horizontal line

**Vulnerability Assessment Report**

**For Cat's Company**

****

**By : Sukhvir Kaur Sahota**

Table of Contents

[**Executive summary**](#_z4gmboy809l5) **2**

[**Introduction**](#_lu0gdmt79zxk) **3**

[**Methodology**](#_ndhhoay79wiz) **6**

[**Findings**](#_dc91epk3fb5j) **7**

[**Risk Assessment**](#_kp8lxgd3p187) **9**

[High Severity Vulnerability](#_rdjezcz79kei) 10

[1. Unprotected OSSEC/Wazuh ossec-authd (authd Protocol)-7.5 (high)](#_7emhnc7ibdtw) 10

[2. HTTP Brute Force Logins With Default Credentials Reporting](#_ccc4kqku65dg) 12

[Medium Severity Vulnerability](#_gp41pvxinzah) 13

[1. SSL/TLS: Renegotiation DoS Vulnerability](#_j08v4hcs9izb) 14

[2. DCE/RPC and MSRPC Services Enumeration Reporting](#_i0o1z1cewwrv) 15

[3. SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection](#_9lb9hcpxwu40) 17

[Low Severity Vulnerability](#_w7wgf6qcy3q0) 19

[1. TCP Timestamps Information Disclosure](#_jaktvs5inbee) 19

[**Recommendations**](#_tg7g3zoh18k6) **21**

[Unprotected OSSEC/Wazuh ossec-authd (authd Protocol)](#_ia6vu9onmhio) 21

[HTTP Brute Force Logins With Default Credentials Reporting](#_7ezz013zsg74) 23

[SSL/TLS: Renegotiation DoS Vulnerability](#_jwsktjd3172) 24

[DCE/RPC and MSRPC Services Enumeration Reporting](#_f0bjonccymt2) 25

[SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection](#_cbt4q43qf1pf) 26

[TCP Timestamps Information Disclosure](#_1gohvcqeml1p) 27

[**Conclusion**](#_ljstsuuklkq) **29**

[**Appendix - Steps to perform a scan**](#_6ijjaft56o7n) **30**

[**Appendix -Linux**](#_wkapie51lau8) **34**

[**Appendix -WinServer**](#_euuvbcieuyrb) **36**

[**Appendix -Windows Workstation**](#_hbycoobkxhnm) **38**

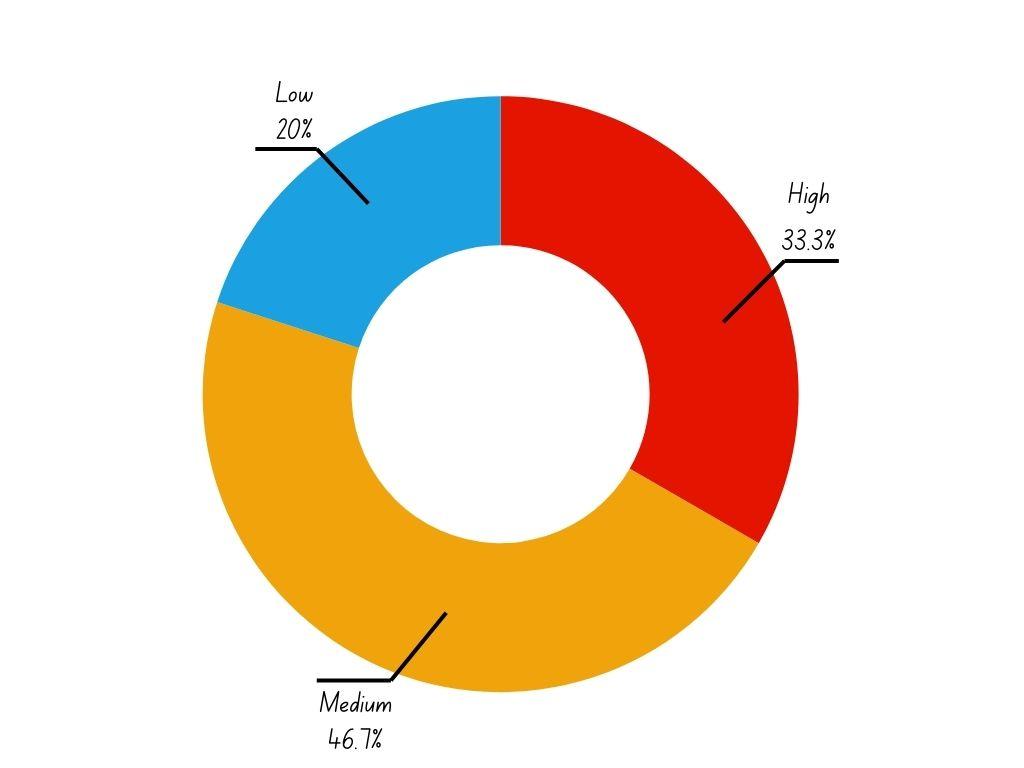
[**References**](#_690489ig5hkg) **40**

# 

# Executive summary

Vulnerability scanning is the process of discovering, analyzing, and reporting on security flaws and vulnerabilities. Vulnerability scanning and assessment is an essential step in the [vulnerability management lifecycle](https://www.beyondtrust.com/resources/glossary/vulnerability-assessment).

The Purpose of this scan for Cat’s Organisation is to find the threats associated with Linux, Winserver, and Windows workstation. This report contains the output of the vulnerability assessment conducted between Thursday, May 2nd, 2024, and Friday, May, 3rd, 2024.

From the scan, a total of 15 vulnerabilities were found out of which 5 high-severity vulnerabilities, 7 medium-severity vulnerabilities, and 3 low-severity vulnerabilities were found.

These vulnerabilities consist of outdated or end-of-life engines, HTTP brute force, SSL/TLS-related vulnerabilities, and ICMP/TCP timestamp disclosure. DCE/RPC and MSRPC service enumeration and Deprecated TLS V1.0 and 1.1 protocol detection.

In our recommendation, we suggested the immediate actions to be taken to resolve the top 6 vulnerabilities by applying authentication methods, patches for old or outdated TLS versions, firewall implementations, and an Intrusion detection and prevention system (IDPS). The suggestions are based on NIST SP 800-53 controls.

We have used the [Greenbone Vulnerability Management (GVM)](https://www.greenbone.net/wp-content/uploads/solution_comparison_EN.pdf) tool to conduct all scans, the [CVE MITRE](https://cve.mitre.org/) tool, and the [NVD NIST](https://nvd.nist.gov/) tool to research more details about the vulnerabilities. All-access contr suggestions were taken from NIST RMF SP 800 -53.

By understanding the identified vulnerabilities, their potential impact, and the recommended mitigation strategies, the Cat’s executive team can make informed decisions and allocate necessary resources to strengthen the organization's security posture.

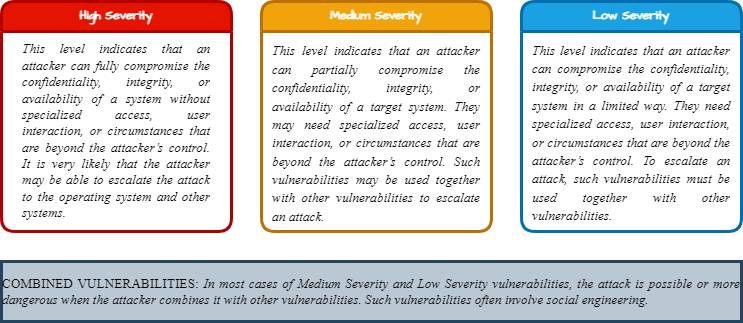
# Introduction

A vulnerability is a flaw in an application or device that malicious hackers can exploit. Attackers can exploit a vulnerability to achieve a goal such as stealing sensitive information, compromising the system by making it unavailable (in a denial-of-service scenario), or corrupting the data. The impact of vulnerabilities varies depending on the exploit.

In this report, we will be discussing the vulnerability assessment scan results for Cat’s organization. The assessment identified a range of vulnerabilities with varying severity levels. The scan is focused on the Linux server, Windows Server, and Windows workstation. The purpose of this is to save businesses from damages as Unmitigated vulnerabilities can have significant consequences for the organization, including data breaches and unauthorized access, system disruptions and downtime, financial losses, and reputational damage.

**Scan Results**

In our Report, the Cat’s company will assign severity based on the impact that the exploit may have on the system. It is based on the severity levels assigned to the identified vulnerabilities. This means that vulnerabilities with higher severity levels, such as High Severity, will be prioritized and recommended for immediate remediation. The recommendations may also take into account the business criticality of systems, ensuring that essential systems with higher criticality receive prompt attention.



The assignment of recommendations for addressing vulnerabilities in a company involves a coordinated effort across multiple teams, with a focus on prioritizing remediation efforts based on risk, impact, and business criticality. Collaboration, communication, and accountability are key to effectively managing vulnerabilities and mitigating security risks.

