CYBLACK SOC ACADEMY

VULNERABILITY ASSESMENT REPORT

Submitted by TEAM 1 (ALPHA TECH)

PRINCE ORUMA

ONYEKA VANESSA

DEBBIE AYOOLA

IVIE OSOIYE

KEHINDE OJUKO

MODUPE OJUGBELE

NDABEZINHLE REGAL SIBANDA

VERONICA OJUKWU

1.0 EXECUTIVE
SUMMARY 3

2.0 OPEN SSH
CONFIGURATION.....3

3.0 APACHE HTTP CONFIGURATION 4

4.0 NGINX
CONFIGURATION.....5

4.1 CREDENTIAL SCAN.....6

5.0 WEB APPLICATION
SCAN.....7

6.0 EMAIL CONFIGURATION FOR AUTOMATED
REPORTING.....8

6.1 CVE RESEARCH AND ANALYSIS 9

7.0 CONCLUSION 10

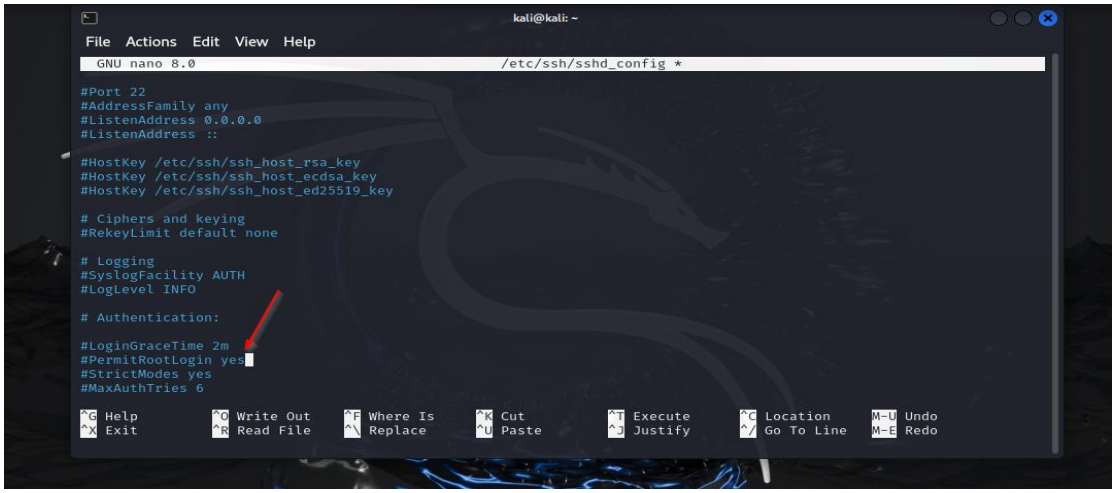
1.0 EXECUTIVE SUMMARY

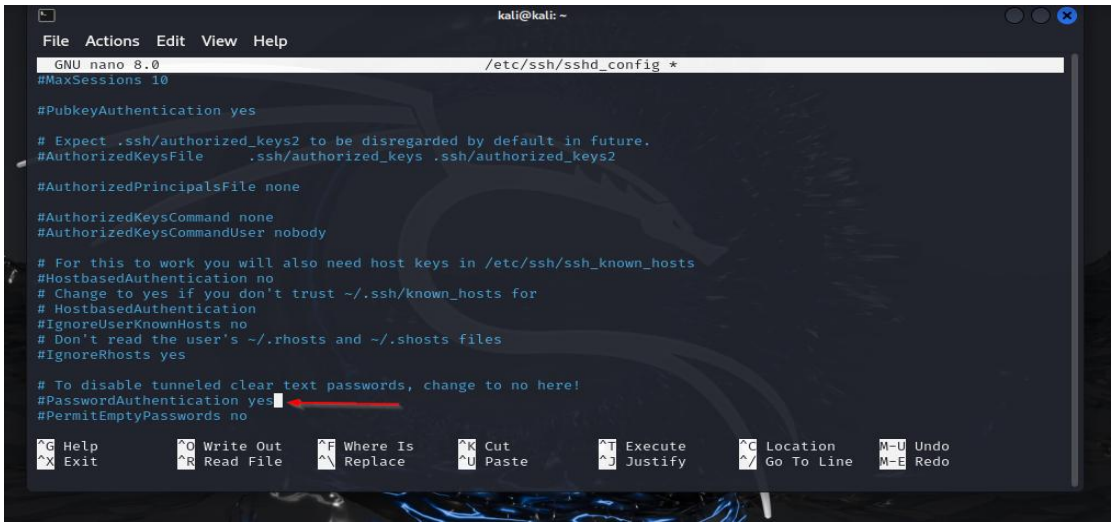
Our team has been assigned the role of cybersecurity analysts at CyberTech Solutions, a company focused on managing and securing client networks. Our primary objective is to assess the company's internal infrastructure to identify vulnerabilities within its Linux systems and web applications. This includes performing credentialed scans on the Linux servers, conducting security assessments of the web applications, and configuring Nessus to automate the process of sending vulnerability reports via email, ensuring continuous monitoring and improvement of the company's security posture.

