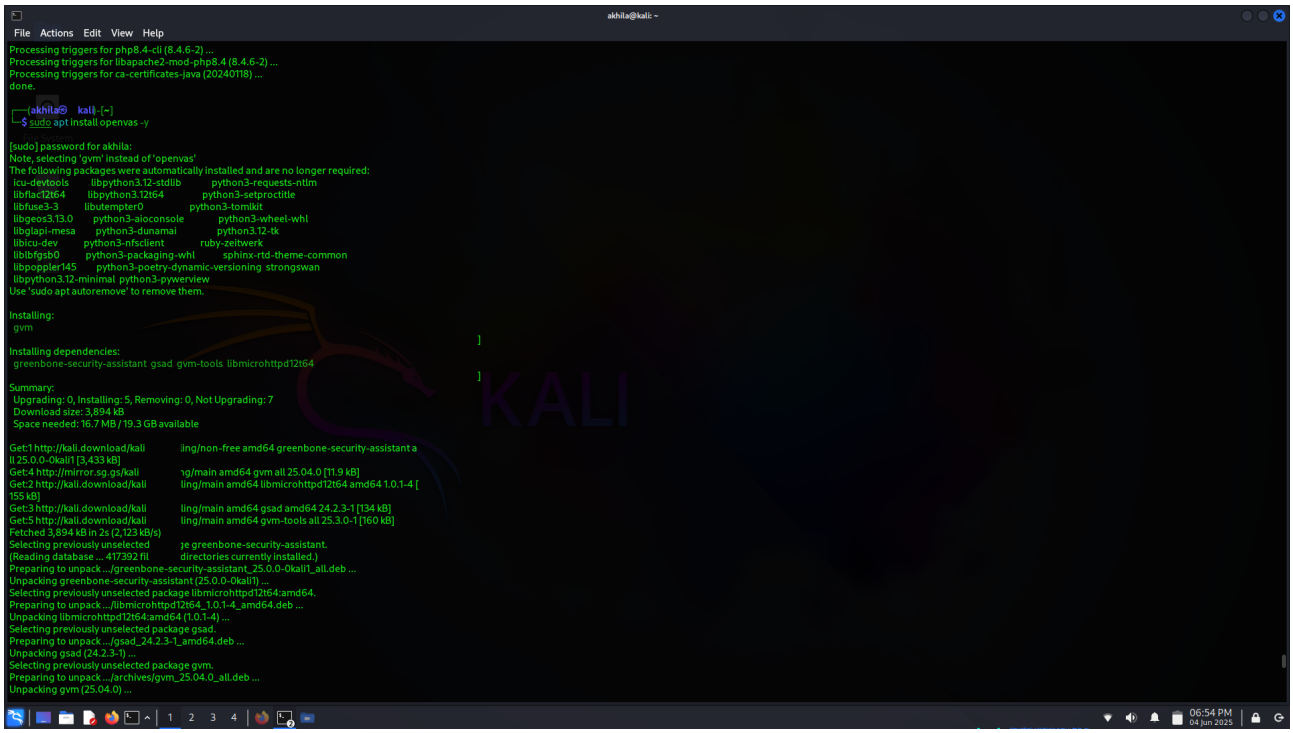

Figure 2: Install OpenVAS and Dependencies

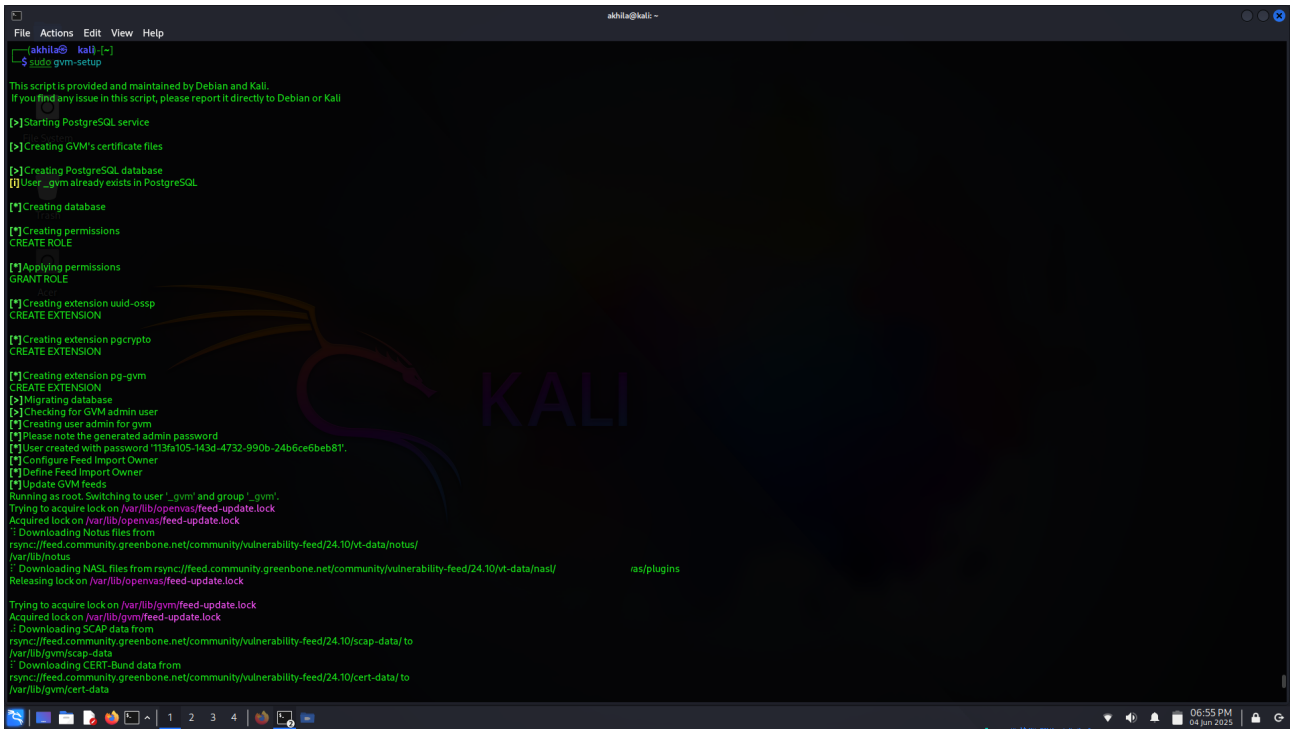

Figure 3: Set Up the OpenVAS Environment

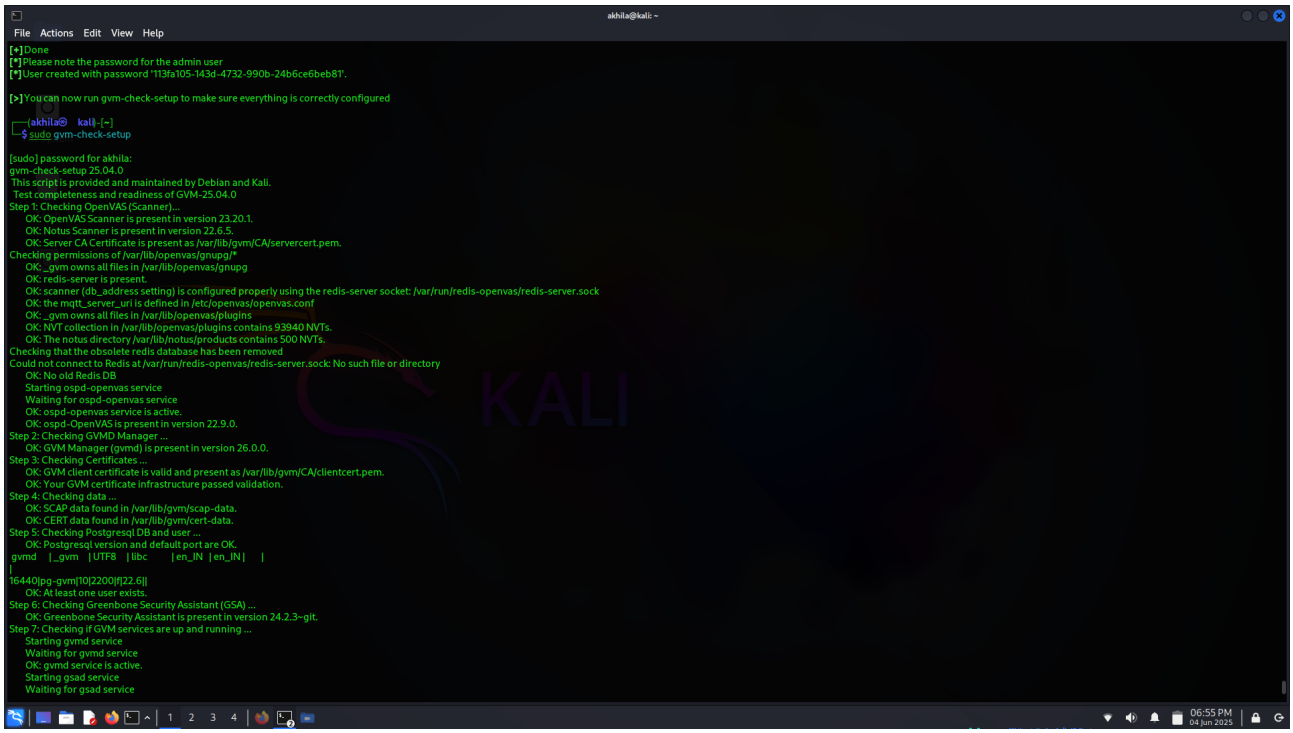

Figure 4: Checking GVM Setup

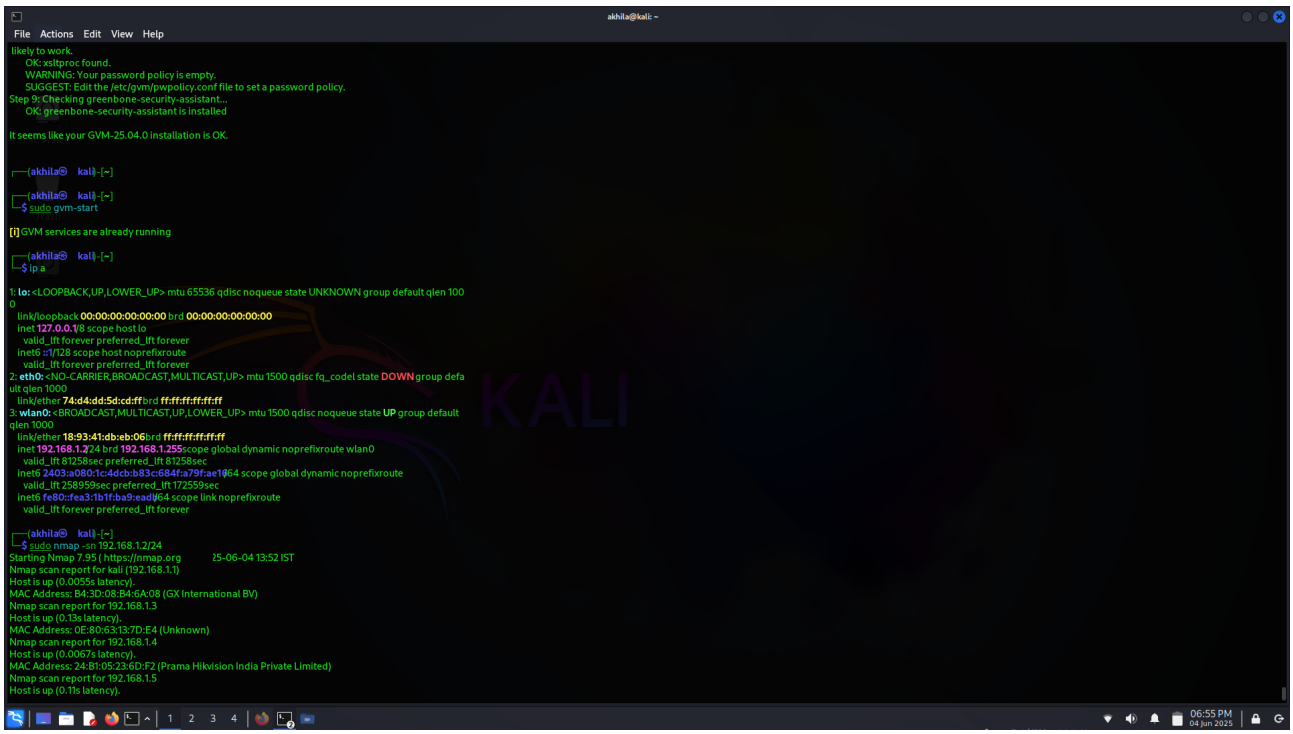

Figure 5: Starting GVM

Step 1: Web UI Access

- Open browser and navigate to:
  https://127.0.0.1:9392
  Accept the SSL warning
  Login using the generated admin credentials. (Shown in figure 6)