**The full list of all vulnerability results for all 3 systems**

The table below shows the

1. Systems on which Vulnerability assessment is done and their OS details.
2. Name of the corresponding vulnerability.
3. [QoD](https://docs.greenbone.net/GCS-Manual/gcs/en/reports.html) is short for “Quality of Detection” and shows the reliability of the detection of a vulnerability. By default, only results that were detected by a VT with a QoD of 70 % or higher are displayed. The possibility of false positives is thereby lower.
4. IP address host for which the result was found.
5. The port number and protocol type are used to find the result on the host.
6. The severity of the corresponding vulnerability. It is displayed with the color according to the severity level to support the analysis of the results.

| Linux-  [Ubuntu 20.04.6 LTS (Focal Fossa)](https://fridge.ubuntu.com/2023/03/23/ubuntu-20-04-6-lts-released/#:~:text=The%20Ubuntu%20team%20is%20pleased,on%20Secure%20Boot%20enabled%20systems.) | **Vulnerability** | **QoD** | **Host Ip** | **Location/ Service**  **(port)** | **Threat Level** |
| --- | --- | --- | --- | --- | --- |
| Report outdated / end-of-life Scan Engine / Environment (local) | 97% | 172.16.14.52 | general/  tcp | High (10.0) |
| Unprotected OSSEC/Wazuh ossec-authd (authd Protocol) | 80% | 172.16.14.52 | 1515/tcp | High (7.5) |
| HTTP Brute Force Logins With Default Credentials Reporting | 95% | 172.16.14.52 | 9200/tcp | High (7.5) |
| SSL/TLS: Die-Hellman Key Exchange Insouciant DH Group Strength Vulnerability | 99% | 172.16.14.52 | 1515/tcp | Medium (5.0) |
| SSL/TLS: Certificate Expired | 70% | 172.16.14.52 | 1515/tcp | Medium (5.0) |
| SSL/TLS: Renegotiation DoS Vulnerability | 80% | 172.16.14.52 | 9300/tcp | Medium (4.0) |
| Die-Hellman Key Exchange Insouciant DH Group Strength Vulnerability | 80% | 172.16.14.52 | 9200/tcp | Medium (4.0) |
| TCP Timestamps Information Disclosure | 80% | 172.16.14.52 | general/  tcp | Low (2.6) |
| ICMP Timestamp Reply Information Disclosure | 80% | 172.16.14.52 | general/  icmp | Low (2.1) |
|  | | | | | |
| Win Server-  [Windows Server 2022](https://learn.microsoft.com/en-us/windows-server/get-started/whats-new-in-windows-server-2022) | Report outdated / end-of-life Scan Engine / Environment (local) | 97% | 172.16.14.53 | general/tcp | High  (10.0) |
| DCE/RPC and MSRPC Services Enumeration Reporting | 80% | 172.16.14.53 | 135/tcp | Medium (5.0) |
| Deprecated TLSv1.0 and TLSv1.1 Protocol Detection | 98% | 172.16.14.53 | 3398/tcp | Medium (4.3) |
| TCP Timestamps Information Disclosure | 80% | 172.16.14.53 | general/  tcp | Low  (2.1) |
|  | | | | | |
| Windows Workstation-  [Windows 10 Pro](https://en.wikipedia.org/wiki/Windows_10,_version_1809) | Report outdated / end-of-life Scan Engine / Environment (local) | 97% | 172.16.14.50 | general/  tcp | High (10.0) |
| SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection | 98% | 172.16.14.50 | 3389/tcp | Medium (4.3) |

**This report will further include the following information**

* ***Vulnerability Summary:*** An overview of the total number of vulnerabilities detected, categorized by severity levels (High, Medium, Low), and possibly by business criticality.
* ***Detailed Vulnerability Reports:*** A report for each identified vulnerability, containing detailed information such as severity level, description, affected systems, potential impact, CVE references, and recommended remediation steps.
* ***Risk Assessment:*** Assessment of the overall risk posture of the organization's IT infrastructure, considering both the severity of vulnerabilities and the criticality of affected systems.
* ***Remediation Recommendations***: Specific recommendations for mitigating identified vulnerabilities, including patching systems, applying configuration changes, implementing security controls, or conducting further security testing.

# Methodology

1. **Greenbone vulnerability management**

For scanning, we have used the Greenbone vulnerability management (GVM) tool. The [Greenbone Vulnerability Management (GVM)](https://www.greenbone.net/wp-content/uploads/solution_comparison_EN.pdf) is a framework originally built as a community project named “OpenVAS” and is primarily developed and forwarded by Greenbone.

With the help of this tool, we ran a scan in our Kali System of Remote desktops by targeting the following environments. For detailed results from the OpenVAS vulnerability scan with details on how the scan was completed see [Appendix -A](#_6ijjaft56o7n)

| Assets | Scan IP | Type of Scan |
| --- | --- | --- |
| Windows1 | 172.16.14.50 | Full and Fast Vulnerability Scan |
| WinServer | 172.16.14.53 | Full and Fast Vulnerability Scan |
| Linux systems | 172.16.1452 | Full and Fast Vulnerability Scan |

**2. CVE Mitre and NVD NIST**

For this analysis, I have used the [CVE MITRE](https://cve.mitre.org/) tool and [NVD NIST](https://nvd.nist.gov) tool to research more details about the vulnerabilities.

**3. NIST Risk Management Framework Controls - SP 800 -53**

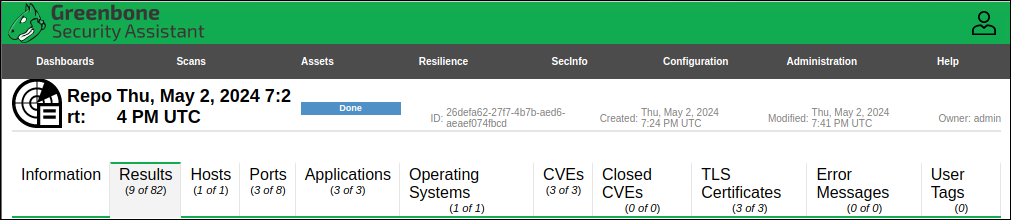
The NIST Risk Management Framework (RMF) provides a structured approach for managing cybersecurity risk within organizations. It consists of six steps: Prepare, Categorize, Select, Implement, Assess, and Authorize. Throughout these steps, various controls from NIST Special Publication 800-53 are applied to manage and mitigate risks effectively. There are a total of 20 [control families](https://csrc.nist.rip/Projects/risk-management/sp800-53-controls/release-search#!/families?version=5.1) and we will use the recommendations based on those controls.

# 

# Findings

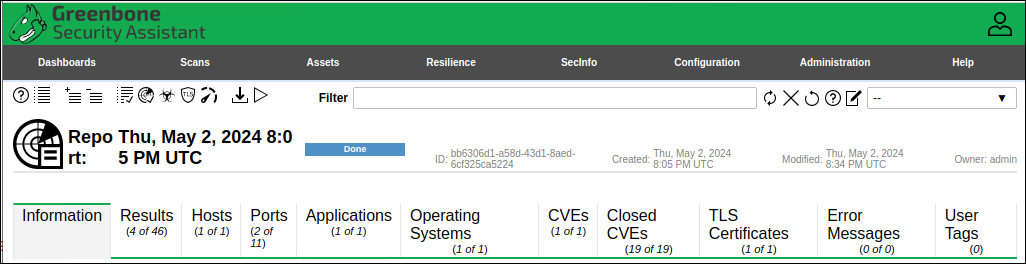
The results from all 3 systems are listed below. It is important to note that the results of all ports are not shown. Only 3 out of 8 ports were scanned for Linux system, similarly, 1 out of 2 open ports were scanned for Windows 1 workstation, and only 2 out of 11 ports were scanned for WinServer.

**Linux**

****

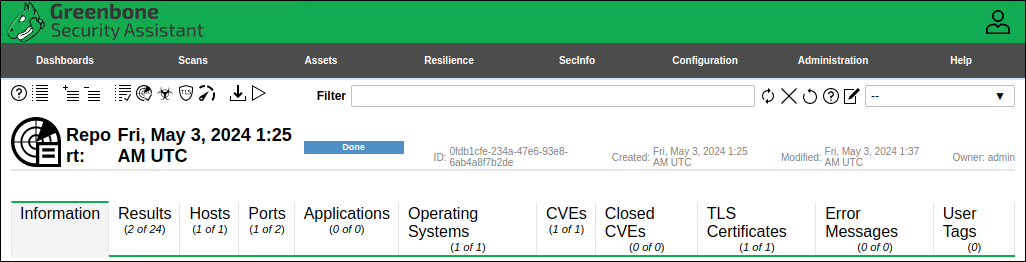
* 3 out of 8 ports were scanned for the Linux system.
* 3 applications of CPE were not able to show any severity.
* Issues with the threat level of “Log”, “Debug”, and “False Positive” are not shown. Only results with a minimum QoD of 70 are shown.
* contains all 9 results selected by the filtering described above. Before filtering there were 82 results.
* The scan started at Thu May 2 19:25:07 2024 UTC and ended at Thu May 2 19:41:42 2024 UTC.
* See detailed results in [Appendix -Linux](#_wkapie51lau8).
* Linux [Raw scan report from the GVM](https://drive.google.com/file/d/1fTP3LGKJU-vEraI5rgIY9LJm6oLpLci9/view?usp=sharing)

**WinServer**

****

* 2 out of 11 ports were scanned for WinServer.
* 1application of CPE was not able to show any severity.
* Issues with the threat level of “Log”, “Debug”, and “False Positive” are not shown. Only results with a minimum QoD of 70 are shown.
* Only 4 results were selected by the filtering described above. Before filtering there were 46 results.
* The task was WinServer. The scan started at Thu May 2 20:06:29 2024 UTC and ended at Thu May 2 20:34:28 2024 UTC.
* See detailed results in [Appendix -WinServer](#_euuvbcieuyrb).
* WinServer [Raw scan report from the GVM](https://drive.google.com/file/d/1UReNh477bvxOZrHOUVf5lL7hFsQ7ys1B/view?usp=sharing)

**Windows1**

****

* 1 out of 2 open ports were scanned for Windows 1 workstation.
* Issues with the threat level of “Log”, “Debug”, and “False Positive” are not shown. Only results with a minimum QoD of 70 are shown.
* Only 2 results were selected by the filtering described above. Before filtering there were 24 results.
* The scan started on Fri May 3 01:26:03 2024 UTC and ended on Fri May 3 01:37:41 2024 UTC.
* See detailed results in [Appendix -Windows workstation](#_hbycoobkxhnm).
* Windows1 [Raw scan report from the GVM](https://drive.google.com/file/d/1djN3xkc5YaNxfQtuIR9z-7bEzRD9HWS-/view?usp=sharing)

**Type of Scan results**

A full listing of the scanned host is available in the “ [Scan results](#_3zqr6ehstxb4)” section of the report :

1. ***Outdated Scan Engin*e:** Reported outdated/end-of-life scan Engine /Environment(local) - All 3 systems reported the outdated Greenbone enterprise trail.
2. ***SSL/TLS*:** There are reports of vulnerabilities regarding SSL/TLS such as Certification expired and renegotiation DoS vulnerability.
3. ***TCP/ICMP Timestamps***: These consist of the details of remote host timestamps and timestamp replies on ICMP messages.