2.0 OPEN SSH CONFIGURATION

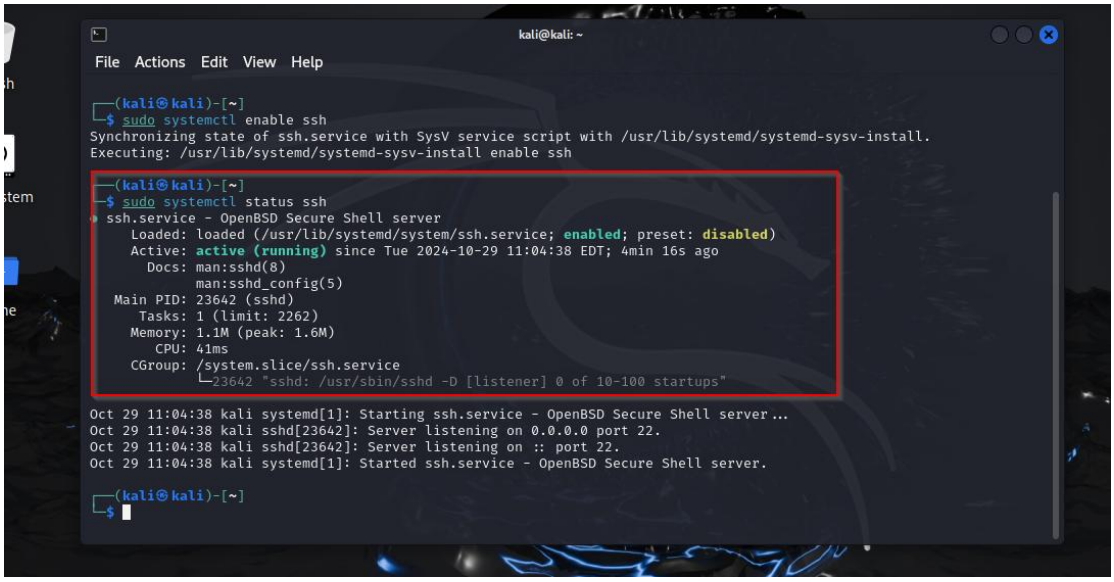
The process of configuring SSH on the system involved several key steps to ensure secure and functional remote access. First, the OpenSSH configuration file located at [/etc/ssh/sshd_config] was edited to modify essential settings. The configuration for root login was changed to [PermitRootLogin yes], allowing root access via SSH. Additionally, the authentication method was updated by enabling password-based authentication, changing the line [PasswordAuthentication no] to [PasswordAuthentication yes]. After saving these changes and exiting the editor, the SSH service was restarted using [sudo systemctl restart sshd] to apply the new configuration settings.