Step 2: Run the First Scan

- Add a Target
  Configuration → Targets → New Target
  - Fill in:
    Name: Localhost (or any name)
    Hosts: 192.168.1.1 (or target IP/domain)
    Port List: Leave default (All IANA assigned TCP)

  Click Save. (Shown in figure 7)

- Create a Task
  Scans → Tasks → New Task
  Select the target you just created
  Use default scan config: Full and fast and Save it. (Shown in figure 8)

- Run the Scan
  Click the play button beside the task
  Wait for it to complete. (Shown in figure 9)

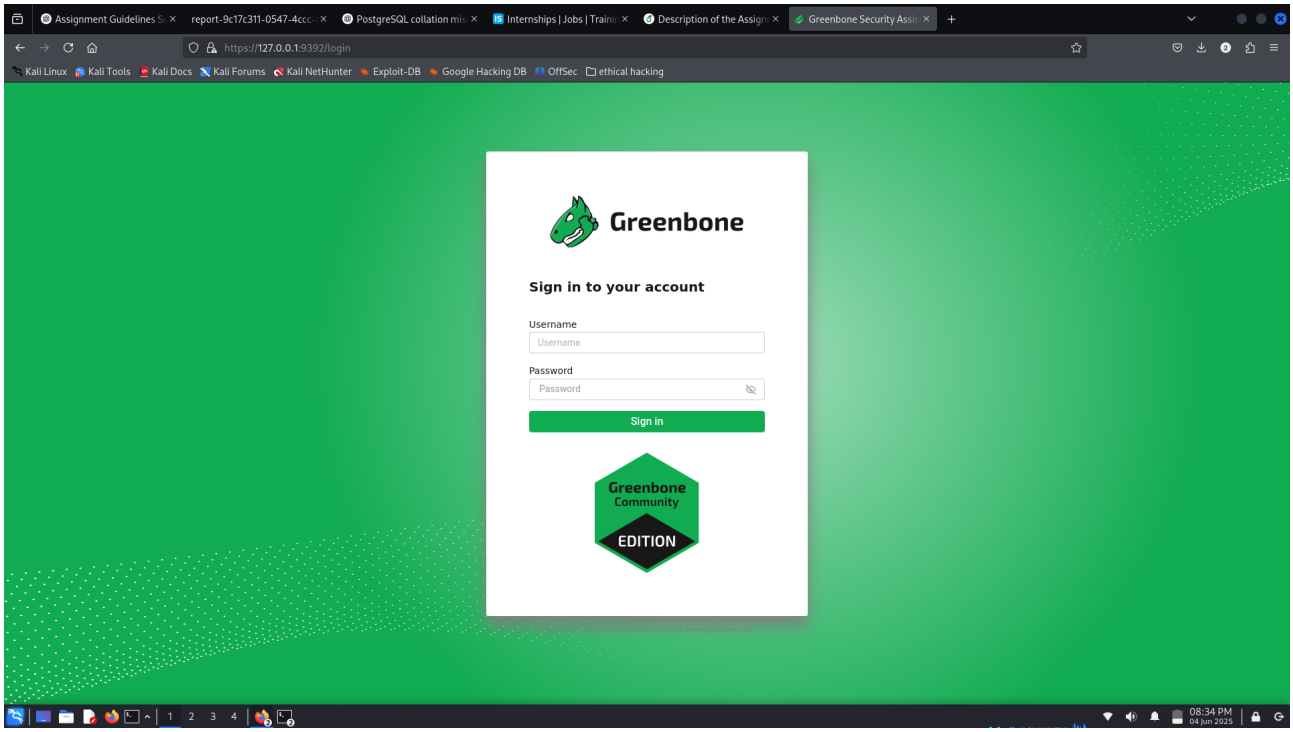

Figure 6: GVM Login Screen

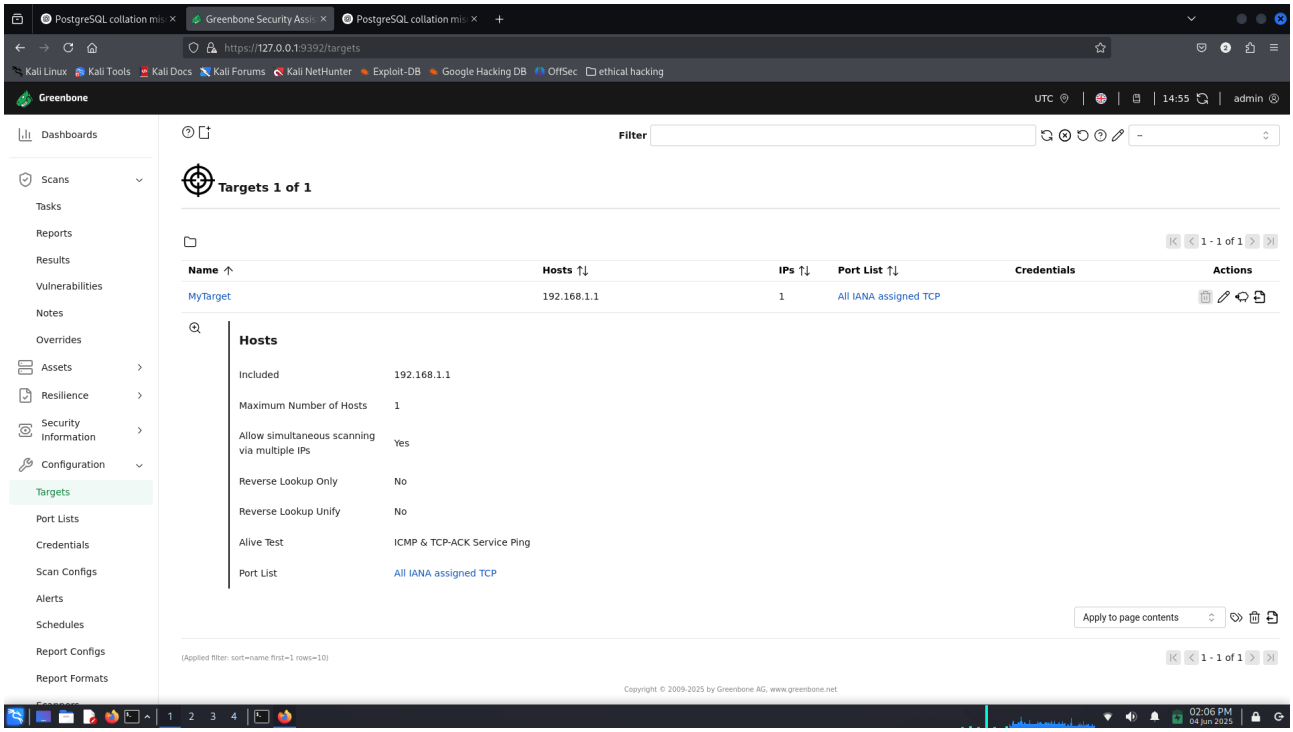

Figure 7: Add a Target

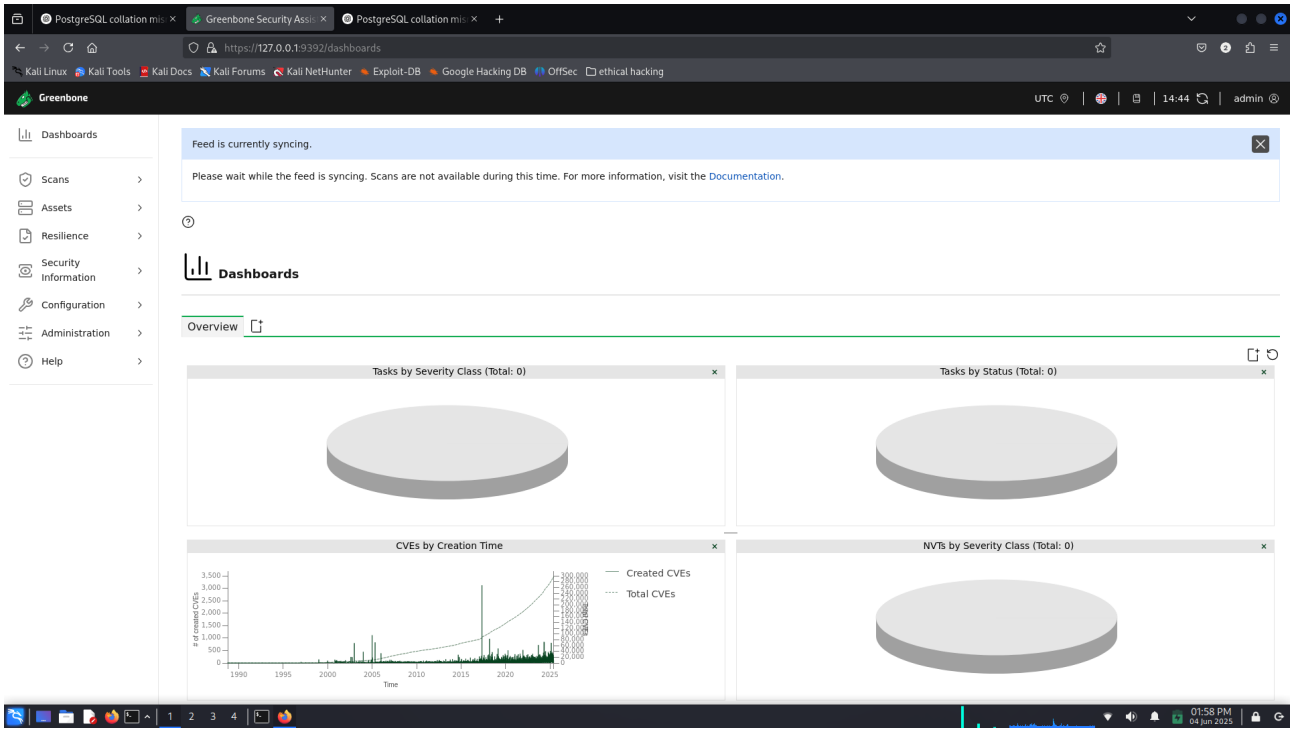

Figure 8: Create a Task

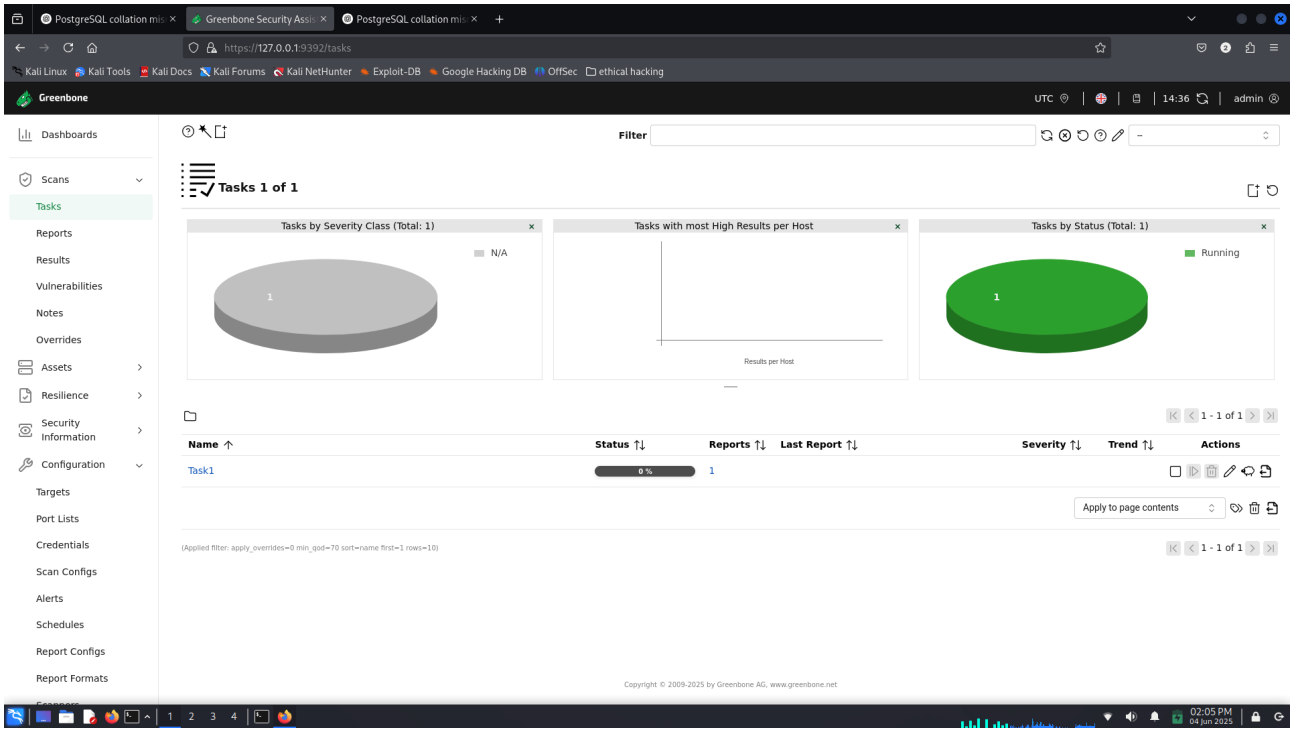

Figure 9: Run the Scan

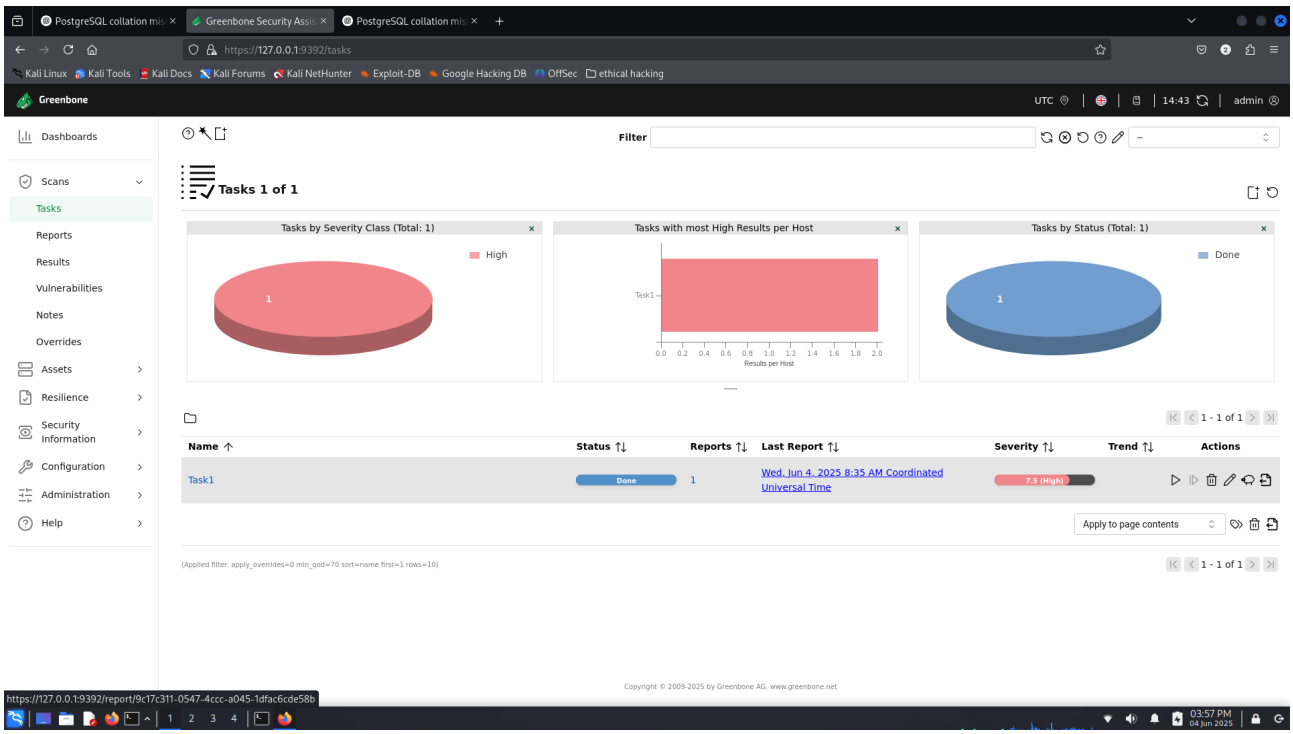

Figure 10: Results

Step 3: View Results

- Once the scan finishes:
  Go to Scans → Reports
  Click the report entry
- We can see a summary:
  - Number of vulnerabilities
  - Severity levels (High, Medium, Low)
  - Affected services/ports
  - Recommended remediation steps

Step 4: Export Reports

- Click on any scan report
- Use the "Export" option (top-right) to save as:
  - PDF
  - XML
  - CSV
  - HTML

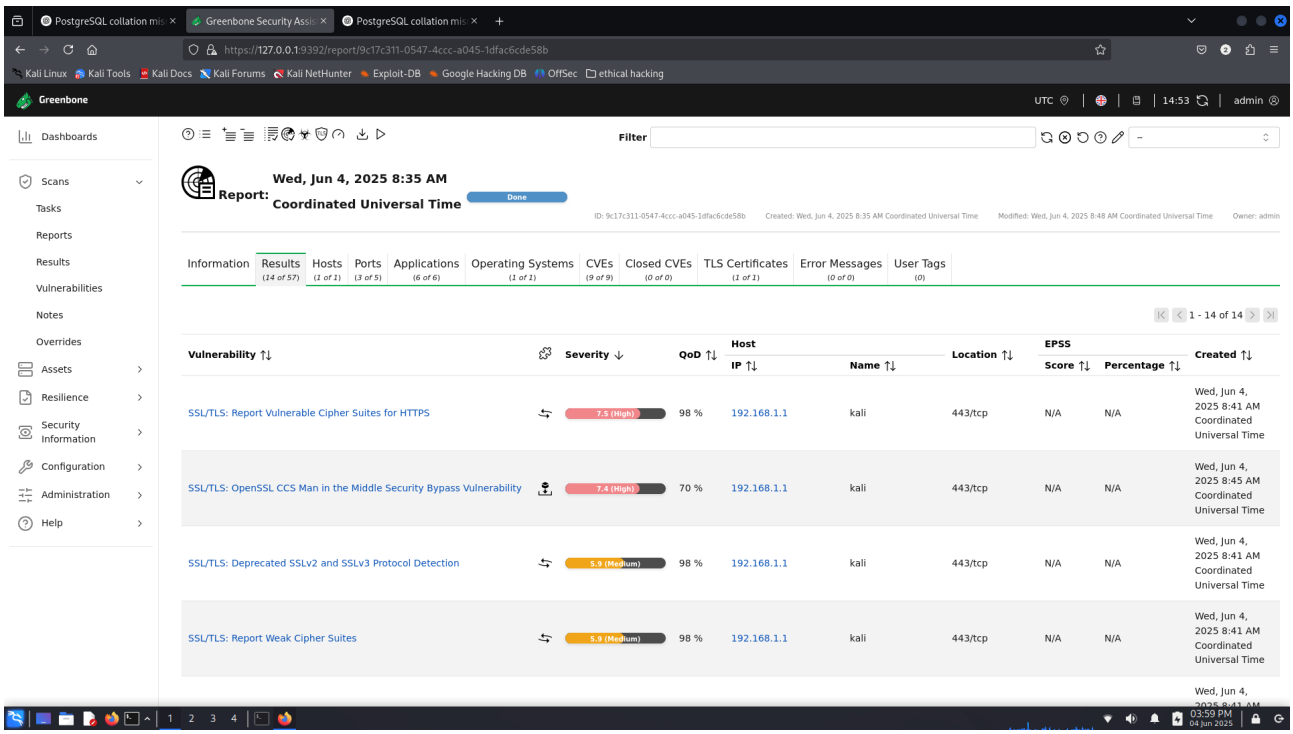

Figure 11: Detailed Report

Key Vulnerabilities Identified During the Scan

The OpenVAS scan on the target system 192.168.1.1 (hostname: kali) uncovered multiple medium to high-severity vulnerabilities, mostly associated with SSL/TLS misconfigurations, outdated protocols, and weak cryptographic practices.

1. SSL/TLS: Vulnerable Cipher Suites for HTTPS

- Severity: 7.5 (High)
- Port: 443/tcp
- Description: The HTTPS service supports cipher suites that are known to be weak or vulnerable to known attacks. This vulnerability highlights that the HTTPS service on the target system (192.168.1.1) accepts insecure cipher suites. These ciphers are cryptographically weak, making the system susceptible to several attacks, such as:
- SWEET32 (CVE-2016-2183): Exploits 64-bit block ciphers like 3DES to recover sensitive information from encrypted sessions.
- Support for outdated algorithms and configurations can increase the risk of:
  - Confidentiality breaches
  - Replay or MITM (Man-in-the-Middle) attacks