# 

# Risk Assessment

This report identifies security risks that could have a significant impact on mission-critical applications used for day-to-day business operations. We have performed a scan on three major servers used by Cat’s organization and the following are the results of the number of vulnerabilities we found on each machine during our scan.

| **System** | **High Severity** | **Medium Severity** | **Low Severity** |
| --- | --- | --- | --- |
| Linux | 3 | 4 | 2 |
| WinServer | 1 | 2 | 1 |
| Windows1 | 1 | 1 | 0 |

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

**Top six vulnerabilities**

1. Unprotected OSSEC/Wazuh ossec-authd (authd Protocol)-7.5 (high)
2. HTTP Brute Force Logins With Default Credentials Reporting-7.5 (high)
3. SSL/TLS: Renegotiation DoS Vulnerability -5.0 (Medium)
4. DCE/RPC and MSRPC Services Enumeration Reporting - 5.0 (Medium)
5. SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection -4.3 (Medium)
6. TCP Timestamps Information Disclosure- 2.6 (Low)

## High Severity Vulnerability

5 of the Vulnerabilities are of high severity, these required immediate attention. These are relatively easy for attackers to exploit and can lead to the complete loss of control of the affected system.

The top 2 high vulnerabilities are described below

### Unprotected OSSEC/Wazuh ossec-authd (authd Protocol)-7.5 (high)

| **High (CVSS: 7.5)**  **Unprotected OSSEC/Wazuh ossec-authd (authd Protocol)** | |
| --- | --- |
| ***Date of its discovery*** | Thu, May 2, 2024 7:28 PM UTC |
| ***A detailed description of the vulnerability*** | ***Summary:*** The remote OSSEC/Wazuh ossec-authd service is not protected by password authentication or client certificate verification.  ***Detection Result*:** The vulnerability was detected according to the Detection Method.  ***Insight:*** It was possible to connect to the remote OSSEC/Wazuh ossec-authd service without providing a password or a valid client certificate.  ***Detection Method:*** Evaluate if the remote OSSEC/Wazuh ossec-authd service is protected by password authentication or client certificate verification.  ***Details:*** Unprotected OSSEC/Wazuh ossec-auth (authd Protocol) OID: 1.3.6.1.4.1.25623.1.0.108547  ***Impact*:** This issue may be misused by a remote attacker to register  arbitrary agents at the remote service or overwrite the registration of existing ones taking them out of service.  This vulnerability is related to command injection, so the impact is high. |
| ***Detailed description of the affected system*** | Ubuntu 20.04.6 LTS (Focal Fossa) is a specific version of the popular Linux operating system released by Canonical. Here's a breakdown:  **Key Points:**   * Version: 20.04.6 * Codename: Focal Fossa * Release Type: Long Term Support (LTS) * Release Date: April 2020 (version 20.04), with point release 6 being the latest update. * LTS (Long Term Support) releases: Supported for 5 years of standard support and an additional 5 years of Extended Security Maintenance (ESM)   **Features:** Ubuntu 20.04.6 LTS includes various features and improvements over previous versions, including:   * Updated Linux kernel (version 5.11 in 20.04.6) * Enhanced hardware support * Security updates and bug fixes * New software versions (e.g., GNOME desktop environment) |
| ***Details of the process to correct the vulnerability*** | Enable password authentication or client certificate verification within the configuration of ossec-authd. Please see the manual of this service for more information.  [Detailed steps for mitigation and recommended control](#_ia6vu9onmhio) can be found in the recommendation section. |
| ***References CVE from analysis*** | [**CVE-2023-50260**](https://nvd.nist.gov/vuln/detail/cve-2023-50260)  A wrong validation in the `host\_deny` script allows to write any string in the `hosts. deny` file, which can end in an arbitrary command execution on the target system. This vulnerability is part of the active response feature, which can automatically trigger actions in response to alerts. By default, active responses are limited to a set of pre-defined executables. This is enforced by only allowing executables stored under `/var/ossec/active-response/bin` to be run as an active response. However, the `/var/ossec/active-response/bin/host\_deny` can be exploited. `host\_deny` is used to add IP address to the `/etc/hosts.deny` file to block incoming connections on a service level by using TCP wrappers. An attacker can inject an arbitrary command into the `/etc/hosts.deny` file and execute arbitrary commands by using the spawn directive. The active response can be triggered by writing events either to the local `execd` queue on the server or to the `ar` queue which forwards the events to agents. So, it can lead to LPE on the server as root and RCE on the agent as root. This vulnerability is fixed in 4.7.2. |

### HTTP Brute Force Logins With Default Credentials Reporting

| **High (CVSS: 7.5)**  **HTTP Brute Force Logins With Default Credentials Reporting** | |
| --- | --- |
| ***Date of its discovery*** | Thu, May 2, 2024 7:38 PM UTC |
| ***A detailed description of the vulnerability*** | ***Summary***: It was possible to login into the remote Web Application using default credentials.  ***Detection Result***: It was possible to log in with the following credentials (<URL>:<User>:<Password>:<HTTP status code>) https://172.16.14.52:9200/:admin:admin:HTTP/1.1 200 OK  ***Insight***: As the VT 'HTTP Brute Force Logins With Default Credentials' (OID: 1.3.6.1.4.1.25623.1.0.108041) might run into a timeout the actual reporting of this vulnerability takes place in this VT instead.  ***Detection Method***: Reports default credentials detected by the VT. ‘'HTTP Brute Force Logins With Default Credentials' (OID: 1.3.6.1.4.1.25623.1.0.108041)  ***Impact***: This issue may be exploited by a remote attacker to e.g. gain access to sensitive information or modify system configuration. |
| ***Detailed description of the affected system*** | Ubuntu 20.04.6 LTS (Focal Fossa) is a specific version of the popular Linux operating system released by Canonical. Here's a breakdown:  ***Key Points:***   * Version: 20.04.6 * Codename: Focal Fossa * Release Type: Long Term Support (LTS) * Release Date: April 2020 (version 20.04), with point release 6 being the latest update. * LTS (Long Term Support) releases: Supported for 5 years of standard support and an additional 5 years of Extended Security Maintenance (ESM)   ***Features:*** Ubuntu 20.04.6 LTS includes various features and improvements over previous versions, including:   * Updated Linux kernel (version 5.11 in 20.04.6) * Enhanced hardware support * Security updates and bug fixes * New software versions (e.g., GNOME desktop environment) |
| ***Details of the process to correct the vulnerability*** | Change the password as soon as possible. Implement Account Lockout Policies.  [Detailed steps for mitigation and recommended control](#_7ezz013zsg74) can be found in the recommendation section. |
| ***References CVE from GVM*** | [**CVE-1999-0501**](https://nvd.nist.gov/vuln/detail/CVE-1999-0501)  A Unix account has a guessable password.  [**CVE-1999-0502**](https://nvd.nist.gov/vuln/detail/CVE-1999-0502)  A Unix account has a default, null, blank, or missing password.  [**CVE-1999-0507**](https://nvd.nist.gov/vuln/detail/CVE-1999-0507)  An account on a router, firewall, or other network device has a guessable password.  [**CVE-1999-0508**](https://nvd.nist.gov/vuln/detail/CVE-1999-0508)  An account on a router, firewall, or other network device has a default, null, blank, or missing password. |
| ***References CVE from analysis*** | [**CVE-2023-33868**](https://nvd.nist.gov/vuln/detail/cve-2023-33868)  The number of login attempts is not limited. This could allow an attacker to perform a brute force on HTTP basic authentication. |

## Medium Severity Vulnerability

7 of the Vulnerabilities are of medium severity, these required attention. These should also be fixed promptly but are not as urgent as the other vulnerabilities.

The top 3 medium vulnerabilities are described below

### 

### SSL/TLS: Renegotiation DoS Vulnerability

| **Medium (CVSS : 5.0)**  **SSL/TLS: Renegotiation DoS Vulnerability** | |
| --- | --- |
| ***Date of its discovery*** | Thu, May 2, 2024 7:37 PM UTC |
| ***A detailed description of the vulnerability*** | ***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.  ***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.  ***Impact***: The flaw might make it easier for remote attackers to cause a DoS (CPU consumption) by performing many renegotiations within a single connection. |
| ***A detailed description of the affected system*** | Ubuntu 20.04.6 LTS (Focal Fossa) is a specific version of the popular Linux operating system released by Canonical. Here's a breakdown:  ***Key Points:***   * Version: 20.04.6 * Codename: Focal Fossa * Release Type: Long Term Support (LTS) * Release Date: April 2020 (version 20.04), with point release 6 being the latest update. * LTS (Long Term Support) releases: Supported for 5 years of standard support and an additional 5 years of Extended Security Maintenance (ESM)   ***Features:***  Ubuntu 20.04.6 LTS includes various features and improvements over previous versions, including:   * Updated Linux kernel (version 5.11 in 20.04.6) * Enhanced hardware support * Security updates and bug fixes * New software versions (e.g., GNOME desktop environment) |
| ***Details of the process to correct the vulnerability*** | 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.  [Detailed steps for mitigation and recommended control](#_jwsktjd3172) can be found in the recommendation section. |
| ***References CVE from GVM*** | [**CVE-2011-1473**](https://nvd.nist.gov/vuln/detail/CVE-2011-1473)  OpenSSL before 0.9.8l, and 0.9.8m through 1. x, does not properly restrict client-initiated renegotiation within the SSL and TLS protocols, which might make it easier for remote attackers to cause a denial of service (CPU consumption) by performing many renegotiations within a single connection, a different vulnerability than CVE-2011-5094. NOTE: 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. |
| ***References CVE from analysis*** | [**CVE-2021-3449**](https://nvd.nist.gov/vuln/detail/cve-2021-3449)  An OpenSSL TLS server may crash if a maliciously crafted renegotiation ClientHello message from a client. If a TLSv1.2 renegotiation ClientHello omits the signature\_algorithms extension (where it was present in the initial ClientHello), but includes a signature\_algorithms\_cert extension then a NULL pointer dereference will result, leading to a crash and a denial of service attack. A server is only vulnerable if it has TLSv1.2 and renegotiation enabled (which is the default configuration). OpenSSL TLS clients are not impacted by this issue. |