In parallel, SSH was installed on the system. The package list was first updated with [sudo apt update], followed by the installation of the OpenSSH server using [sudo apt install openssh-server -y]. The SSH service was started and enabled to launch automatically on system boot with [sudo systemctl start ss] and [sudo systemctl enable ssh]. Finally, the status of the SSH service was checked to ensure it was running properly with the command [sudo systemctl status ssh]. To verify the correct application of the cipher settings, a connection test was conducted with [ssh -oCiphers=arcfour], confirming the changes were active. Overall, the SSH configuration was successfully completed, enabling secure remote access with updated settings.





Restart SSH, the check the status with [sudo systemctl status ssh]

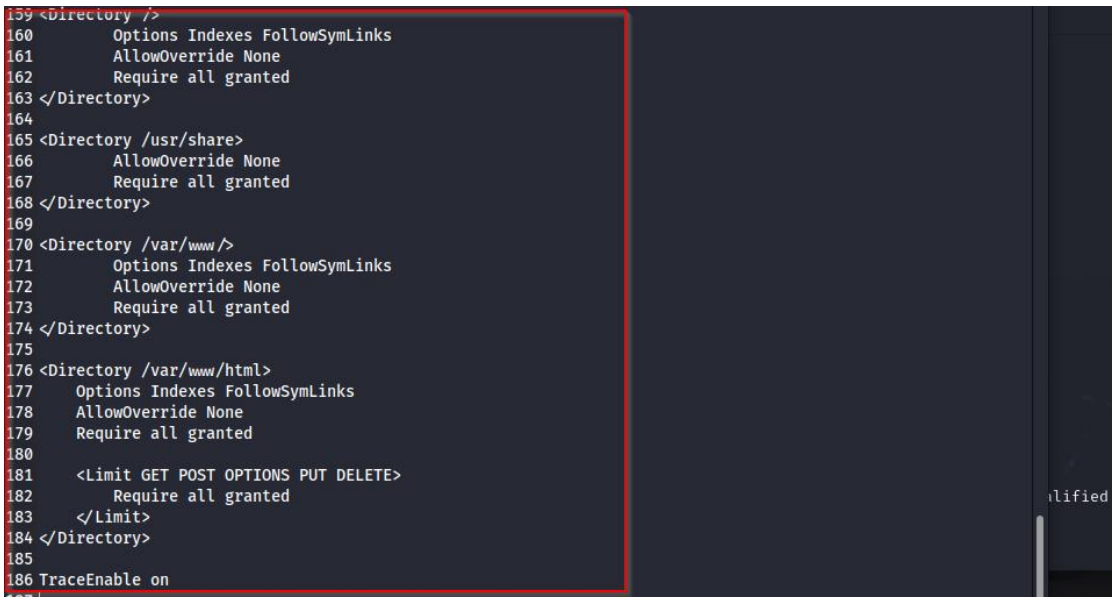


3.0 APACHE HTTP CONFIGURATION

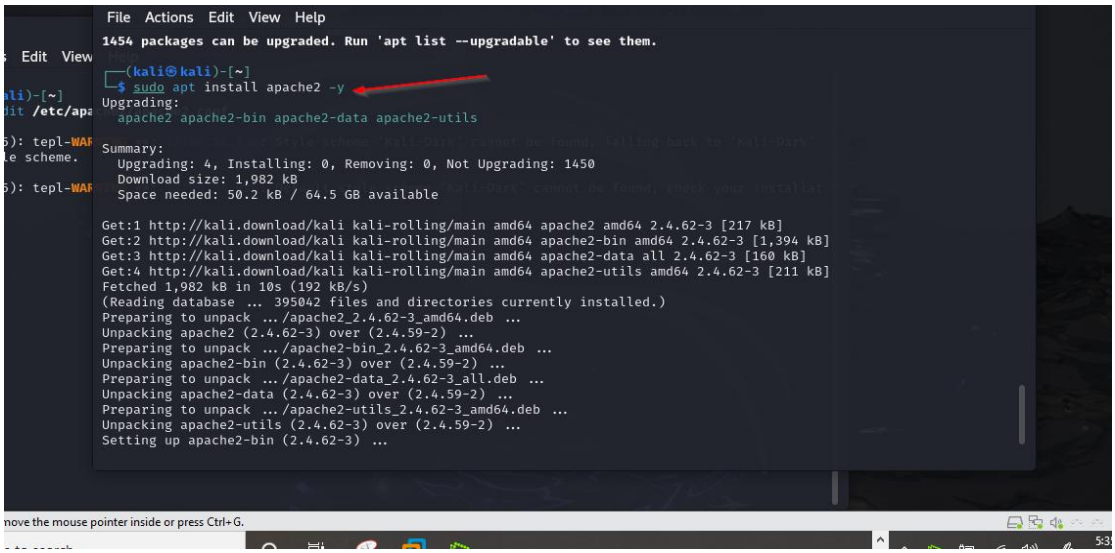
The process of configuring the Apache web server involved a series of steps to enable specific HTTP methods and ensure the server was installed and running properly. First, the Apache configuration file was accessed for editing, either the main configuration file located at [/etc/apache2/apache2.conf] or the virtual host file at [/etc/apache2/sites-available/000-default.conf`] using the command [sudo nano /etc/apache2/apache2.conf]. Within this file, modifications were made to allow HTTP methods like DELETE and PUT by adding a `<Limit>` directive under the `<Directory /var/www/html>` section. The configuration was updated to include the following block of code: `<Limit GET POST OPTIONS PUT DELETE> Require all granted </Limit>`, and `TraceEnable on` was set to allow trace requests. After making these changes, the file was saved and exited.

To apply the changes, the Apache service was restarted using the command `sudo systemctl restart apache2`. To verify the new configuration, a test was performed using `curl` commands to check if the DELETE and PUT methods were successfully enabled. Running `curl -v -X DELETE http://<your-kali-ip>` and `curl -v -X PUT http://<your-kali-ip>` returned responses confirming that these methods were allowed.