  - Non-compliance with industry standards (e.g., PCI-DSS, NIST)
- Impact: If not mitigated, an attacker may Intercept or decrypt secure HTTPS traffic, Exploit known weaknesses in cipher algorithms to obtain sensitive data such as session cookies, login credentials, or private communications.
- Risk: Allows attackers to decrypt or manipulate encrypted traffic.
- Fix: Reconfigure the web server (e.g., Apache, Nginx) to only use strong ciphers (e.g., AES-GCM, TLSv1.2+).

2. OpenSSL CCS Injection (CVE-2014-0224)

- Severity: 7.4 (High)
- Port: 443/tcp
- Description: Vulnerability in OpenSSL allows a man-in-the-middle (MITM) attacker to inject ChangeCipherSpec messages and compromise encrypted communication.
  This vulnerability occurs because OpenSSL fails to properly handle ChangeCipherSpec (CCS) messages, allowing attackers to:
  - Trigger the use of a zero-length master key during the TLS handshake
  - Hijack sessions
  - Perform a man-in-the-middle (MITM) attack to steal or manipulate sensitive communication
- It primarily affects OpenSSL versions:
  - Before 0.9.8za
  - 1.0.0 before 1.0.0m
  - 1.0.1 before 1.0.1h
- Impact: A successful exploit could allow Unauthorized access to encrypted data, Session hijacking and Credential theft or data manipulation during a secure HTTPS connection
- Fix: Update OpenSSL to a patched version ($\geq$ 1.0.1h or $\geq$ 1.0.0m).

3. Weak Cipher Suites for HTTPS

- Severity: 5.9 (Medium)
- Port: 443/tcp
- Description: The HTTPS service accepts weak encryption algorithms like RC4 or 3DES.
  The scan flagged multiple weak cipher suites. Based on cryptographic analysis, these include:
  - RC4 cipher – known vulnerabilities (CVE-2013-2566, CVE-2015-2808)
  - 64-bit ciphers – vulnerable to brute-force attacks (CVE-2015-4000)
  - 1024-bit RSA – considered insecure due to insufficient key length
  - Short-lived ciphers – secure for fewer than 10 years

  Such ciphers no longer meet modern cryptographic standards and can weaken overall HTTPS/SSL security.
- Impact: A system using weak cipher suites may allow Passive eavesdropping (attackers decrypting SSL traffic), Session compromise and Data interception during TLS communication. This compromises confidentiality and trust in HTTPS/SSL communications.
- Fix: Disable weak ciphers in server configuration and enforce modern cryptography.