### DCE/RPC and MSRPC Services Enumeration Reporting

| **Medium (CVSS: 5.0)**  **DCE/RPC and MSRPC Services Enumeration Reporting** | |
| --- | --- |
| ***Date of its discovery*** | Thu, May 2, 2024, 8:23 PM UTC |
| ***A detailed description of the vulnerability*** | ***Summary***: Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.  ***Detection Method***: DCE/RPC and MSRPC Services Enumeration Reporting OID: 1.3.6.1.4.1.25623.1.0.10736.  ***Impact***: An attacker may use this fact to gain more knowledge  about the remote host. |
| ***A detailed description of the affected system*** | Windows Server 2022 v 21H2 is a major release of the Windows Server operating system designed for enterprise-level workloads.  It's an LTSC version, meaning it receives security updates and bug fixes but not major feature updates throughout its support lifecycle  Here's a detailed overview:  ***Key Points:***   * Version: Windows Server 2022, version 21H2 * Release Date: August 18, 2021 * Channel: Long-Term Servicing Channel (LTSC) * Support: Mainstream Support: Until October 13, 2026 * Extended Support: Until October 14, 2031   ***Features****:*   * *Azure Hybrid Capabilities*: Offers seamless integration with Azure services for hybrid cloud deployments. * *Replacement of Internet Explorer:* Microsoft Edge replaces Internet Explorer as the default browser, providing enhanced security and features. * *Security Enhancements*: Includes features like TPM 2.0, UEFI Secure Boot, and Boot DMA Protection for improved security posture. * *Networking Improvements*: Supports DNS-over-HTTPS for secure DNS resolution and Server Message Block (SMB) compression for efficient data transfer. |
| ***Details of the process to correct the vulnerability*** | Incoming traffic to these ports, Firewall Configuration, Service Hardening, Intrusion Detection and Prevention Systems (IDPS).  [Detailed steps for mitigation and recommended control](#_f0bjonccymt2) can be found in the recommendation section. |
| ***References CVE from analysis*** | [**CVE-2023-42669**](https://nvd.nist.gov/vuln/detail/CVE-2023-42669)  A vulnerability was found in Samba's "rpcecho" development server, a non-Windows RPC server used to test Samba's DCE/RPC stack elements. This vulnerability stems from an RPC function that can be blocked indefinitely. The issue arises because the "rpcecho" service operates with only one worker in the main RPC task, allowing calls to the "rpcecho" server to be blocked for a specified time, causing service disruptions. This disruption is triggered by a "sleep()" call in the "dcesrv\_echo\_TestSleep()" function under specific conditions. Authenticated users or attackers can exploit this vulnerability to make calls to the "rpcecho" server, requesting it to block for a specified duration, effectively disrupting most services and leading to a complete denial of service on the AD DC. The DoS affects all other services as "rpcecho" runs in the main RPC task.  [**CVE-2018-8407**](https://nvd.nist.gov/vuln/detail/CVE-2018-8407)  An information disclosure vulnerability exists when the "Kernel Remote Procedure Call Provider" driver improperly initializes objects in memory, aka "MSRPC Information Disclosure Vulnerability." This affects Windows 7, Windows Server 2012 R2, Windows RT 8.1, Windows Server 2008, Windows Server 2019, Windows Server 2012, Windows 8.1, Windows Server 2016, Windows Server 2008 R2, Windows 10, Windows 10 Servers. |

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

| **Medium (CVSS: 4.3)**  **SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection** | |
| --- | --- |
| ***Date of its discovery*** | Fri, May 3, 2024 1:31 AM UTC |
| ***A Detailed description of the vulnerability*** | ***Summary***: It was possible to detect the usage of the deprecated TLSv1.0 and/or TLSv1.1 protocol on this system.  ***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 can be found in the 'SSL/TLS: Report Supported Cipher Suites' (OID: 1.3.6.1.4.1.25623.1.0.802067) VT.  ***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  ***Impact***: An attacker might be able to use the known cryptographic flaws to eavesdrop on the connection between clients and the service to get access to sensitive data transferred within the secured connection. |
| ***A detailed description of the affected system*** | Windows 10 Pro v 1809, also known as the October 2018 Update or "Redstone 5," was a major update released in October 2018.  ***Features***   * High-Efficiency Image File Format (HEIF) support * New safe remove experience for external GPUs * New privacy settings layout during setup * Improvements to Windows Defender Application Guard |
| ***Details of the process to correct the vulnerability*** | It is recommended to disable the deprecated TLSv1.0 and/or TLSv1.1 protocols in favor of the TLSv1.2+ protocols.  [Detailed steps for mitigation and recommended control](#_cbt4q43qf1pf) can be found in the recommendation section. |
| ***References CVE from GVM*** | [**CVE-2011-3389**](https://nvd.nist.gov/vuln/detail/CVE-2011-3389)  The SSL protocol, as used in certain configurations in Microsoft Windows and Microsoft Internet Explorer, Mozilla Firefox, Google Chrome, Opera, and other products, encrypts data by using CBC mode with chained initialization vectors, which allows man-in-the-middle attackers to obtain plaintext HTTP headers via a blockwise chosen-boundary attack (BCBA) on an HTTPS session, in conjunction with JavaScript code that uses (1) the HTML5 WebSocket API, (2) the Java URLConnection API, or (3) the Silverlight WebClient API, aka a "BEAST" attack.  [**CVE-2015-0204**](https://nvd.nist.gov/vuln/detail/CVE-2015-0204)  The ssl3\_get\_key\_exchange function in s3\_clnt.c in OpenSSL before 0.9.8zd, 1.0.0 before 1.0.0p, and 1.0.1 before 1.0.1k allows remote SSL servers to conduct RSA-to-EXPORT\_RSA downgrade attacks and facilitate brute-force decryption by offering a weak ephemeral RSA key in a non-compliant role, related to the "FREAK" issue. NOTE: the scope of this CVE is only client code based on OpenSSL, not EXPORT\_RSA issues associated with servers or other TLS implementations. |
| ***References CVE from analysis*** | [**CVE-2022-36937**](https://nvd.nist.gov/vuln/detail/CVE-2022-36937)  HHVM 4.172.0 and all prior versions use TLS 1.0 for secure connections when handling tls:// URLs in the stream extension. TLS1.0 has numerous published vulnerabilities and is deprecated. HHVM 4.153.4, 4.168.2, 4.169.2, 4.170.2, 4.171.1, 4.172.1, 4.173.0 replaces TLS1.0 with TLS1.3. Applications that call stream\_socket\_server or stream\_socket\_client functions with a URL starting with tls:// are affected. |

## Low Severity Vulnerability

3 of the Vulnerabilities are of low severity. These are not required to be fixed promptly but should be in consideration.

The top 1 low vulnerability is described below:

### TCP Timestamps Information Disclosure

| **Low (CVSS: 2.6 )**  **TCP Timestamps Information Disclosure** | |
| --- | --- |
| ***Date of its discovery*** | Thu, May 2, 2024 8:21 PM UTC |
| ***A detailed description of the vulnerability*** | ***Summary***: The remote host implements TCP timestamps and therefore allows to compute the uptime.  ***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: 1341799  Packet 2: 1342863  ***Insight***: The remote host implements TCP timestamps, as defined by  RFC1323/RFC7323.  ***Detection Method***: Special IP packets are forged and sent with a little delay in between to the target IP. The responses are searched for timestamps. If found, the timestamps are reported.  ***Impact***: A side effect of this feature is that the uptime of the remote  host can sometimes be computed. |
| ***A Detailed description of the affected system*** | Windows Server 2022 v 21H2 is a major release of the Windows Server operating system designed for enterprise-level workloads.  It's an LTSC version, meaning it receives security updates and bug fixes but not major feature updates throughout its support lifecycle  Here's a detailed overview:  ***Key Points:***   * Version: Windows Server 2022, version 21H2 * Release Date: August 18, 2021 * Channel: Long-Term Servicing Channel (LTSC) * Support: Mainstream Support: Until October 13, 2026 * Extended Support: Until October 14, 2031   ***Features***:   * *Azure Hybrid Capabilities*: Offers seamless integration with Azure services for hybrid cloud deployments. * *Replacement of Internet Explorer:* Microsoft Edge replaces Internet Explorer as the default browser, providing enhanced security and features. * *Security Enhancements*: Includes features like TPM 2.0, UEFI Secure Boot, and Boot DMA Protection for improved security posture. * *Networking Improvements*: Supports DNS-over-HTTPS for secure DNS resolution and Server Message Block (SMB) compression for efficient data transfer.   TCP implementations that implement RFC1323/RFC7323. |
| ***Details of the process to correct the vulnerability*** | Disable timestamp. [Detailed steps for mitigation and recommended control](#_1gohvcqeml1p) can be found in the recommendation section. |
| ***References CVE from GVM*** | [**CVE-1999-0524**](https://nvd.nist.gov/vuln/detail/CVE-1999-0524)  ICMP information such as (1) netmask and (2) timestamp is allowed from arbitrary hosts. |
| ***References CVE from analysis*** | [**CVE-2019-18625**](https://nvd.nist.gov/vuln/detail/CVE-2019-18625)  An issue was discovered in Suricata 5.0.0. It was possible to bypass/evade any TCP-based signature by faking a closed TCP session using an evil server. After the TCP SYN packet, it is possible to inject an RST ACK and a FIN ACK packet with a bad TCP Timestamp option. The client will ignore the RST ACK and the FIN ACK packets because of the bad TCP Timestamp option. Both Linux and Windows clients are ignoring the injected packets.  [**CVE-2006-6893**](https://nvd.nist.gov/vuln/detail/CVE-2006-6893)  Tor allows remote attackers to discover the IP address of a hidden service by accessing this service at a high rate, thereby changing the server's CPU temperature and consequently changing the pattern of time values visible through (1) ICMP timestamps, (2) TCP sequence numbers, and (3) TCP timestamps, a different vulnerability than CVE-2006-0414. NOTE: it could be argued that this is a laws-of-physics vulnerability that is a fundamental design limitation of certain hardware implementations, so perhaps this issue should not be included in CVE. |

# 

# Recommendations

For Cat’sorganization, the Linux server is used for critical infrastructure services such as web hosting, database management, file serving, and networking service. This server may host sensitive data and critical applications, making them essential assets to protect.s.