In addition to configuring the HTTP methods, Apache was installed and configured to run on the system. The installation process began with updating the package list using `sudo apt update`, followed by installing Apache2 with `sudo apt install apache2 -y`. The Apache service was then started with `sudo systemctl start apache2` and enabled to automatically start on system boot using `sudo systemctl enable apache2`. Finally, the status of the Apache service was checked to confirm it was running successfully using `sudo systemctl status apache2`. The configuration changes and installation were completed successfully, ensuring that the Apache server was fully functional with the desired HTTP methods enabled.




```
160         Options Indexes FollowSymLinks
161         AllowOverride None
162         Require all granted
163 </Directory>
164
165 <Directory /usr/share>
166         AllowOverride None
167         Require all granted
168 </Directory>
169
170 <Directory /var/www/>
171         Options Indexes FollowSymLinks
172         AllowOverride None
173         Require all granted
174 </Directory>
175
176 <Directory /var/www/html>
177     Options Indexes FollowSymLinks
178     AllowOverride None
179     Require all granted
180
181     <Limit GET POST OPTIONS PUT DELETE>
182         Require all granted
183     </Limit>
184 </Directory>
185
186 TraceEnable on
```


File Actions Edit View Help
1454 packages can be upgraded. Run 'apt list --upgradable' to see them.
(kali㉿kali)-[~]
$ sudo apt install apache2 -y
Upgrading:
apache2 apache2-bin apache2-data apache2-utils
Summary:
Upgrading: 4, Installing: 0, Removing: 0, Not Upgrading: 1450
Download size: 1,982 kB
Space needed: 50.2 kB / 64.5 GB available
Get:1 http://kali.download/kali kali-rolling/main amd64 apache2 amd64 2.4.62-3 [217 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 apache2-bin amd64 2.4.62-3 [1,394 kB]
Get:3 http://kali.download/kali kali-rolling/main amd64 apache2-data all 2.4.62-3 [160 kB]
Get:4 http://kali.download/kali kali-rolling/main amd64 apache2-utils amd64 2.4.62-3 [211 kB]
Fetched 1,982 kB in 10s (192 kB/s)
(Reading database ... 395042 files and directories currently installed.)
Preparing to unpack .../apache2_2.4.62-3_amd64.deb ...
Unpacking apache2 (2.4.62-3) over (2.4.59-2) ...
Preparing to unpack .../apache2-bin_2.4.62-3_amd64.deb ...
Unpacking apache2-bin (2.4.62-3) over (2.4.59-2) ...
Preparing to unpack .../apache2-data_2.4.62-3_all.deb ...
Unpacking apache2-data (2.4.62-3) over (2.4.59-2) ...
Preparing to unpack .../apache2-utils_2.4.62-3_amd64.deb ...
Unpacking apache2-utils (2.4.62-3) over (2.4.59-2) ...
Setting up apache2-bin (2.4.62-3) ...
nove the mouse pointer inside or press Ctrl+G.