4. Deprecated SSLv2 and SSLv3 Protocols Enabled

- Severity: 5.9 (Medium)
- Port: 443/tcp
- Description: The SSLv2 and SSLv3 protocols are known to be insecure.
  The following cryptographic vulnerabilities were associated with SSLv2/SSLv3:
  - CVE-2014-3566: [POODLE Attack] – Allows attackers to decrypt HTTPS traffic by exploiting padding oracle vulnerabilities in SSLv3.
  - CVE-2016-0800: [DROWN Attack] – Decrypting RSA using obsolete cryptography through SSLv2 fallback mechanisms.

Since SSLv2 and SSLv3 are outdated, they are no longer patched by vendors and new vulnerabilities may never be fixed.

- Impact: If enabled, an attacker can Eavesdrop secure traffic by downgrading connections to use weak SSL versions. Extract session cookies, credentials, or unencrypted data. Bypass modern encryption protections, especially during handshake downgrades (e.g., POODLE).
- Fix: Disable these protocols in your server configuration.

5. SSH: Weak Key Exchange (KEX) Algorithms

- Severity: 5.3 (Medium)
- Port: 22/tcp
- Description: SSH supports deprecated key exchange methods.
  This issue arises due to the SSH server using weak or outdated key exchange (KEX) methods, especially:
  - 1024-bit MODP Diffie-Hellman groups
  - Ephemeral key exchange using SHA-1
  - RSA with 1024-bit modulus

  These are vulnerable because:
  - Weak primes (like 1024-bit) are breakable by precomputation using advanced techniques like Number Field Sieve.
  - SHA-1 is broken and unsuitable for secure cryptographic use.
  - RSA-1024 has insufficient key length for modern security standards.
- Impact: An attacker with network access (e.g., on the same Wi-Fi or LAN) can Decrypt SSH sessions, Steal credentials or transmitted commands and Impersonate the server (MITM attack). This compromises the confidentiality and integrity of your SSH sessions.
- Fix: Update SSH server config to use only strong KEX algorithms (e.g., diffie-hellman-group-exchange-sha256).

6. Boa Webserver Terminal Escape Sequence Injection

- Severity: 5.0 (Medium)
- Port: 80/tcp
- Description: Malicious input could be interpreted as terminal commands in server logs.
  The vulnerability lies in how the Boa web server logs HTTP requests. It does not sanitize terminal escape sequences that could be included in user-agent strings or URLs.If a system administrator views the logs in a vulnerable terminal (like xterm, gnome-terminal), malicious escape sequences can:
  - Execute arbitrary shell commands
  - Alter terminal settings
  - Redirect output or input streams
- Impact: An attacker sends a specially crafted HTTP request to the web server. The malicious payload is stored in the access log. When the admin views the logs (e.g., using cat or less), the escape sequence gets executed.
- Fix: Update or replace the Boa web server. Sanitize log inputs.

7. SSL/TLS: Renegotiation DoS (CVE-2011-1473, CVE-2011-5094)

- Severity: 5.0 (Medium)
- Port: 443/tcp
- Description: SSL renegotiation may lead to denial-of-service.
  The vulnerability stems from how SSL/TLS renegotiation is handled. In a normal scenario, renegotiation is used to refresh keys or change encryption settings mid-session. However, when client-initiated renegotiation is left unrestricted, it:
  - Allows attackers to trigger multiple renegotiations on the same connection.
  - Consumes server CPU, especially under high load or repeated requests.
  - Can lead to performance degradation or complete unavailability of service.
- Impact: Easy, requires no authentication
  Consequence: Unavailability of services due to excessive SSL renegotiation causing performance drops.
- Fix: Disable renegotiation or upgrade OpenSSL to a secure version.

8. HTTP Transmits Sensitive Data in Cleartext

- Severity: 4.8 (Medium)
- Port: 80/tcp
- Description: HTTP allows sensitive information (like login forms) to be sent unencrypted.
  The vulnerability was discovered by analyzing traffic and input forms on pages served over HTTP, which include:
  - Basic Authentication Headers transmitted in cleartext.
  - Login forms using ¡input type=”password”¿ without being protected by HTTPS.

  When sensitive data is transmitted without encryption, any attacker on the same network (e.g., public Wi-Fi, internal LAN, or via a compromised router) can:
  - Intercept the data using packet sniffers (e.g., Wireshark).
  - Steal credentials and replay authentication requests.
  - Perform session hijacking or impersonation.
- Impact:
  - Exposure of usernames and passwords
  - Increased risk of phishing and credential reuse attacks
  - Non-compliance with security standards (e.g., OWASP A2, GDPR, PCI-DSS)
- Fix: Enforce HTTPS using redirects or HSTS.

9. SSH: Weak Encryption Algorithms Supported

- Severity: 4.3 (Medium)
- Port: 22/tcp
- Description: Insecure encryption algorithms (e.g., cbc, arcfour) are enabled.
  The TLSv1.0 and TLSv1.1 protocols contain known cryptographic flaws like:
  - CVE-2011-3389: Browser Exploit Against SSL/TLS (BEAST)
  - CVE-2015-0204: Factoring Attack on RSA-EXPORT Keys Padding Oracle On Downgraded Legacy Encryption (FREAK)
- Impact: An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.Furthermore newly uncovered vulnerabilities in this protocols won't receive security updates anymore.
- Fix: Limit to strong algorithms such as aes256-ctr.

10. Deprecated TLSv1.0 and TLSv1.1 Protocols Enabled

- Severity: 4.3 (Medium)
- Port: 443/tcp
- Description: TLS versions 1.0 and 1.1 are outdated and no longer recommended.
  The 'arcfour' cipher is the Arcfour stream cipher with 128-bit keys. The Arcfour cipher is believed to be compatible with the RC4 cipher [SCHNEIER]. Arcfour (and RC4) has problems with weak keys, and should not be used anymore.
  The 'none' algorithm specifies that no encryption is to be done. Note that this method provides no confidentiality protection, and it is NOT RECOMMENDED to use it.
  A vulnerability exists in SSH messages that employ CBC mode that may allow an attacker to recover plaintext from a block of ciphertext.
- Fix: Support only TLS 1.2 and above.

Scan Report

June 4, 2025

Summary

This document reports on the results of an automatic security scan. All dates are displayed using the timezone "Coordinated Universal Time", which is abbreviated "UTC". The task was "Task1". The scan started at Wed Jun 4 08:35:38 2025 UTC and ended at Wed Jun 4 08:48:56 2025 UTC. The report first summarises the results found. Then, for each host, the report describes every issue found. Please consider the advice given in each description, in order to rectify the issue.

Contents

1 Result Overview

Host	High	Medium	Low	Log	False Positive
192.168.1.1 kali	2	9	3	0	0
Total: 1	2	9	3	0	0

Vendor security updates are not trusted.
Overrides are off. Even when a result has an override, this report uses the actual threat of the result.
Information on overrides is included in the report.
Notes are included in the report.
This report might not show details of all issues that were found.
Issues with the threat level "Log" are not shown.
Issues with the threat level "Debug" are not shown.
Issues with the threat level "False Positive" are not shown.
Only results with a minimum QoD of 70 are shown.

This report contains all 14 results selected by the filtering described above. Before filtering there were 57 results.

2 Results per Host

2.1 192.168.1.1

Host scan start	Wed Jun 4 08:35:51 2025 UTC
Host scan end	Wed Jun 4 08:48:50 2025 UTC

Service (Port)	Threat Level
443/tcp	High
80/tcp	Medium
443/tcp	Medium
22/tcp	Medium
general/icmp	Low
general/tcp	Low
443/tcp	Low

2.1.1 High 443/tcp

High (CVSS: 7.5)

NVT: SSL/TLS: Report Vulnerable Cipher Suites for HTTPS

. . . continues on next page . . .

... continued from previous page ...

Product detection result
cpe:/a:ietf:transport_layer_security
Detected by SSL/TLS: Report Supported Cipher Suites (OID: 1.3.6.1.4.1.25623.1.0.
↪802067)

Summary
This routine reports all SSL/TLS cipher suites accepted by a service where attack vectors exists only on HTTPS services.

Quality of Detection (QoD): 98%

Vulnerability Detection Result
'Vulnerable' cipher suites accepted by this service via the SSLv3 protocol:
TLS_RSA_WITH_3DES_EDE_CBC_SHA (SWEET32)
TLS_RSA_WITH_DES_CBC_SHA (SWEET32)
'Vulnerable' cipher suites accepted by this service via the TLSv1.0 protocol:
TLS_RSA_WITH_3DES_EDE_CBC_SHA (SWEET32)
TLS_RSA_WITH_DES_CBC_SHA (SWEET32)
'Vulnerable' cipher suites accepted by this service via the TLSv1.1 protocol:
TLS_RSA_WITH_3DES_EDE_CBC_SHA (SWEET32)
TLS_RSA_WITH_DES_CBC_SHA (SWEET32)
'Vulnerable' cipher suites accepted by this service via the TLSv1.2 protocol:
TLS_RSA_WITH_3DES_EDE_CBC_SHA (SWEET32)
TLS_RSA_WITH_DES_CBC_SHA (SWEET32)

Impact
This could allow remote attackers to obtain sensitive information or have other, unspecified impacts.

Solution:
Solution type: Mitigation
The configuration of this services should be changed so that it does not accept the listed cipher suites anymore.
Please see the references for more resources supporting you with this task.

Affected Software/OS
All services accepting vulnerable SSL/TLS cipher suites via HTTPS.

Vulnerability Insight
These rules are applied for the evaluation of the vulnerable cipher suites:
- 64-bit block cipher 3DES vulnerable to the SWEET32 attack (CVE-2016-2183).

Vulnerability Detection Method
Checks previous collected cipher suites.
Details: SSL/TLS: Report Vulnerable Cipher Suites for HTTPS

... continues on next page ...

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.108031
Version used: 2025-03-27T05:38:50Z

Product Detection Result
Product: cpe:/a:ietf:transport_layer_security
Method: SSL/TLS: Report Supported Cipher Suites
OID: 1.3.6.1.4.1.25623.1.0.802067)

References
cve: CVE-2016-2183
cve: CVE-2016-6329
cve: CVE-2020-12872
url: https://ssl-config.mozilla.org
url: https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidel
↪ines/TG02102/BSI-TR-02102-1.html
url: https://www.bsi.bund.de/EN/Themen/Oeffentliche-Verwaltung/Mindeststandards/
↪TLS-Protokoll/TLS-Protokoll_node.html
url: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Technisch
↪eRichtlinien/TR03116/BSI-TR-03116-4.html
url: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Mindeststandards/Mindes
↪tstandard_BSI_TLS_Version_2_4.html
url: https://web.archive.org/web/20240113175943/https://www.bettercrypto.org
url: https://www.enisa.europa.eu/publications/algorithms-key-size-and-parameters
↪-report-2014
url: https://sweet32.info
cert-bund: WID-SEC-2024-1277
cert-bund: WID-SEC-2024-0209
cert-bund: WID-SEC-2024-0064
cert-bund: WID-SEC-2022-2226
cert-bund: WID-SEC-2022-1955
cert-bund: CB-K21/1094
cert-bund: CB-K20/1023
cert-bund: CB-K20/0321
cert-bund: CB-K20/0314
cert-bund: CB-K20/0157
cert-bund: CB-K19/0618
cert-bund: CB-K19/0615
cert-bund: CB-K18/0296
cert-bund: CB-K17/1980
cert-bund: CB-K17/1871
cert-bund: CB-K17/1803
cert-bund: CB-K17/1753
cert-bund: CB-K17/1750
cert-bund: CB-K17/1709
cert-bund: CB-K17/1558
cert-bund: CB-K17/1273

... continues on next page ...