Winserverare is commonly used for various enterprise services such as Active Directory, Microsoft Exchange, SharePoint, SQL Server, and application servers. They often play a central role in managing user authentication, file sharing, email communication, and business-critical applications.

Windows workstation is used by individual employees for day-to-day tasks. While they may not host critical services like servers, compromised workstations can serve as entry points for attackers to access sensitive information spread malware, or perform other malicious activities within the network. So, Protecting Windows workstations is crucial for maintaining the overall security posture of an organization

Based on the above it is best to tackle the ones with the highest severity first. In this case, "high" severity vulnerabilities should be addressed before "medium" or "low" severity ones. Below are the mitigation steps for each mentioned vulnerability from high to low.

## 

## 

## Unprotected OSSEC/Wazuh ossec-authd (authd Protocol)

**Mitigation steps :**

1. **Enable Password Authentication:**The following steps serve as a guide on how to enroll a Linux/Unix endpoint with password authentication :
   * Launch the terminal as a root user.
   * Create the file /var/ossec/etc/authd.pass with the enrollment password in it.
   * Add the Wazuh manager IP address or DNS name in the <client><server><address> section of the agent configuration file /var/ossec/etc/ossec.conf.
2. **Enable Client Certificate Verification (Optional):**
   * If desired, configure client certificate verification for enhanced security.
   * This involves creating and distributing client certificates to authorized agents and configuring the ossec-authd service to verify their authenticity.
3. **Restart the ossec-authd Service:**
   * After making configuration changes, restart the ossec-authd service to apply the new settings. For more details see the [link](https://documentation.wazuh.com/current/user-manual/agent/agent-enrollment/security-options/using-password-authentication.html).

**SP 800-53 Controls:**

The implemented corrective measures align with several SP 800-53 security controls:

* [**AC-3(3): Mandatory Access Control**](https://csf.tools/reference/nist-sp-800-53/r5/ac/ac-3/ac-3-3/)**:** Enforcing password authentication or client certificate verification restricts access to the ossec-authd service, adhering to the principle of least privilege.
* [**IA-2(2): Multi-factor Authentication to Non-privileged Accounts**](https://csf.tools/reference/nist-sp-800-53/r5/ia/ia-2/ia-2-2/)**:** Utilizing passwords or certificates for authentication ensures that only authorized entities can interact with the service.

**Additional Considerations:**

* Regularly update OSSEC/Wazuh to the latest version to benefit from security patches and address newly discovered vulnerabilities.
* Implement intrusion detection and prevention systems (IDS/IPS) to monitor network activity and potentially detect unauthorized attempts to access the ossec-authd service.
* Adhere to security best practices like using strong passwords, following the principle of least privilege, and regularly reviewing system configurations.

By implementing these controls, you significantly strengthen the security posture of your OSSEC/Wazuh system and mitigate the risks associated with the identified vulnerabilities.

## HTTP Brute Force Logins With Default Credentials Reporting

**Mitigation steps :**

1. **Change Default Credentials:**
   * Immediately change the default credentials for the web application to strong and unique passwords that are difficult to guess.
   * Consider using a password manager to generate and manage complex passwords securely.
2. **Enforce Password Complexity Requirements:**
   * Implement password complexity requirements within the web application, such as minimum length, character diversity (uppercase, lowercase, numbers, symbols), and password history.
3. **Implement Account Lockout Policies:**
   * Configure account lockout policies to automatically lock accounts after a certain number of failed login attempts. This prevents brute-force attacks from succeeding.
4. **Disable Accounts When Not in Use:**
   * Disable user accounts that are no longer actively used to minimize the attack surface.
5. **Security Awareness Training:**
   * Provide security awareness training to users to educate them about the dangers of using weak passwords and the importance of good password hygiene.

**SP 800-53 Controls:**

The implemented corrective measures align with several SP 800-53 security controls:

* [**IA-5(1): Password-Based Authentication**](https://csf.tools/reference/nist-sp-800-53/r4/ia/ia-5/ia-5-1/): Authenticator Management Changing default credentials and enforcing password complexity requirements strengthen the authentication process, ensuring only authorized users can access the web application.
* [**AC-7: Unsuccessful Logon Attempts**](https://csf.tools/reference/nist-sp-800-53/r5/ac/ac-7/)**:** Implementing account lockout policies and disabling unused accounts adheres to the principle of least privilege and reduces the potential attack surface.

**Additional Considerations:**

* Regularly update the web application to the latest version to benefit from security patches and address newly discovered vulnerabilities.
* Consider implementing additional security measures like multi-factor authentication (MFA) for enhanced protection.
* Monitor web application logs for suspicious activity and investigate any potential unauthorized access attempts.

By following these steps and aligning with SP 800-53 controls, you can effectively address the vulnerability and significantly reduce the risk of unauthorized access to your web application.

## SSL/TLS: Renegotiation DoS Vulnerability

**Mitigation steps :**

1. **Disable Client-Initiated Renegotiations:**
   * If possible, disable client-initiated renegotiations within the SSL/TLS configuration. This significantly reduces the attack surface and eliminates the possibility of DoS attacks through this specific vulnerability.
2. **Upgrade to Secure Protocols:**
   * Consider upgrading to more recent TLS versions (e.g., TLS 1.3) that offer improved security features and potentially address the renegotiation vulnerability.
3. **Monitor for Suspicious Activity:**
   * Implement mechanisms to monitor network traffic and SSL/TLS connections for signs of excessive renegotiation attempts, potentially indicating a DoS attack.

**SP 800-53 Controls:**

The implemented corrective measures align with several SP 800-53 security controls:

* [**SC-8(1): Cryptographic Protection**](https://csf.tools/reference/nist-sp-800-53/r5/sc/sc-8/sc-8-1/)**:** Disabling client-initiated renegotiations and using secure protocols strengthens the security of the SSL/TLS service, mitigating the DoS vulnerability and protecting the system from potential service disruptions.
* [**SI-4: System Monitoring**](https://csf.tools/reference/nist-sp-800-53/r5/si/si-4/)**:** Regularly monitoring network traffic and SSL/TLS connections helps identify suspicious activity and potential DoS attacks, allowing for timely mitigation actions.

**Additional Considerations:**

* Keep the SSL/TLS libraries and protocols used by the service updated to the latest versions to benefit from security patches and address newly discovered vulnerabilities.
* Consider implementing additional DoS mitigation strategies like rate limiting or connection throttling to further protect against potential attacks.

By following these steps and aligning with SP 800-53 controls, you can effectively address the SSL/TLS renegotiation DoS vulnerability and enhance the overall security and stability of your SSL/TLS service.

## DCE/RPC and MSRPC Services Enumeration Reporting

**Mitigation steps :**

1. **Restrict Access to Port 135:**
   * Configure firewalls to block incoming connections to port 135 (TCP and UDP), which is the default port used for DCE/RPC and MSRPC services. This significantly reduces the attack surface and prevents unauthorized enumeration attempts.
2. **Disable Unnecessary Services:**
   * Identify and disable any unnecessary services running on the system. This minimizes the information exposed through enumeration and reduces the potential attack surface.
3. **Implement Network Segmentation:**
   * Consider implementing network segmentation to isolate sensitive systems and services, further restricting access and reducing the potential impact of successful enumeration.
4. **Keep Systems Updated:**
   * Regularly apply security updates and patches to the operating system and services to address known vulnerabilities that attackers might exploit after successful enumeration.

**SP 800-53 Controls:**

The implemented corrective measures align with several SP 800-53 security controls:

* [**CM-7: Least Functionality**](https://csf.tools/reference/nist-sp-800-53/r5/cm/cm-7/)**:** Disabling unnecessary services adheres to the principle of least privilege, reducing the attack surface and potential vulnerabilities.
* [**SC-7: Boundary Protection**](https://csf.tools/reference/nist-sp-800-53/r5/sc/sc-7/)**:** By restricting access to ports, organizations can mitigate the risk of exploitation by external adversaries and prevent unauthorized access to sensitive information or resources.

**Additional Considerations:**

* Monitor network traffic for suspicious activity, particularly attempts to access port 135 or other ports associated with specific services.
* Consider using intrusion detection and prevention systems (IDS/IPS) to detect and potentially block unauthorized enumeration attempts.
* Regularly review and update security configurations to ensure they remain effective against evolving threats.

By following these steps and aligning with SP 800-53 controls, you can significantly mitigate the risk associated with DCE/RPC and MSRPC services enumeration, reducing the potential attack surface and making it more difficult for attackers to exploit your systems.

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

**Mitigation steps :**

1. **Disable TLSv1.0 and TLSv1.1:**
   * Identify the systems or services using outdated TLS versions.
   * Within the security configurations of those systems or services, disable the use of TLSv1.0 and TLSv1.1 protocols.
2. **Upgrade to TLSv1.2 or Later:**
   * Implement TLSv1.2 or even later versions (e.g., TLS 1.3) that offer stronger cryptographic algorithms and address the vulnerabilities present in older protocols.
3. **Testing and Compatibility:**
   * After disabling older protocols and enabling newer ones, thorough testing is crucial to ensure compatibility with all intended users and applications.

**SP 800-53 Controls:**

The implemented corrective measures align with several SP 800-53 security controls:

* [**SC-8: Transmission Confidentiality and Integrity:**](https://csf.tools/reference/nist-sp-800-53/r5/sc/sc-8/)Disabling outdated protocols and using more secure versions strengthens the security of communication channels, mitigating the risks associated with known cryptographic weaknesses.
* [**CM-4(2): Verification of Controls**](https://csf.tools/reference/nist-sp-800-53/r5/cm/cm-4/cm-4-2/)**:** AfterImplementing secure TLS versions, make sure the system is working as expected.

**Additional Considerations:**

* Upgrading to the latest TLS versions is often a straightforward process through system or application settings.
* If compatibility issues arise with specific software or hardware after upgrading, you might need to check for updated drivers or compatibility information.
* Regularly update the systems and applications to benefit from the latest security patches and address newly discovered vulnerabilities.

By following these steps and aligning with SP 800-53 controls, you can effectively address the risks associated with using deprecated TLS versions and significantly enhance the security of your communication channels.