4.0 NGINX CONFIGURATION

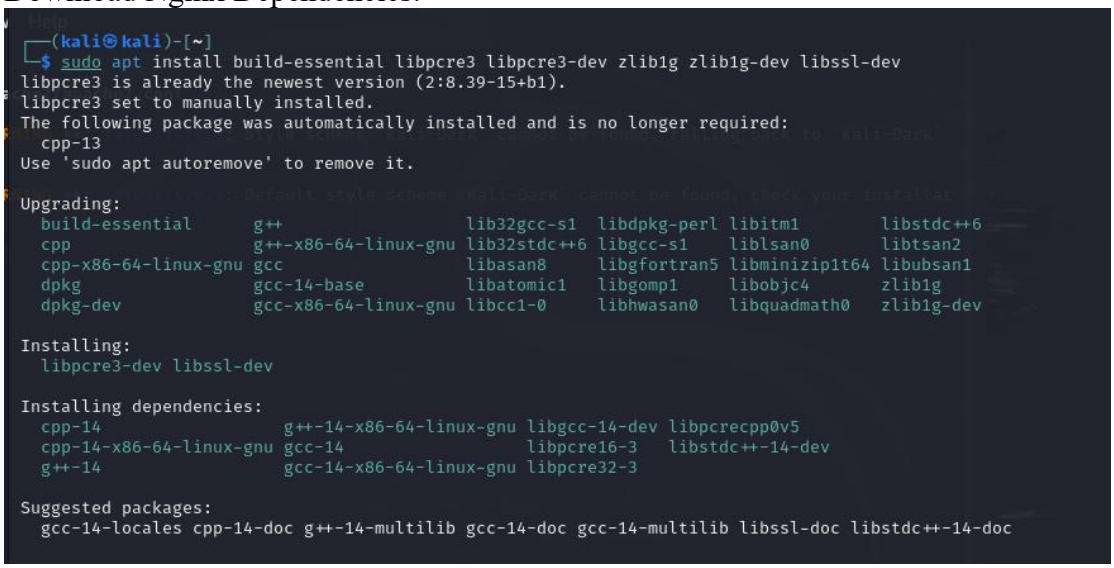
The process of installing Nginx from source involved several steps to ensure proper setup. First, the Nginx package (`nginx-1.15.5.tar.gz`) was downloaded using the command `wget http://nginx.org/download/nginx-1.15.5.tar.gz`. Next, the necessary dependencies for compiling Nginx were installed by updating the package list with `sudo apt update` and then installing required libraries with the command `sudo apt install build-essential libpcre3 libpcre3-dev zlib1g zlib1g-dev libssl-dev`.

After the dependencies were installed, the downloaded tarball was extracted using `tar -zxvf nginx-1.15.5.tar.gz`. The next step involved navigating to the extracted Nginx directory with `cd nginx-1.15.5`, followed by configuring and compiling the software using the commands `./configure`, `make`, and `sudo make install`.