... continued from previous page ...

cert-bund: CB-K17/1202
cert-bund: CB-K17/1196
cert-bund: CB-K17/1055
cert-bund: CB-K17/1026
cert-bund: CB-K17/0939
cert-bund: CB-K17/0917
cert-bund: CB-K17/0915
cert-bund: CB-K17/0877
cert-bund: CB-K17/0796
cert-bund: CB-K17/0724
cert-bund: CB-K17/0661
cert-bund: CB-K17/0657
cert-bund: CB-K17/0582
cert-bund: CB-K17/0581
cert-bund: CB-K17/0506
cert-bund: CB-K17/0504
cert-bund: CB-K17/0467
cert-bund: CB-K17/0345
cert-bund: CB-K17/0098
cert-bund: CB-K17/0089
cert-bund: CB-K17/0086
cert-bund: CB-K17/0082
cert-bund: CB-K16/1837
cert-bund: CB-K16/1830
cert-bund: CB-K16/1635
cert-bund: CB-K16/1630
cert-bund: CB-K16/1624
cert-bund: CB-K16/1622
cert-bund: CB-K16/1500
cert-bund: CB-K16/1465
cert-bund: CB-K16/1307
cert-bund: CB-K16/1296
dfn-cert: DFN-CERT-2025-0041
dfn-cert: DFN-CERT-2021-1618
dfn-cert: DFN-CERT-2021-0775
dfn-cert: DFN-CERT-2021-0770
dfn-cert: DFN-CERT-2021-0274
dfn-cert: DFN-CERT-2020-2141
dfn-cert: DFN-CERT-2020-0368
dfn-cert: DFN-CERT-2019-1455
dfn-cert: DFN-CERT-2019-0068
dfn-cert: DFN-CERT-2018-1296
dfn-cert: DFN-CERT-2018-0323
dfn-cert: DFN-CERT-2017-2070
dfn-cert: DFN-CERT-2017-1954
dfn-cert: DFN-CERT-2017-1885
dfn-cert: DFN-CERT-2017-1831

... continues on next page ...

... continued from previous page ...

dfn-cert: DFN-CERT-2017-1821
dfn-cert: DFN-CERT-2017-1785
dfn-cert: DFN-CERT-2017-1626
dfn-cert: DFN-CERT-2017-1326
dfn-cert: DFN-CERT-2017-1239
dfn-cert: DFN-CERT-2017-1238
dfn-cert: DFN-CERT-2017-1090
dfn-cert: DFN-CERT-2017-1060
dfn-cert: DFN-CERT-2017-0968
dfn-cert: DFN-CERT-2017-0947
dfn-cert: DFN-CERT-2017-0946
dfn-cert: DFN-CERT-2017-0904
dfn-cert: DFN-CERT-2017-0816
dfn-cert: DFN-CERT-2017-0746
dfn-cert: DFN-CERT-2017-0677
dfn-cert: DFN-CERT-2017-0675
dfn-cert: DFN-CERT-2017-0611
dfn-cert: DFN-CERT-2017-0609
dfn-cert: DFN-CERT-2017-0522
dfn-cert: DFN-CERT-2017-0519
dfn-cert: DFN-CERT-2017-0482
dfn-cert: DFN-CERT-2017-0351
dfn-cert: DFN-CERT-2017-0090
dfn-cert: DFN-CERT-2017-0089
dfn-cert: DFN-CERT-2017-0088
dfn-cert: DFN-CERT-2017-0086
dfn-cert: DFN-CERT-2016-1943
dfn-cert: DFN-CERT-2016-1937
dfn-cert: DFN-CERT-2016-1732
dfn-cert: DFN-CERT-2016-1726
dfn-cert: DFN-CERT-2016-1715
dfn-cert: DFN-CERT-2016-1714
dfn-cert: DFN-CERT-2016-1588
dfn-cert: DFN-CERT-2016-1555
dfn-cert: DFN-CERT-2016-1391
dfn-cert: DFN-CERT-2016-1378

High (CVSS: 7.4)

NVT: SSL/TLS: OpenSSL CCS Man in the Middle Security Bypass Vulnerability

Summary
OpenSSL is prone to a security bypass vulnerability.

Quality of Detection (QoD): 70%

... continues on next page ...

... continued from previous page ...

Vulnerability Detection Result
Vulnerability was detected according to the Vulnerability Detection Method.

Impact
Successfully exploiting this issue may allow attackers to obtain sensitive information by conducting a man-in-the-middle attack. This may lead to other attacks.

Solution:
Solution type: VendorFix
Updates are available. Please see the references for more information.

Affected Software/OS
OpenSSL before 0.9.8za, 1.0.0 before 1.0.0m and 1.0.1 before 1.0.1h.

Vulnerability Insight
OpenSSL does not properly restrict processing of ChangeCipherSpec messages, which allows man-in-the-middle attackers to trigger use of a zero-length master key in certain OpenSSL-to-OpenSSL communications, and consequently hijack sessions or obtain sensitive information, via a crafted TLS handshake, aka the 'CCS Injection' vulnerability.

Vulnerability Detection Method
Send two SSL ChangeCipherSpec request and check the response.
Details: `SSL/TLS: OpenSSL CCS Man in the Middle Security Bypass Vulnerability`
OID:1.3.6.1.4.1.25623.1.0.105042
Version used: `2025-01-17T15:39:18Z`

References
`cve: CVE-2014-0224`
`url: https://www.openssl.org/news/secadv/20140605.txt`
`url: http://www.securityfocus.com/bid/67899`
`cert-bund: WID-SEC-2023-0500`
`cert-bund: CB-K15/0567`
`cert-bund: CB-K15/0415`
`cert-bund: CB-K15/0384`
`cert-bund: CB-K15/0080`
`cert-bund: CB-K15/0079`
`cert-bund: CB-K15/0074`
`cert-bund: CB-K14/1617`
`cert-bund: CB-K14/1537`
`cert-bund: CB-K14/1299`
`cert-bund: CB-K14/1297`
`cert-bund: CB-K14/1294`
`cert-bund: CB-K14/1202`
`cert-bund: CB-K14/1174`
`cert-bund: CB-K14/1153`
`cert-bund: CB-K14/0876`

... continues on next page ...

... continued from previous page ...

```
cert-bund: CB-K14/0756
cert-bund: CB-K14/0746
cert-bund: CB-K14/0736
cert-bund: CB-K14/0722
cert-bund: CB-K14/0716
cert-bund: CB-K14/0708
cert-bund: CB-K14/0684
cert-bund: CB-K14/0683
cert-bund: CB-K14/0680
dfn-cert: DFN-CERT-2016-0388
dfn-cert: DFN-CERT-2015-0593
dfn-cert: DFN-CERT-2015-0427
dfn-cert: DFN-CERT-2015-0396
dfn-cert: DFN-CERT-2015-0082
dfn-cert: DFN-CERT-2015-0079
dfn-cert: DFN-CERT-2015-0078
dfn-cert: DFN-CERT-2014-1717
dfn-cert: DFN-CERT-2014-1632
dfn-cert: DFN-CERT-2014-1364
dfn-cert: DFN-CERT-2014-1357
dfn-cert: DFN-CERT-2014-1350
dfn-cert: DFN-CERT-2014-1265
dfn-cert: DFN-CERT-2014-1209
dfn-cert: DFN-CERT-2014-0917
dfn-cert: DFN-CERT-2014-0789
dfn-cert: DFN-CERT-2014-0778
dfn-cert: DFN-CERT-2014-0768
dfn-cert: DFN-CERT-2014-0752
dfn-cert: DFN-CERT-2014-0747
dfn-cert: DFN-CERT-2014-0738
dfn-cert: DFN-CERT-2014-0715
dfn-cert: DFN-CERT-2014-0714
dfn-cert: DFN-CERT-2014-0709
```

[return to 192.168.1.1]

2.1.2 Medium 80/tcp

Medium (CVSS: 5.0)

NVT: Boa Webserver Terminal Escape Sequence in Logs Command Injection Vulnerability

Summary
Boa Webserver is prone to a command-injection vulnerability because it fails to adequately sanitize user-supplied input in logfiles.

... continues on next page ...

... continued from previous page ...

Quality of Detection (QoD): 80%

Vulnerability Detection Result
Vulnerability was detected according to the Vulnerability Detection Method.

Impact
Attackers can exploit this issue to execute arbitrary commands in a terminal.

Solution:
Solution type: WillNotFix
No known solution was made available for at least one year since the disclosure of this vulnerability. Likely none will be provided anymore. General solution options are to upgrade to a newer release, disable respective features, remove the product or replace the product by another one.

Affected Software/OS
Boa Webserver 0.94.14rc21 is vulnerable, other versions may also be affected.

Vulnerability Detection Method
Details: `Boa Webserver Terminal Escape Sequence in Logs Command Injection Vulnerability`
OID:1.3.6.1.4.1.25623.1.0.100443
Version used: `2023-07-28T16:09:07Z`

References
`cve: CVE-2009-4496`
`url: http://www.securityfocus.com/bid/37718`
`url: http://www.securityfocus.com/archive/1/508830`
`dfn-cert: DFN-CERT-2010-0658`

Medium (CVSS: 4.8)

NVT: Cleartext Transmission of Sensitive Information via HTTP

Summary
The host / application transmits sensitive information (username, passwords) in cleartext via HTTP.

Quality of Detection (QoD): 80%

Vulnerability Detection Result
`The following input fields were identified (URL:input name):`
`http://kali/admin/login.asp:password`

Impact

... continues on next page ...

... continued from previous page ...

An attacker could use this situation to compromise or eavesdrop on the HTTP communication between the client and the server using a man-in-the-middle attack to get access to sensitive data like usernames or passwords.

Solution:
Solution type: Workaround
Enforce the transmission of sensitive data via an encrypted SSL/TLS connection. Additionally make sure the host / application is redirecting all users to the secured SSL/TLS connection before allowing to input sensitive data into the mentioned functions.

Affected Software/OS
Hosts / applications which doesn't enforce the transmission of sensitive data via an encrypted SSL/TLS connection.

Vulnerability Detection Method
Evaluate previous collected information and check if the host / application is not enforcing the transmission of sensitive data via an encrypted SSL/TLS connection.
The script is currently checking the following:
- HTTP Basic Authentication (Basic Auth)
- HTTP Forms (e.g. Login) with input field of type 'password'
Details: `Cleartext Transmission of Sensitive Information via HTTP`
OID:1.3.6.1.4.1.25623.1.0.108440
Version used: `2023-09-07T05:05:21Z`

References
`url: https://www.owasp.org/index.php/Top_10_2013-A2-Broken_Authentication_and_Se`
`↪ssion_Management`
`url: https://www.owasp.org/index.php/Top_10_2013-A6-Sensitive_Data_Exposure`
`url: https://cwe.mitre.org/data/definitions/319.html`

[return to 192.168.1.1]

2.1.3 Medium 443/tcp

Medium (CVSS: 5.9)

NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

Product detection result
`cpe:/a:ietf:transport_layer_security:1.0`
`Detected by SSL/TLS: Version Detection (OID: 1.3.6.1.4.1.25623.1.0.105782)`

Summary

... continues on next page ...

... continued from previous page ...

It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

Quality of Detection (QoD): 98%

Vulnerability Detection Result

```
In addition to TLSv1.0+ the service is also providing the deprecated SSLv3 proto
↪col and supports one or more ciphers. Those supported ciphers can be found in
↪the 'SSL/TLS: Report Supported Cipher Suites' (OID: 1.3.6.1.4.1.25623.1.0.8020
↪67) VT.
```

Impact

An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.
Furthermore newly uncovered vulnerabilities in this protocols won't receive security updates anymore.

Solution:
Solution type: Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1.2+ protocols.
Please see the references for more resources supporting you with this task.