## TCP Timestamps Information Disclosure

**Mitigation steps :**

* **Disabling TCP Timestamps :**
  + If desired, some operating systems and network devices allow disabling TCP timestamps within their configuration settings. However, this might have compatibility implications with certain applications or protocols that rely on this feature. This vulnerability is found in both Linux and Windows, The following steps can be taken

*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 system 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.

1. **Network Segmentation and Access Control:**
   * Implement network segmentation and access control measures to restrict unauthorized access to systems and limit the information attackers can gather.
2. **Security Awareness Training:**
   * Educate users about the potential risks of information disclosure and the importance of maintaining good security practices.

**SP 800-53 Controls:**

The implemented corrective measures (disabling TCP timestamps and access control) align with several SP 800-53 security controls:

* [**SC-7(9): Restrict Threatening Outgoing Communications Traffic**](https://csf.tools/reference/nist-sp-800-53/r5/sc/sc-7/sc-7-9/)**:** Disabling TCP timestamps, if feasible, reduces the amount of information exposed through network communication, potentially mitigating the risk of attackers exploiting this information for malicious purposes.
* [**AC-4: Information Flow Enforcement**](https://csf.tools/reference/nist-sp-800-53/r5/ac/ac-4/)**:** Implementing network segmentation and access control creates boundaries that limit unauthorized access to systems and sensitive information, further reducing the potential impact of information disclosure.

**Additional Considerations:**

* The impact of this vulnerability is often limited, and the effort required to exploit it might be greater than the potential gains for attackers.
* Regularly review and update security configurations to ensure they remain effective against evolving threats.

While the immediate risk associated with TCP timestamps information disclosure is generally low, it's still recommended to consider the mitigation measures outlined above, especially if the system handles sensitive information or operates in a high-security environment.

# Conclusion

In conclusion, the identified vulnerabilities present varying levels of risk, with high-severity issues demanding immediate attention due to their potential to compromise system integrity and security. By aligning corrective actions with SP 800-53 security controls, cat’s organizations can effectively reduce attack surfaces and mitigate associated risks.

PPT Link - [Vulnerability Assessment Report For Cat's Company](https://docs.google.com/presentation/d/1pH8llwrkX63lmBLwacg7tt90IwSmoOPLDMO_qahHy9Y/edit?usp=sharing)

PPT Recording: [Report-6.mp4](https://drive.google.com/file/d/1TnOoDnpb43xJFQz4G97ScpDLLE0PsgOr/view?usp=sharing)

# 

# 

# 

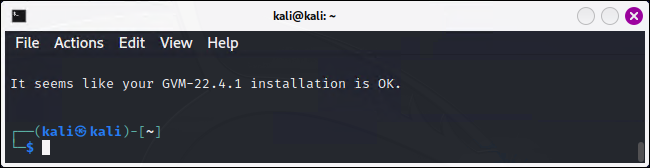
# Appendix - Steps to perform a scan

The following steps were taken to perform the scan :

1. In EVE GVM is installed in Kali. Access your Kali install as you usually would. When using GVM it is sometimes best to start with a fresh wipe, but it is not always necessary.
2. To check if GVM is already installed type the following in a terminal:

* **“sudo gvm-check-setup”**

If it is already installed you will screen with this

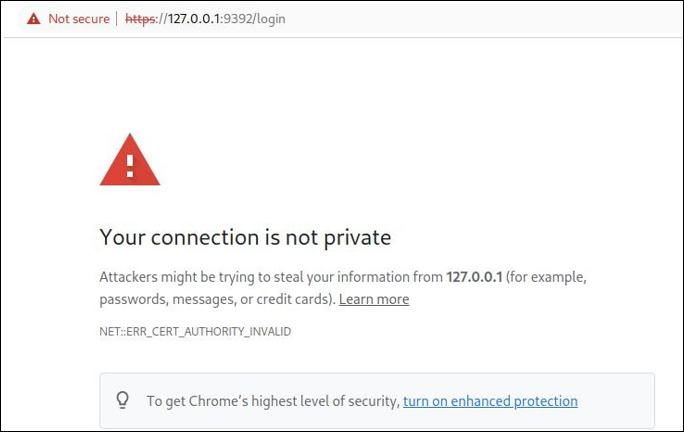


1. To start and stop the GVM

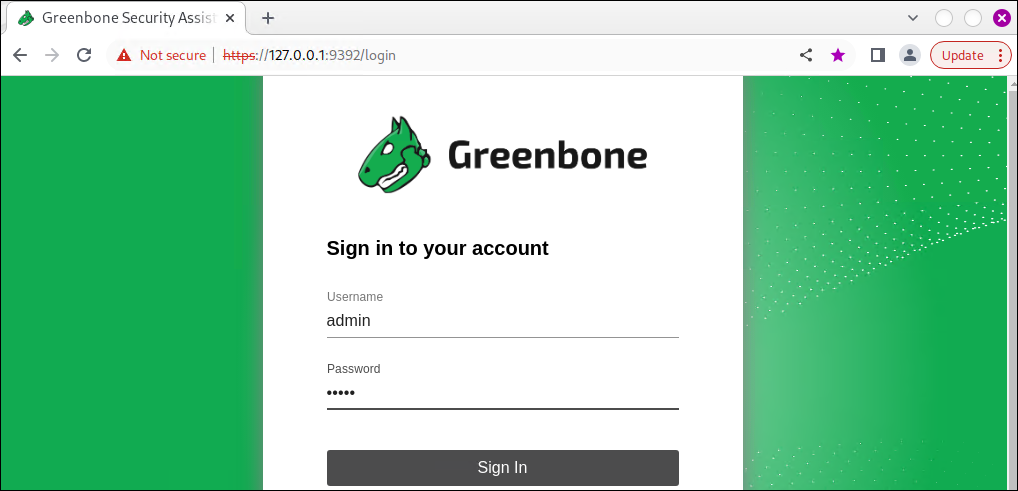
* **“sudo gvm-start**”: to start the GVM system
* **“sudo gvm-stop”**: to stop the tool

1. When GMV is running, open Chrome on Kali and navigate to **127.0.0.1:9392**.

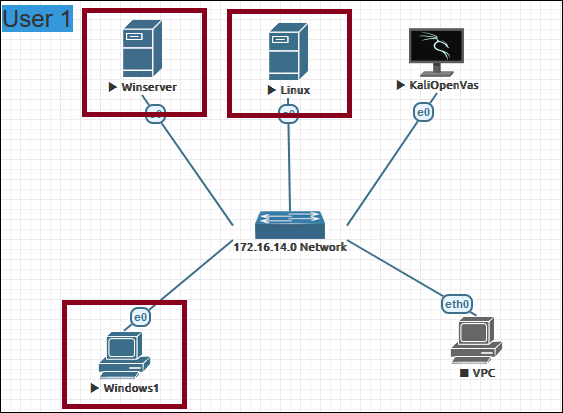
You may get a Chrome security warning, if so, select advanced then “Proceed to 127.0.0.1 (unsafe)”



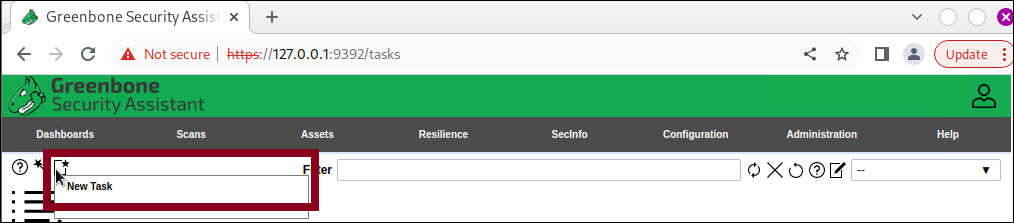
1. Login with Username admin and password admin



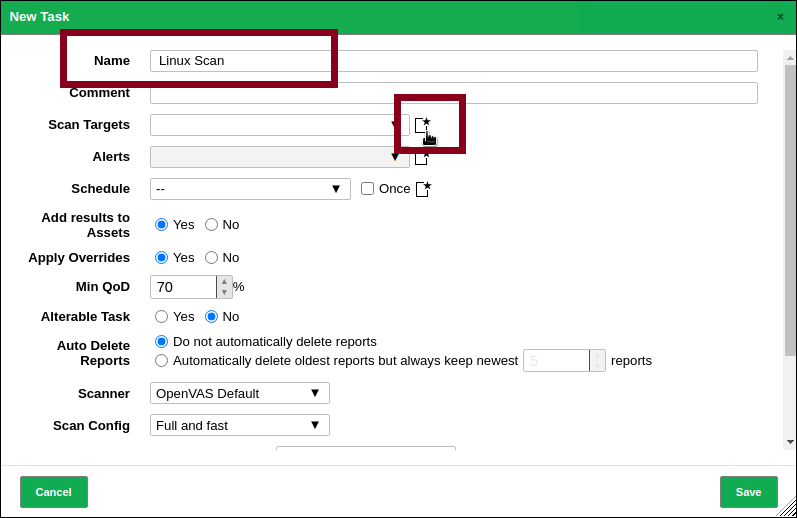
1. Start the machines you want to run the scan for in the EVE topology.



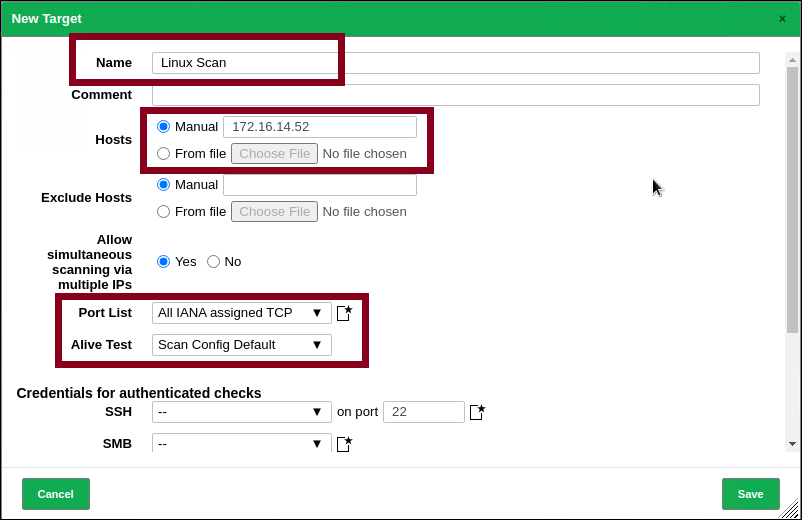
1. On the GVM dashboard select New Task.



1. Name the New Task and then, create a “New Target Scan”. If you already have one saved, select it from the “Scan Targets” list.



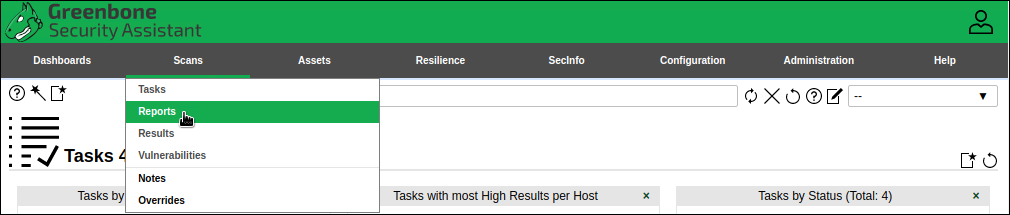
1. Fill in details for Target eg: Name, Host IP address, and type of Alice Test you wanna run, and then “Save”

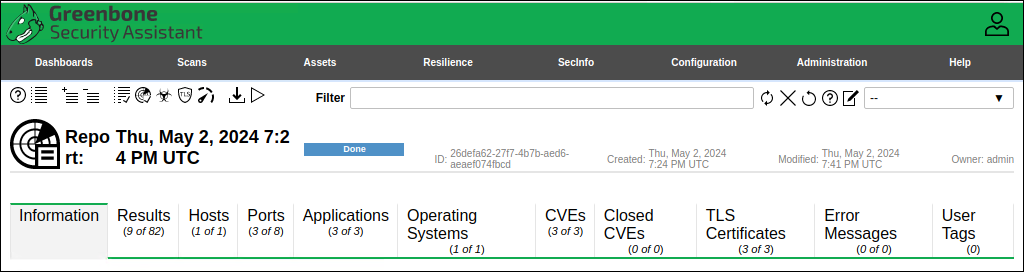


1. You will see the list of Scans with “New” Status. Click on “Start”



1. Repeat for all machines.
2. Once the Scan is finished. Go to Scans->Reports-> You will all the information such as Rsults, hosts, ports, Operating systems, CVEs, and error messages(if any).

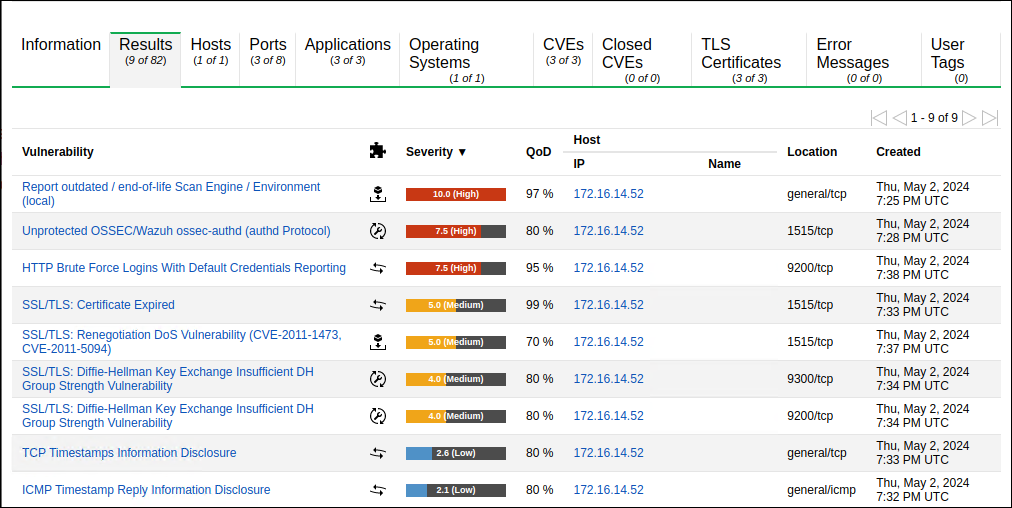




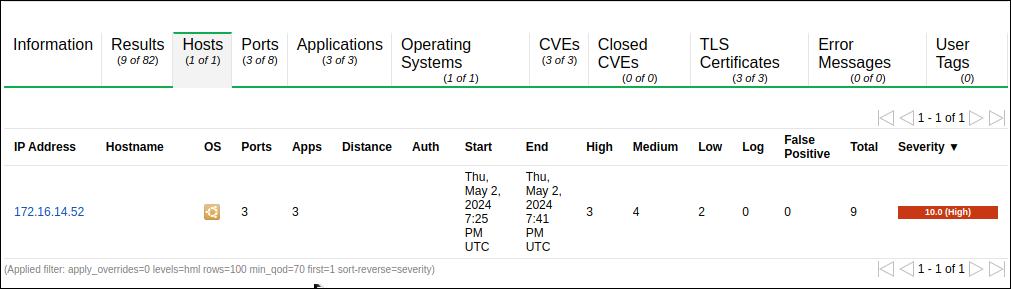
# Appendix -Linux

The Screenshots below show the Scan results of the Linux :

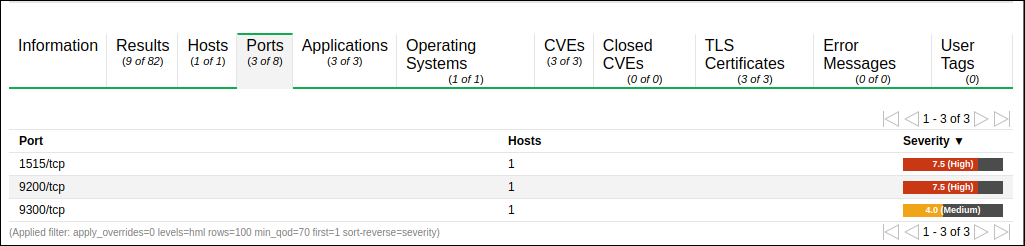
1. Results of All Vulnerabilities, their associated severities, QoD, Host IP, location, time, and date of creation.



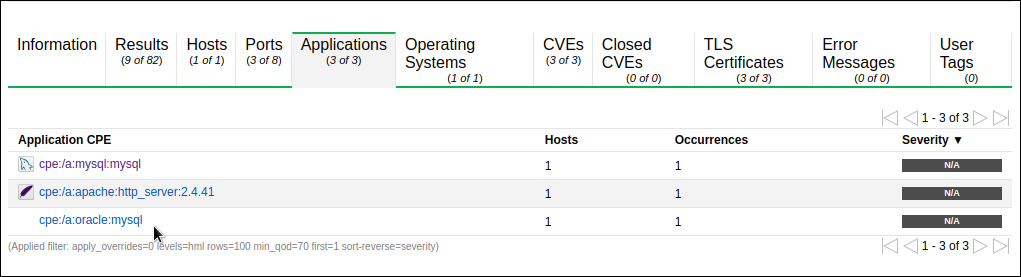
1. Host Details.



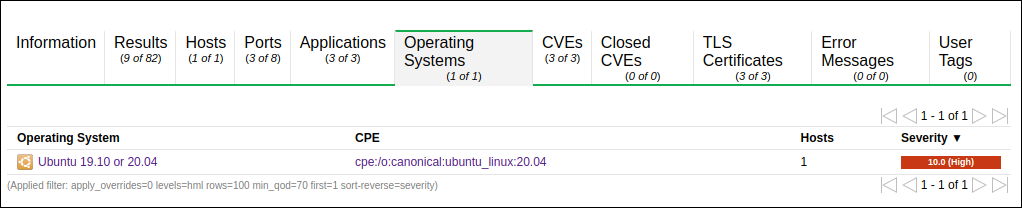
1. Ports where scan happened.



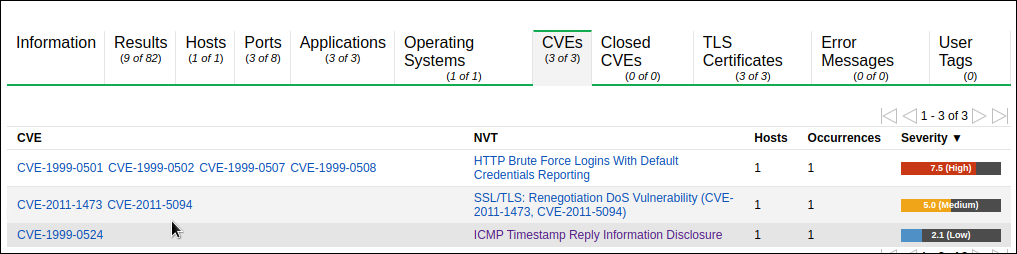
1. Applications that were not able to show any severities.