Affected Software/OS

All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

Vulnerability Insight

The SSLv2 and SSLv3 protocols contain known cryptographic flaws like:
- CVE-2014-3566: Padding Oracle On Downgraded Legacy Encryption (POODLE)
- CVE-2016-0800: Decrypting RSA with Obsolete and Weakened eNcryption (DROWN)

Vulnerability Detection Method

Checks the used SSL protocols of the services provided by this system.
Details: `SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection`
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: `2025-03-27T05:38:50Z`

Product Detection Result

Product: `cpe:/a:ietf:transport_layer_security:1.0`
Method: `SSL/TLS: Version Detection`
OID: 1.3.6.1.4.1.25623.1.0.105782)

References

`cve: CVE-2016-0800`

... continues on next page ...

... continued from previous page ...

```
cve: CVE-2014-3566
url: https://ssl-config.mozilla.org
url: https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidel
↪ines/TG02102/BSI-TR-02102-1.html
url: https://www.bsi.bund.de/EN/Themen/Oeffentliche-Verwaltung/Mindeststandards/
↪TLS-Protokoll/TLS-Protokoll_node.html
url: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Technisch
↪eRichtlinien/TR03116/BSI-TR-03116-4.html
url: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Mindeststandards/Mindes
↪tstandard_BSI_TLS_Version_2_4.html
url: https://web.archive.org/web/20240113175943/https://www.bettercrypto.org
url: https://www.enisa.europa.eu/publications/algorithms-key-size-and-parameters
↪-report-2014
url: https://drownattack.com
url: https://www.imperialviolet.org/2014/10/14/poodle.html
cert-bund: WID-SEC-2023-0431
cert-bund: WID-SEC-2023-0427
cert-bund: CB-K18/0094
cert-bund: CB-K17/1198
cert-bund: CB-K17/1196
cert-bund: CB-K16/1828
cert-bund: CB-K16/1438
cert-bund: CB-K16/1384
cert-bund: CB-K16/1141
cert-bund: CB-K16/1107
cert-bund: CB-K16/1102
cert-bund: CB-K16/0792
cert-bund: CB-K16/0599
cert-bund: CB-K16/0597
cert-bund: CB-K16/0459
cert-bund: CB-K16/0456
cert-bund: CB-K16/0433
cert-bund: CB-K16/0424
cert-bund: CB-K16/0415
cert-bund: CB-K16/0413
cert-bund: CB-K16/0374
cert-bund: CB-K16/0367
cert-bund: CB-K16/0331
cert-bund: CB-K16/0329
cert-bund: CB-K16/0328
cert-bund: CB-K16/0156
cert-bund: CB-K15/1514
cert-bund: CB-K15/1358
cert-bund: CB-K15/1021
cert-bund: CB-K15/0972
cert-bund: CB-K15/0637
cert-bund: CB-K15/0590
```

... continues on next page ...

... continued from previous page ...

cert-bund: CB-K15/0525
cert-bund: CB-K15/0393
cert-bund: CB-K15/0384
cert-bund: CB-K15/0287
cert-bund: CB-K15/0252
cert-bund: CB-K15/0246
cert-bund: CB-K15/0237
cert-bund: CB-K15/0118
cert-bund: CB-K15/0110
cert-bund: CB-K15/0108
cert-bund: CB-K15/0080
cert-bund: CB-K15/0078
cert-bund: CB-K15/0077
cert-bund: CB-K15/0075
cert-bund: CB-K14/1617
cert-bund: CB-K14/1581
cert-bund: CB-K14/1537
cert-bund: CB-K14/1479
cert-bund: CB-K14/1458
cert-bund: CB-K14/1342
cert-bund: CB-K14/1314
cert-bund: CB-K14/1313
cert-bund: CB-K14/1311
cert-bund: CB-K14/1304
cert-bund: CB-K14/1296
dfn-cert: DFN-CERT-2018-0096
dfn-cert: DFN-CERT-2017-1238
dfn-cert: DFN-CERT-2017-1236
dfn-cert: DFN-CERT-2016-1929
dfn-cert: DFN-CERT-2016-1527
dfn-cert: DFN-CERT-2016-1468
dfn-cert: DFN-CERT-2016-1216
dfn-cert: DFN-CERT-2016-1174
dfn-cert: DFN-CERT-2016-1168
dfn-cert: DFN-CERT-2016-0884
dfn-cert: DFN-CERT-2016-0841
dfn-cert: DFN-CERT-2016-0644
dfn-cert: DFN-CERT-2016-0642
dfn-cert: DFN-CERT-2016-0496
dfn-cert: DFN-CERT-2016-0495
dfn-cert: DFN-CERT-2016-0465
dfn-cert: DFN-CERT-2016-0459
dfn-cert: DFN-CERT-2016-0453
dfn-cert: DFN-CERT-2016-0451
dfn-cert: DFN-CERT-2016-0415
dfn-cert: DFN-CERT-2016-0403
dfn-cert: DFN-CERT-2016-0388

... continues on next page ...

... continued from previous page ...

```
dfn-cert: DFN-CERT-2016-0360
dfn-cert: DFN-CERT-2016-0359
dfn-cert: DFN-CERT-2016-0357
dfn-cert: DFN-CERT-2016-0171
dfn-cert: DFN-CERT-2015-1431
dfn-cert: DFN-CERT-2015-1075
dfn-cert: DFN-CERT-2015-1026
dfn-cert: DFN-CERT-2015-0664
dfn-cert: DFN-CERT-2015-0548
dfn-cert: DFN-CERT-2015-0404
dfn-cert: DFN-CERT-2015-0396
dfn-cert: DFN-CERT-2015-0259
dfn-cert: DFN-CERT-2015-0254
dfn-cert: DFN-CERT-2015-0245
dfn-cert: DFN-CERT-2015-0118
dfn-cert: DFN-CERT-2015-0114
dfn-cert: DFN-CERT-2015-0083
dfn-cert: DFN-CERT-2015-0082
dfn-cert: DFN-CERT-2015-0081
dfn-cert: DFN-CERT-2015-0076
dfn-cert: DFN-CERT-2014-1717
dfn-cert: DFN-CERT-2014-1680
dfn-cert: DFN-CERT-2014-1632
dfn-cert: DFN-CERT-2014-1564
dfn-cert: DFN-CERT-2014-1542
dfn-cert: DFN-CERT-2014-1414
dfn-cert: DFN-CERT-2014-1366
dfn-cert: DFN-CERT-2014-1354
```

Medium (CVSS: 5.9)

NVT: SSL/TLS: Report Weak Cipher Suites

Product detection result

```
cpe:/a:ietf:transport_layer_security
Detected by SSL/TLS: Report Supported Cipher Suites (OID: 1.3.6.1.4.1.25623.1.0.
↪802067)
```

Summary

This routine reports all weak SSL/TLS cipher suites accepted by a service.

Quality of Detection (QoD): 98%

Vulnerability Detection Result

```
'Weak' cipher suites accepted by this service via the SSLv3 protocol:
```

... continues on next page ...

... continued from previous page ...

```
TLS_RSA_WITH_RC4_128_MD5
TLS_RSA_WITH_RC4_128_SHA
TLS_RSA_WITH_SEED_CBC_SHA
'Weak' cipher suites accepted by this service via the TLSv1.0 protocol:
TLS_RSA_WITH_RC4_128_MD5
TLS_RSA_WITH_RC4_128_SHA
TLS_RSA_WITH_SEED_CBC_SHA
'Weak' cipher suites accepted by this service via the TLSv1.1 protocol:
TLS_RSA_WITH_RC4_128_MD5
TLS_RSA_WITH_RC4_128_SHA
TLS_RSA_WITH_SEED_CBC_SHA
'Weak' cipher suites accepted by this service via the TLSv1.2 protocol:
TLS_RSA_WITH_RC4_128_MD5
TLS_RSA_WITH_RC4_128_SHA
TLS_RSA_WITH_SEED_CBC_SHA
```

Impact
This could allow remote attackers to obtain sensitive information or have other, unspecified impacts.

Solution:
Solution type: Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

Affected Software/OS
All services providing an encrypted communication using weak SSL/TLS cipher suites.

Vulnerability Insight
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808)
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000)
- 1024 bit RSA authentication is considered to be insecure and therefore as weak
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

Vulnerability Detection Method
Checks previous collected cipher suites.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `2025-03-27T05:38:50Z`

... continues on next page ...

... continued from previous page ...

Product Detection Result
Product: cpe:/a:ietf:transport_layer_security
Method: SSL/TLS: Report Supported Cipher Suites
OID: 1.3.6.1.4.1.25623.1.0.802067)

References
cve: CVE-2013-2566
cve: CVE-2015-2808
cve: CVE-2015-4000
url: https://ssl-config.mozilla.org
url: https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidel
↪ines/TG02102/BSI-TR-02102-1.html
url: https://www.bsi.bund.de/EN/Themen/Oeffentliche-Verwaltung/Mindeststandards/
↪TLS-Protokoll/TLS-Protokoll_node.html
url: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Technisch
↪eRichtlinien/TR03116/BSI-TR-03116-4.html
url: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Mindeststandards/Mindes
↪tstandard_BSI_TLS_Version_2_4.html
url: https://web.archive.org/web/20240113175943/https://www.bettercrypto.org
url: https://www.enisa.europa.eu/publications/algorithms-key-size-and-parameters
↪-report-2014
cert-bund: CB-K21/0067
cert-bund: CB-K19/0812
cert-bund: CB-K17/1750
cert-bund: CB-K16/1593
cert-bund: CB-K16/1552
cert-bund: CB-K16/1102
cert-bund: CB-K16/0617
cert-bund: CB-K16/0599
cert-bund: CB-K16/0168
cert-bund: CB-K16/0121
cert-bund: CB-K16/0090
cert-bund: CB-K16/0030
cert-bund: CB-K15/1751
cert-bund: CB-K15/1591
cert-bund: CB-K15/1550
cert-bund: CB-K15/1517
cert-bund: CB-K15/1514
cert-bund: CB-K15/1464
cert-bund: CB-K15/1442
cert-bund: CB-K15/1334
cert-bund: CB-K15/1269
cert-bund: CB-K15/1136
cert-bund: CB-K15/1090
cert-bund: CB-K15/1059

... continues on next page ...

... continued from previous page ...

cert-bund: CB-K15/1022
cert-bund: CB-K15/1015
cert-bund: CB-K15/0986
cert-bund: CB-K15/0964
cert-bund: CB-K15/0962
cert-bund: CB-K15/0932
cert-bund: CB-K15/0927
cert-bund: CB-K15/0926
cert-bund: CB-K15/0907
cert-bund: CB-K15/0901
cert-bund: CB-K15/0896
cert-bund: CB-K15/0889
cert-bund: CB-K15/0877
cert-bund: CB-K15/0850
cert-bund: CB-K15/0849
cert-bund: CB-K15/0834
cert-bund: CB-K15/0827
cert-bund: CB-K15/0802
cert-bund: CB-K15/0764
cert-bund: CB-K15/0733
cert-bund: CB-K15/0667
cert-bund: CB-K14/0935
cert-bund: CB-K13/0942
dfn-cert: DFN-CERT-2023-2939
dfn-cert: DFN-CERT-2021-0775
dfn-cert: DFN-CERT-2020-1561
dfn-cert: DFN-CERT-2020-1276
dfn-cert: DFN-CERT-2017-1821
dfn-cert: DFN-CERT-2016-1692
dfn-cert: DFN-CERT-2016-1648
dfn-cert: DFN-CERT-2016-1168
dfn-cert: DFN-CERT-2016-0665
dfn-cert: DFN-CERT-2016-0642
dfn-cert: DFN-CERT-2016-0184
dfn-cert: DFN-CERT-2016-0135
dfn-cert: DFN-CERT-2016-0101
dfn-cert: DFN-CERT-2016-0035
dfn-cert: DFN-CERT-2015-1853
dfn-cert: DFN-CERT-2015-1679
dfn-cert: DFN-CERT-2015-1632
dfn-cert: DFN-CERT-2015-1608
dfn-cert: DFN-CERT-2015-1542
dfn-cert: DFN-CERT-2015-1518
dfn-cert: DFN-CERT-2015-1406
dfn-cert: DFN-CERT-2015-1341
dfn-cert: DFN-CERT-2015-1194
dfn-cert: DFN-CERT-2015-1144

... continues on next page ...