1. Operating system information.



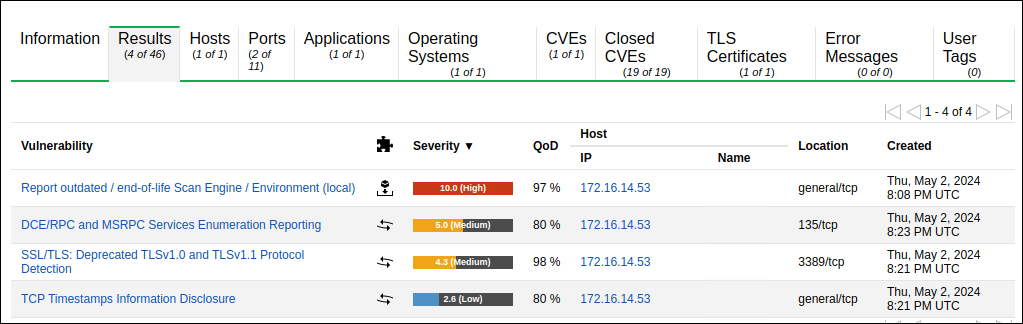
1. Associated CVEs



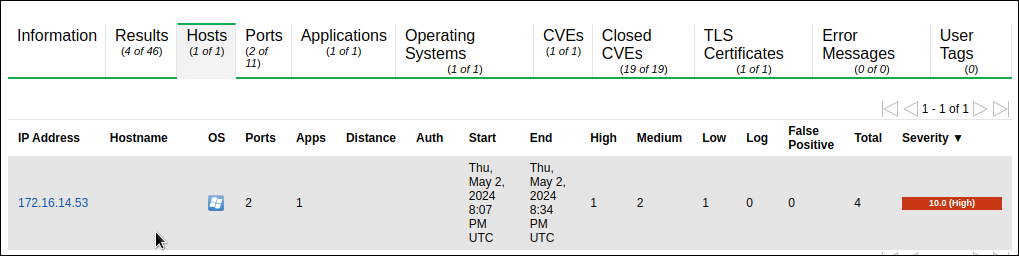
# Appendix -WinServer

The Screenshots below show the Scan results of the Windows server

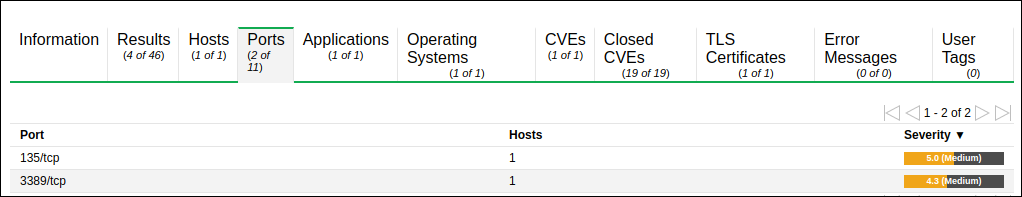
1. Results of All Vulnerabilities, their associated severities, QoD, Host IP, location, time, and date of creation.



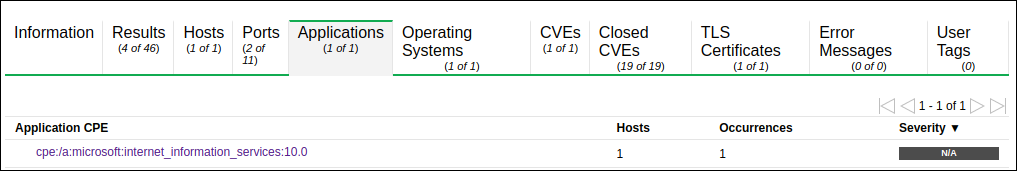
1. Host Details.



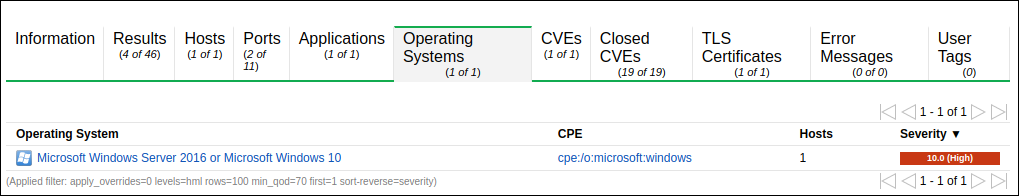
1. Ports where scan happened.



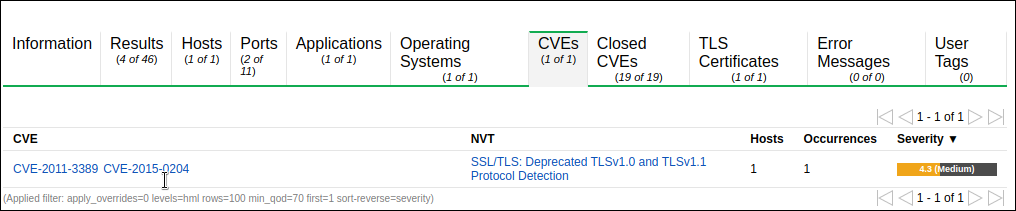
1. Applications that were not able to show any severities.



1. Operating system information.



1. Associated CVEs



# 

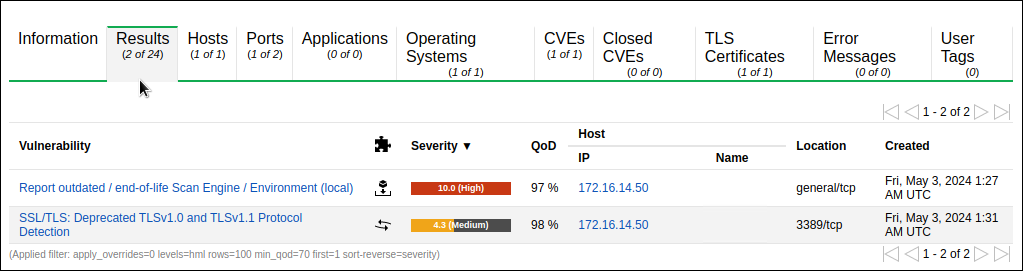
# 

# 

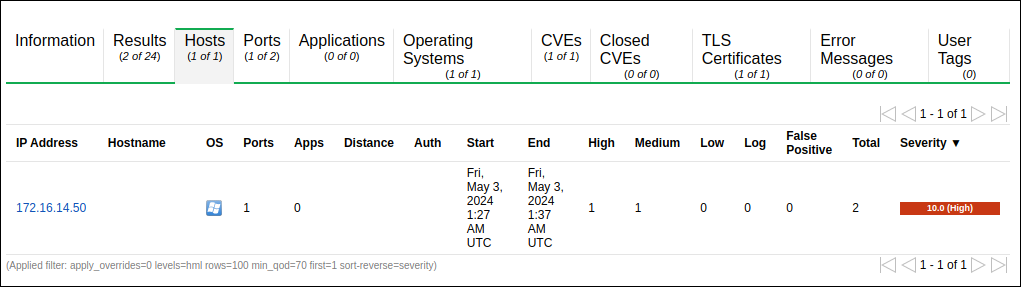
# Appendix -Windows workstation

The Screenshots below show the Scan results of the Windows workstation :

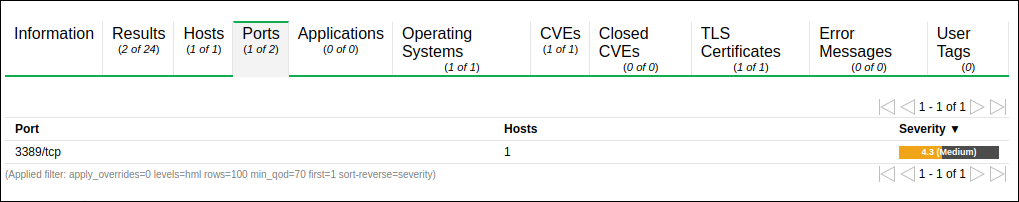
1. Results of All Vulnerabilities, their associated severities, QoD, Host IP, location, time, and date of creation.



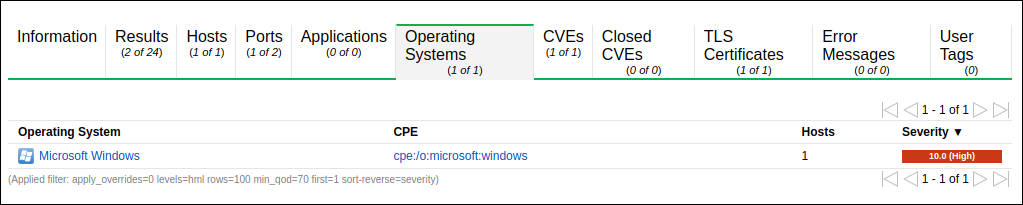
1. Host Details.



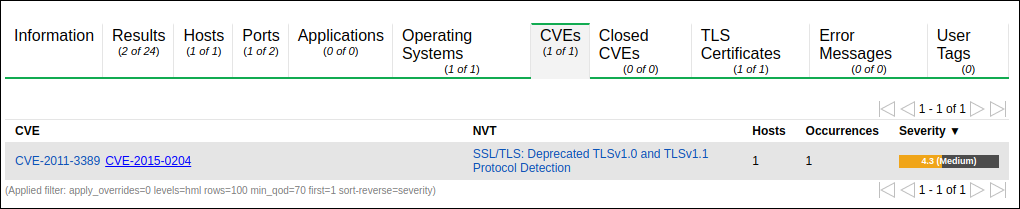
1. Ports where scan happened.



1. Operating system information.



1. Associated CVEs



# References

1. Guiverc. (2023, March 23). Ubuntu Fridge | Ubuntu 20.04.6 LTS Released. <https://fridge.ubuntu.com/2023/03/23/ubuntu-20-04-6-lts-released/#:~:text=The%20Ubuntu%20team%20is%20pleased,on%20Secure%20Boot%20enabled%20systems>.
2. Robinharwood. (2024, April 2). What’s new in Windows Server 2022. Microsoft Learn. <https://learn.microsoft.com/en-us/windows-server/get-started/whats-new-in-windows-server-2022>
3. Greenbone AG. (n.d.). Our solutions in comparison. <https://www.greenbone.net/wp-content/uploads/solution_comparison_EN.pdf>
4. CVE - CVE. (n.d.). <https://cve.mitre.org/>
5. NVD - Home. (n.d.). <https://nvd.nist.gov/>
6. Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology, U.S. Department of Commerce. (n.d.). Release Search - NIST Risk Management Framework | CSRC | CSRC. <https://csrc.nist.rip/Projects/risk-management/sp800-53-controls/release-search#!/controls?version=5.1&family=AC>
7. Wazuh. (n.d.). Using password authentication - Additional security options. <https://documentation.wazuh.com/current/user-manual/agent/agent-enrollment/security-options/using-password-authentication.html#password-authentication-linux-unix-endpoint>
8. Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology, U.S. Department of Commerce. (n.d.-b). Release Search - NIST Risk Management Framework | CSRC | CSRC. <https://csrc.nist.rip/Projects/risk-management/sp800-53-controls/release-search#!/families?version=5.1>
9. CSF Tools. (2021, December 5). NIST Special Publication 800-53 - CSF Tools. CSF Tools - the Cybersecurity Framework for Humans. <https://csf.tools/reference/nist-sp-800-53/>