... continued from previous page ...

```
dfn-cert: DFN-CERT-2015-1113
dfn-cert: DFN-CERT-2015-1078
dfn-cert: DFN-CERT-2015-1067
dfn-cert: DFN-CERT-2015-1038
dfn-cert: DFN-CERT-2015-1016
dfn-cert: DFN-CERT-2015-1012
dfn-cert: DFN-CERT-2015-0980
dfn-cert: DFN-CERT-2015-0977
dfn-cert: DFN-CERT-2015-0976
dfn-cert: DFN-CERT-2015-0960
dfn-cert: DFN-CERT-2015-0956
dfn-cert: DFN-CERT-2015-0944
dfn-cert: DFN-CERT-2015-0937
dfn-cert: DFN-CERT-2015-0925
dfn-cert: DFN-CERT-2015-0884
dfn-cert: DFN-CERT-2015-0881
dfn-cert: DFN-CERT-2015-0879
dfn-cert: DFN-CERT-2015-0866
dfn-cert: DFN-CERT-2015-0844
dfn-cert: DFN-CERT-2015-0800
dfn-cert: DFN-CERT-2015-0737
dfn-cert: DFN-CERT-2015-0696
dfn-cert: DFN-CERT-2014-0977
```

Medium (CVSS: 5.0)

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

Summary

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

Quality of Detection (QoD): 70%

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
-------------------------------------------------------------------------------
↪-------------------------------------------------
TLSv1.0          | 10
TLSv1.1          | 10
TLSv1.2          | 10
```

Impact

... continues on next page ...

... continued from previous page ...

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.

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

Vulnerability 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.

Vulnerability 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: `2024-09-27T05:05:23Z`

References
`cve: CVE-2011-1473`
`cve: CVE-2011-5094`
`url: https://web.archive.org/web/20211201133213/https://orchilles.com/ssl-renego`
`↪tiation-dos/`
`url: https://mailarchive.ietf.org/arch/msg/tls/wdg46VE_jkYBbgJ5yE4P9nQ-8IU/`
`url: https://vincent.bernat.ch/en/blog/2011-ssl-dos-mitigation`
`url: https://www.openwall.com/lists/oss-security/2011/07/08/2`
`cert-bund: WID-SEC-2024-1591`
`cert-bund: WID-SEC-2024-0796`
`cert-bund: WID-SEC-2023-1435`
`cert-bund: CB-K17/0980`
`cert-bund: CB-K17/0979`
`cert-bund: CB-K14/0772`
`cert-bund: CB-K13/0915`
`cert-bund: CB-K13/0462`
`dfn-cert: DFN-CERT-2025-0933`
`dfn-cert: DFN-CERT-2017-1013`

... continues on next page ...

... continued from previous page ...

```
dfn-cert: DFN-CERT-2017-1012
dfn-cert: DFN-CERT-2014-0809
dfn-cert: DFN-CERT-2013-1928
dfn-cert: DFN-CERT-2012-1112
```

Medium (CVSS: 4.3)

NVT: SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection

Product detection result

```
cpe:/a:ietf:transport_layer_security:1.0
Detected by SSL/TLS: Version Detection (OID: 1.3.6.1.4.1.25623.1.0.105782)
```

Summary

It was possible to detect the usage of the deprecated TLSv1.0 and/or TLSv1.1 protocol on this system.

Quality of Detection (QoD): 98%

Vulnerability Detection Result

```
In addition to TLSv1.2+ the service is also providing the deprecated TLSv1.0 and
↪ TLSv1.1 protocols and supports one or more ciphers. Those supported ciphers c
↪an be found in the 'SSL/TLS: Report Supported Cipher Suites' (OID: 1.3.6.1.4.1
↪.25623.1.0.802067) VT.
```

Impact

An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.
Furthermore newly uncovered vulnerabilities in this protocols won't receive security updates anymore.

Solution:
Solution type: Mitigation
It is recommended to disable the deprecated TLSv1.0 and/or TLSv1.1 protocols in favor of the TLSv1.2+ protocols.
Please see the references for more resources supporting you with this task.

Affected Software/OS

- All services providing an encrypted communication using the TLSv1.0 and/or TLSv1.1 protocols
- CVE-2023-41928: Kiloview P1 4G and P2 4G Video Encoder
- CVE-2024-41270: Gorush v1.18.4
- CVE-2025-3200: Multiple products from Wiesemann & Theis

... continues on next page ...

... continued from previous page ...

Vulnerability Insight
The TLSv1.0 and TLSv1.1 protocols contain known cryptographic flaws like:
- CVE-2011-3389: Browser Exploit Against SSL/TLS (BEAST)
- CVE-2015-0204: Factoring Attack on RSA-EXPORT Keys Padding Oracle On Downgraded Legacy Encryption (FREAK)

Vulnerability Detection Method
Checks the used TLS protocols of the services provided by this system.
Details: SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection
OID:1.3.6.1.4.1.25623.1.0.117274
Version used: 2025-04-30T05:39:51Z

Product Detection Result
Product: cpe:/a:ietf:transport_layer_security:1.0
Method: SSL/TLS: Version Detection
OID: 1.3.6.1.4.1.25623.1.0.105782)

References
cve: CVE-2011-3389
cve: CVE-2015-0204
cve: CVE-2023-41928
cve: CVE-2024-41270
cve: CVE-2025-3200
url: https://ssl-config.mozilla.org
url: https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidel
↪ines/TG02102/BSI-TR-02102-1.html
url: https://www.bsi.bund.de/EN/Themen/Oeffentliche-Verwaltung/Mindeststandards/
↪TLS-Protokoll/TLS-Protokoll_node.html
url: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Technisch
↪eRichtlinien/TR03116/BSI-TR-03116-4.html
url: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Mindeststandards/Mindes
↪tstandard_BSI_TLS_Version_2_4.html
url: https://web.archive.org/web/20240113175943/https://www.bettercrypto.org
url: https://www.enisa.europa.eu/publications/algorithms-key-size-and-parameters
↪-report-2014
url: https://datatracker.ietf.org/doc/rfc8996/
url: https://vnhacker.blogspot.com/2011/09/beast.html
url: https://web.archive.org/web/20201108095603/https://censys.io/blog/freak
url: https://certvde.com/en/advisories/VDE-2025-031/
url: https://gist.github.com/nyxfqq/cfae38fada582a0f576d154be1aeb1fc
url: https://advisories.ncsc.nl/advisory?id=NCSC-2024-0273
cert-bund: WID-SEC-2023-1435
cert-bund: CB-K18/0799
cert-bund: CB-K16/1289
cert-bund: CB-K16/1096
cert-bund: CB-K15/1751

... continues on next page ...

... continued from previous page ...

cert-bund: CB-K15/1266
cert-bund: CB-K15/0850
cert-bund: CB-K15/0764
cert-bund: CB-K15/0720
cert-bund: CB-K15/0548
cert-bund: CB-K15/0526
cert-bund: CB-K15/0509
cert-bund: CB-K15/0493
cert-bund: CB-K15/0384
cert-bund: CB-K15/0365
cert-bund: CB-K15/0364
cert-bund: CB-K15/0302
cert-bund: CB-K15/0192
cert-bund: CB-K15/0079
cert-bund: CB-K15/0016
cert-bund: CB-K14/1342
cert-bund: CB-K14/0231
cert-bund: CB-K13/0845
cert-bund: CB-K13/0796
cert-bund: CB-K13/0790
dfn-cert: DFN-CERT-2020-0177
dfn-cert: DFN-CERT-2020-0111
dfn-cert: DFN-CERT-2019-0068
dfn-cert: DFN-CERT-2018-1441
dfn-cert: DFN-CERT-2018-1408
dfn-cert: DFN-CERT-2016-1372
dfn-cert: DFN-CERT-2016-1164
dfn-cert: DFN-CERT-2016-0388
dfn-cert: DFN-CERT-2015-1853
dfn-cert: DFN-CERT-2015-1332
dfn-cert: DFN-CERT-2015-0884
dfn-cert: DFN-CERT-2015-0800
dfn-cert: DFN-CERT-2015-0758
dfn-cert: DFN-CERT-2015-0567
dfn-cert: DFN-CERT-2015-0544
dfn-cert: DFN-CERT-2015-0530
dfn-cert: DFN-CERT-2015-0396
dfn-cert: DFN-CERT-2015-0375
dfn-cert: DFN-CERT-2015-0374
dfn-cert: DFN-CERT-2015-0305
dfn-cert: DFN-CERT-2015-0199
dfn-cert: DFN-CERT-2015-0079
dfn-cert: DFN-CERT-2015-0021
dfn-cert: DFN-CERT-2014-1414
dfn-cert: DFN-CERT-2013-1847
dfn-cert: DFN-CERT-2013-1792
dfn-cert: DFN-CERT-2012-1979

... continues on next page ...

... continued from previous page ...

dfn-cert: DFN-CERT-2012-1829
dfn-cert: DFN-CERT-2012-1530
dfn-cert: DFN-CERT-2012-1380
dfn-cert: DFN-CERT-2012-1377
dfn-cert: DFN-CERT-2012-1292
dfn-cert: DFN-CERT-2012-1214
dfn-cert: DFN-CERT-2012-1213
dfn-cert: DFN-CERT-2012-1180
dfn-cert: DFN-CERT-2012-1156
dfn-cert: DFN-CERT-2012-1155
dfn-cert: DFN-CERT-2012-1039
dfn-cert: DFN-CERT-2012-0956
dfn-cert: DFN-CERT-2012-0908
dfn-cert: DFN-CERT-2012-0868
dfn-cert: DFN-CERT-2012-0867
dfn-cert: DFN-CERT-2012-0848
dfn-cert: DFN-CERT-2012-0838
dfn-cert: DFN-CERT-2012-0776
dfn-cert: DFN-CERT-2012-0722
dfn-cert: DFN-CERT-2012-0638
dfn-cert: DFN-CERT-2012-0627
dfn-cert: DFN-CERT-2012-0451
dfn-cert: DFN-CERT-2012-0418
dfn-cert: DFN-CERT-2012-0354
dfn-cert: DFN-CERT-2012-0234
dfn-cert: DFN-CERT-2012-0221
dfn-cert: DFN-CERT-2012-0177
dfn-cert: DFN-CERT-2012-0170
dfn-cert: DFN-CERT-2012-0146
dfn-cert: DFN-CERT-2012-0142
dfn-cert: DFN-CERT-2012-0126
dfn-cert: DFN-CERT-2012-0123
dfn-cert: DFN-CERT-2012-0095
dfn-cert: DFN-CERT-2012-0051
dfn-cert: DFN-CERT-2012-0047
dfn-cert: DFN-CERT-2012-0021
dfn-cert: DFN-CERT-2011-1953
dfn-cert: DFN-CERT-2011-1946
dfn-cert: DFN-CERT-2011-1844
dfn-cert: DFN-CERT-2011-1826
dfn-cert: DFN-CERT-2011-1774
dfn-cert: DFN-CERT-2011-1743
dfn-cert: DFN-CERT-2011-1738
dfn-cert: DFN-CERT-2011-1706
dfn-cert: DFN-CERT-2011-1628
dfn-cert: DFN-CERT-2011-1627
dfn-cert: DFN-CERT-2011-1619

... continues on next page ...

... continued from previous page ...

dfn-cert: DFN-CERT-2011-1482

Medium (CVSS: 4.0)

NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

Summary
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

Quality of Detection (QoD): 80%

Vulnerability Detection Result

```
The following certificates are part of the certificate chain but using insecure
↪signature algorithms:
Subject:              1.2.840.113549.1.9.1=#3139322E3136382E312E31,CN=192.168.1.
↪1,OU=realtek,O=realtek,L=suzhou,ST=Jiangsu,C=CN
Signature Algorithm:  sha1WithRSAEncryption
```

Solution:
Solution type: Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

Vulnerability Insight
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1, Fingerprint2

Vulnerability Detection Method
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

... continues on next page ...

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.105880
Version used: 2021-10-15T11:13:32Z

References
url: https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/

[return to 192.168.1.1]

2.1.4 Medium 22/tcp

Medium (CVSS: 5.3)

NVT: Weak Key Exchange (KEX) Algorithm(s) Supported (SSH)

Product detection result
cpe:/a:ietf:secure_shell_protocol
Detected by SSH Protocol Algorithms Supported (OID: 1.3.6.1.4.1.25623.1.0.105565)

Summary
The remote SSH server is configured to allow / support weak key exchange (KEX) algorithm(s).

Quality of Detection (QoD): 80%

Vulnerability Detection Result
The remote SSH server supports the following weak KEX algorithm(s):

KEX algorithm	Reason
diffie-hellman-group1-sha1	Using Oakley Group 2 (a 1024-bit MODP group) and SHA-1

Impact
An attacker can quickly break individual connections.

Solution:
Solution type: Mitigation
Disable the reported weak KEX algorithm(s)
- 1024-bit MODP group / prime KEX algorithms:
Alternatively use elliptic-curve Diffie-Hellmann in general, e.g. Curve 25519.

Vulnerability Insight

... continues on next page ...

... continued from previous page ...

- 1024-bit MODP group / prime KEX algorithms:
Millions of HTTPS, SSH, and VPN servers all use the same prime numbers for Diffie-Hellman key exchange. Practitioners believed this was safe as long as new key exchange messages were generated for every connection. However, the first step in the number field sieve-the most efficient algorithm for breaking a Diffie-Hellman connection-is dependent only on this prime.
A nation-state can break a 1024-bit prime.

Vulnerability Detection Method
Checks the supported KEX algorithms of the remote SSH server.
Currently weak KEX algorithms are defined as the following:
- non-elliptic-curve Diffie-Hellmann (DH) KEX algorithms with 1024-bit MODP group / prime
- ephemerally generated key exchange groups uses SHA-1
- using RSA 1024-bit modulus key
Details: `Weak Key Exchange (KEX) Algorithm(s) Supported (SSH)`
OID:1.3.6.1.4.1.25623.1.0.150713
Version used: `2024-06-14T05:05:48Z`

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)

References
`url: https://weakdh.org/sysadmin.html`
`url: https://www.rfc-editor.org/rfc/rfc9142`
`url: https://www.rfc-editor.org/rfc/rfc9142#name-summary-guidance-for-implem`
`url: https://www.rfc-editor.org/rfc/rfc6194`
`url: https://www.rfc-editor.org/rfc/rfc4253#section-6.5`

Medium (CVSS: 4.3)

NVT: Weak Encryption Algorithm(s) Supported (SSH)

Product detection result
`cpe:/a:ietf:secure_shell_protocol`
`Detected by SSH Protocol Algorithms Supported (OID: 1.3.6.1.4.1.25623.1.0.105565`
`↪)`

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

Quality of Detection (QoD): 80%

Vulnerability Detection Result

... continues on next page ...

... continued from previous page ...

```
The remote SSH server supports the following weak client-to-server encryption al
↪gorithm(s):
3des-cbc
The remote SSH server supports the following weak server-to-client encryption al
↪gorithm(s):
3des-cbc
```

Solution:
Solution type: Mitigation
Disable the reported weak encryption algorithm(s).

Vulnerability Insight
- The 'arcfour' cipher is the Arcfour stream cipher with 128-bit keys. The Arcfour cipher is believed to be compatible with the RC4 cipher [SCHNEIER]. Arcfour (and RC4) has problems with weak keys, and should not be used anymore.
- The 'none' algorithm specifies that no encryption is to be done. Note that this method provides no confidentiality protection, and it is NOT RECOMMENDED to use it.
- A vulnerability exists in SSH messages that employ CBC mode that may allow an attacker to recover plaintext from a block of ciphertext.

Vulnerability Detection Method
Checks the supported encryption algorithms (client-to-server and server-to-client) of the remote SSH server.
Currently weak encryption algorithms are defined as the following:
- Arcfour (RC4) cipher based algorithms
- 'none' algorithm
- CBC mode cipher based algorithms
Details: `Weak Encryption Algorithm(s) Supported (SSH)`
OID:1.3.6.1.4.1.25623.1.0.105611
Version used: `2024-06-14T05:05:48Z`

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)

References
`url: https://www.rfc-editor.org/rfc/rfc8758`
`url: https://www.kb.cert.org/vuls/id/958563`
`url: https://www.rfc-editor.org/rfc/rfc4253#section-6.3`

[return to 192.168.1.1]

2.1.5 Low general/icmp

Low (CVSS: 2.1)

NVT: ICMP Timestamp Reply Information Disclosure

Summary
The remote host responded to an ICMP timestamp request.

Quality of Detection (QoD): 80%

Vulnerability Detection Result
```
The following response / ICMP packet has been received:
- ICMP Type: 14
- ICMP Code: 0
```

Impact
This information could theoretically be used to exploit weak time-based random number generators in other services.

Solution:
Solution type: Mitigation
Various mitigations are possible:
- Disable the support for ICMP timestamp on the remote host completely
- Protect the remote host by a firewall, and block ICMP packets passing through the firewall in either direction (either completely or only for untrusted networks)

Vulnerability Insight
The Timestamp Reply is an ICMP message which replies to a Timestamp message. It consists of the originating timestamp sent by the sender of the Timestamp as well as a receive timestamp and a transmit timestamp.

Vulnerability Detection Method
Sends an ICMP Timestamp (Type 13) request and checks if a Timestamp Reply (Type 14) is received.
Details: `ICMP Timestamp Reply Information Disclosure`
OID:1.3.6.1.4.1.25623.1.0.103190
Version used: `2025-01-21T05:37:33Z`

References
```
cve: CVE-1999-0524
url: https://datatracker.ietf.org/doc/html/rfc792
url: https://datatracker.ietf.org/doc/html/rfc2780
cert-bund: CB-K15/1514
cert-bund: CB-K14/0632
dfn-cert: DFN-CERT-2014-0658
```

[return to 192.168.1.1]

2.1.6 Low general/tcp

Low (CVSS: 2.6)

NVT: TCP Timestamps Information Disclosure

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

Quality of Detection (QoD): 80%

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: 7147642
Packet 2: 7147751
```

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

Solution:
Solution type: Mitigation
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.

Affected Software/OS
TCP implementations that implement RFC1323/RFC7323.

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 Information Disclosure`
OID:1.3.6.1.4.1.25623.1.0.80091
Version used: `2023-12-15T16:10:08Z`

References
`url: https://datatracker.ietf.org/doc/html/rfc1323`

... continues on next page ...

... continued from previous page ...

```
url: https://datatracker.ietf.org/doc/html/rfc7323
url: https://web.archive.org/web/20151213072445/http://www.microsoft.com/en-us/d
↪ownload/details.aspx?id=9152
url: https://www.fortiguard.com/psirt/FG-IR-16-090
```

[return to 192.168.1.1]

2.1.7 Low 443/tcp

Low (CVSS: 3.4)

NVT: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE)

Product detection result

```
cpe:/a:ietf:transport_layer_security
Detected by SSL/TLS: Report Supported Cipher Suites (OID: 1.3.6.1.4.1.25623.1.0.
↪802067)
```

Summary

This host is prone to an information disclosure vulnerability.

Quality of Detection (QoD): 80%

Vulnerability Detection Result

Vulnerability was detected according to the Vulnerability Detection Method.

Impact

Successful exploitation will allow a man-in-the-middle attackers gain access to the plain text data stream.

Solution:

Solution type: Mitigation

Possible Mitigations are:

- Disable SSLv3
- Disable cipher suites supporting CBC cipher modes
- Enable TLS_FALLBACK_SCSV if the service is providing TLSv1.0+

Vulnerability Insight

The flaw is due to the block cipher padding not being deterministic and not covered by the Message Authentication Code

Vulnerability Detection Method

... continues on next page ...

... continued from previous page ...

Evaluate previous collected information about this service.
Details: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability .
↪..
OID:1.3.6.1.4.1.25623.1.0.802087
Version used: 2024-09-30T08:38:05Z

Product Detection Result
Product: cpe:/a:ietf:transport_layer_security
Method: SSL/TLS: Report Supported Cipher Suites
OID: 1.3.6.1.4.1.25623.1.0.802067)

References
cve: CVE-2014-3566
url: https://www.openssl.org/~bodo/ssl-poodle.pdf
url: http://www.securityfocus.com/bid/70574
url: https://www.imperialviolet.org/2014/10/14/poodle.html
url: https://www.dfranke.us/posts/2014-10-14-how-poodle-happened.html
url: http://googleonlinesecurity.blogspot.in/2014/10/this-poodle-bites-exploitin
↪g-ssl-30.html
cert-bund: WID-SEC-2023-0431
cert-bund: CB-K17/1198
cert-bund: CB-K17/1196
cert-bund: CB-K16/1828
cert-bund: CB-K16/1438
cert-bund: CB-K16/1384
cert-bund: CB-K16/1102
cert-bund: CB-K16/0599
cert-bund: CB-K16/0156
cert-bund: CB-K15/1514
cert-bund: CB-K15/1358
cert-bund: CB-K15/1021
cert-bund: CB-K15/0972
cert-bund: CB-K15/0637
cert-bund: CB-K15/0590
cert-bund: CB-K15/0525
cert-bund: CB-K15/0393
cert-bund: CB-K15/0384
cert-bund: CB-K15/0287
cert-bund: CB-K15/0252
cert-bund: CB-K15/0246
cert-bund: CB-K15/0237
cert-bund: CB-K15/0118
cert-bund: CB-K15/0110
cert-bund: CB-K15/0108
cert-bund: CB-K15/0080
cert-bund: CB-K15/0078

... continues on next page ...

... continued from previous page ...

```
cert-bund: CB-K15/0077
cert-bund: CB-K15/0075
cert-bund: CB-K14/1617
cert-bund: CB-K14/1581
cert-bund: CB-K14/1537
cert-bund: CB-K14/1479
cert-bund: CB-K14/1458
cert-bund: CB-K14/1342
cert-bund: CB-K14/1314
cert-bund: CB-K14/1313
cert-bund: CB-K14/1311
cert-bund: CB-K14/1304
cert-bund: CB-K14/1296
dfn-cert: DFN-CERT-2017-1238
dfn-cert: DFN-CERT-2017-1236
dfn-cert: DFN-CERT-2016-1929
dfn-cert: DFN-CERT-2016-1527
dfn-cert: DFN-CERT-2016-1468
dfn-cert: DFN-CERT-2016-1168
dfn-cert: DFN-CERT-2016-0884
dfn-cert: DFN-CERT-2016-0642
dfn-cert: DFN-CERT-2016-0388
dfn-cert: DFN-CERT-2016-0171
dfn-cert: DFN-CERT-2015-1431
dfn-cert: DFN-CERT-2015-1075
dfn-cert: DFN-CERT-2015-1026
dfn-cert: DFN-CERT-2015-0664
dfn-cert: DFN-CERT-2015-0548
dfn-cert: DFN-CERT-2015-0404
dfn-cert: DFN-CERT-2015-0396
dfn-cert: DFN-CERT-2015-0259
dfn-cert: DFN-CERT-2015-0254
dfn-cert: DFN-CERT-2015-0245
dfn-cert: DFN-CERT-2015-0118
dfn-cert: DFN-CERT-2015-0114
dfn-cert: DFN-CERT-2015-0083
dfn-cert: DFN-CERT-2015-0082
dfn-cert: DFN-CERT-2015-0081
dfn-cert: DFN-CERT-2015-0076
dfn-cert: DFN-CERT-2014-1717
dfn-cert: DFN-CERT-2014-1680
dfn-cert: DFN-CERT-2014-1632
dfn-cert: DFN-CERT-2014-1564
dfn-cert: DFN-CERT-2014-1542
dfn-cert: DFN-CERT-2014-1414
dfn-cert: DFN-CERT-2014-1366
dfn-cert: DFN-CERT-2014-1354
```

[return to 192.168.1.1]