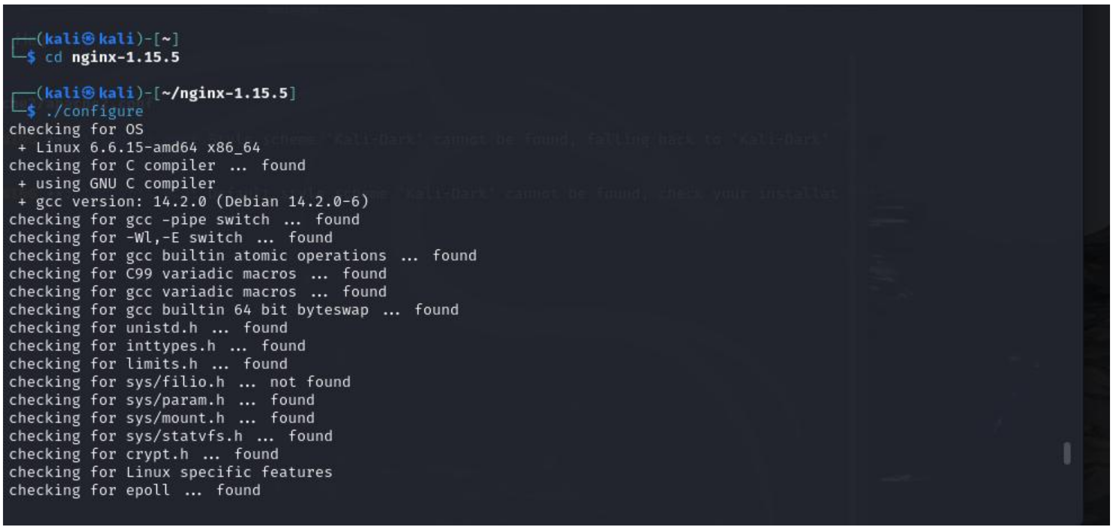
Once the installation was complete, Nginx was started with `sudo /usr/local/nginx/sbin/nginx`. Finally, to verify that Nginx was running correctly, the command `sudo /usr/local/nginx/sbin/nginx -t` was used, confirming that the installation was successful and Nginx was operational.

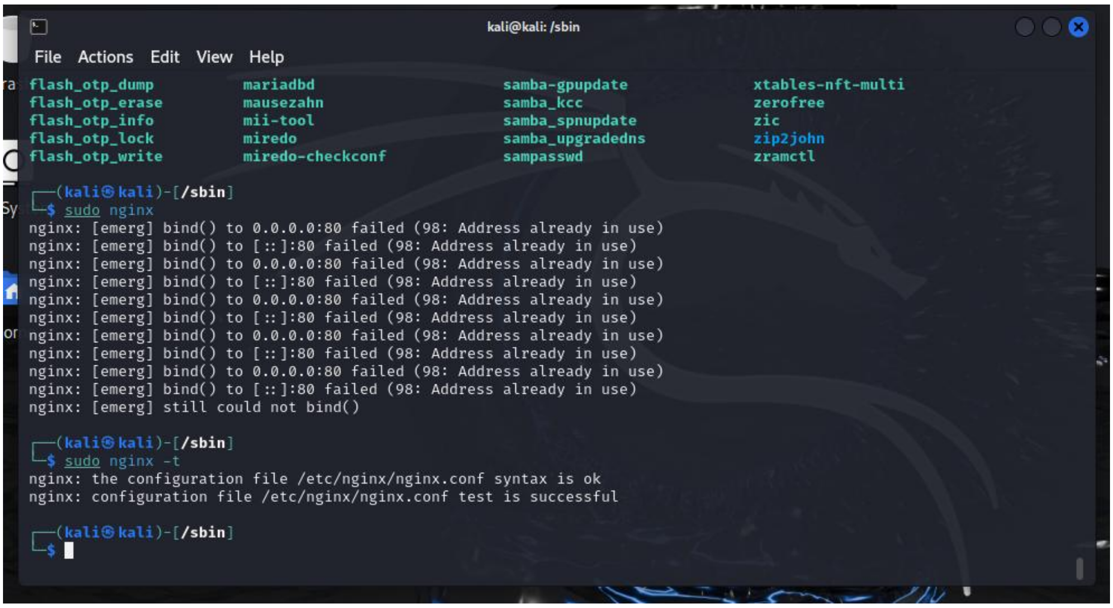
Download Nginx:

Download Nginx Dependencies:



Extract the file:





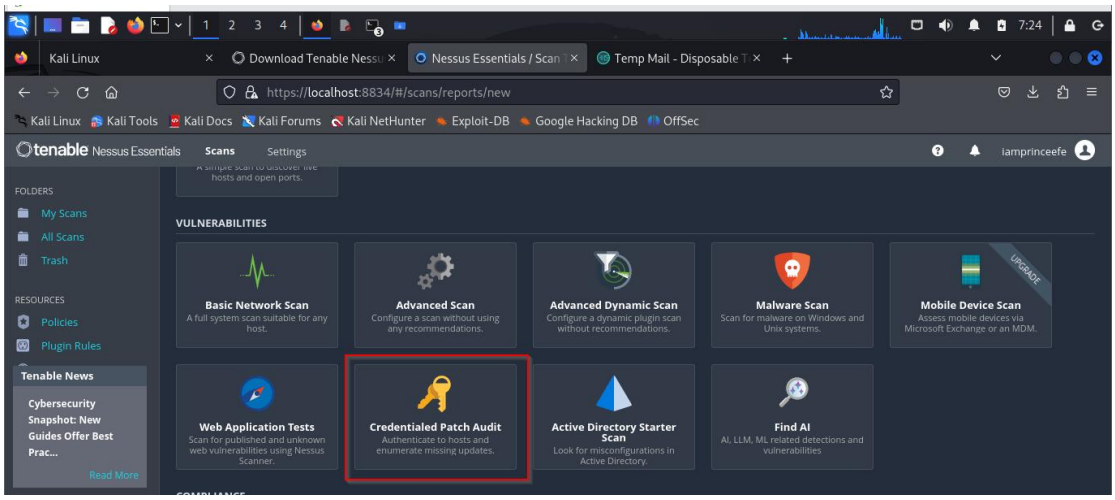
Test the Nginx:

VULNERABILITY ASSESMENT USING NESSUS

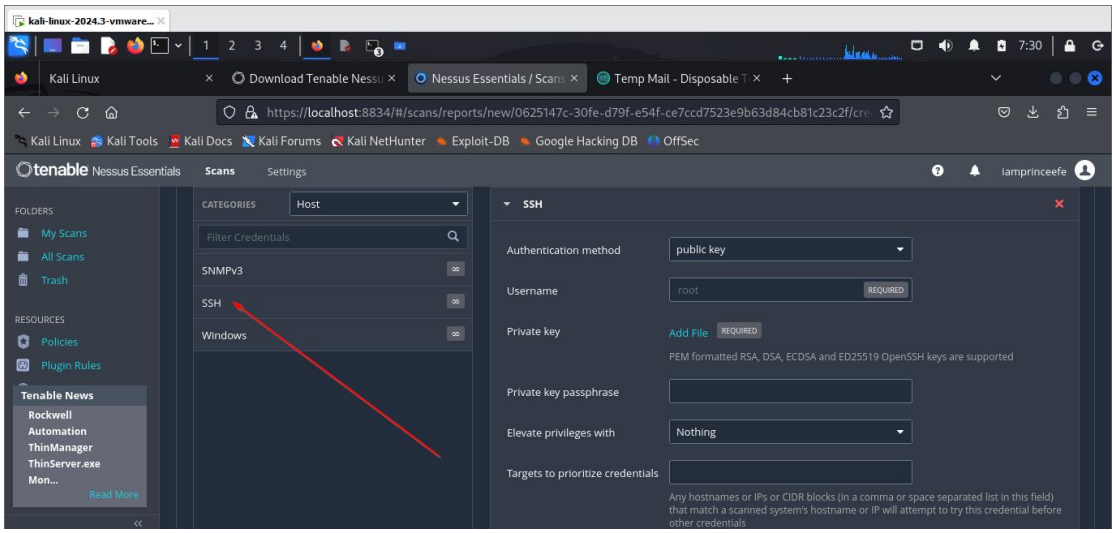
4.1 CREDENTIAL SCAN

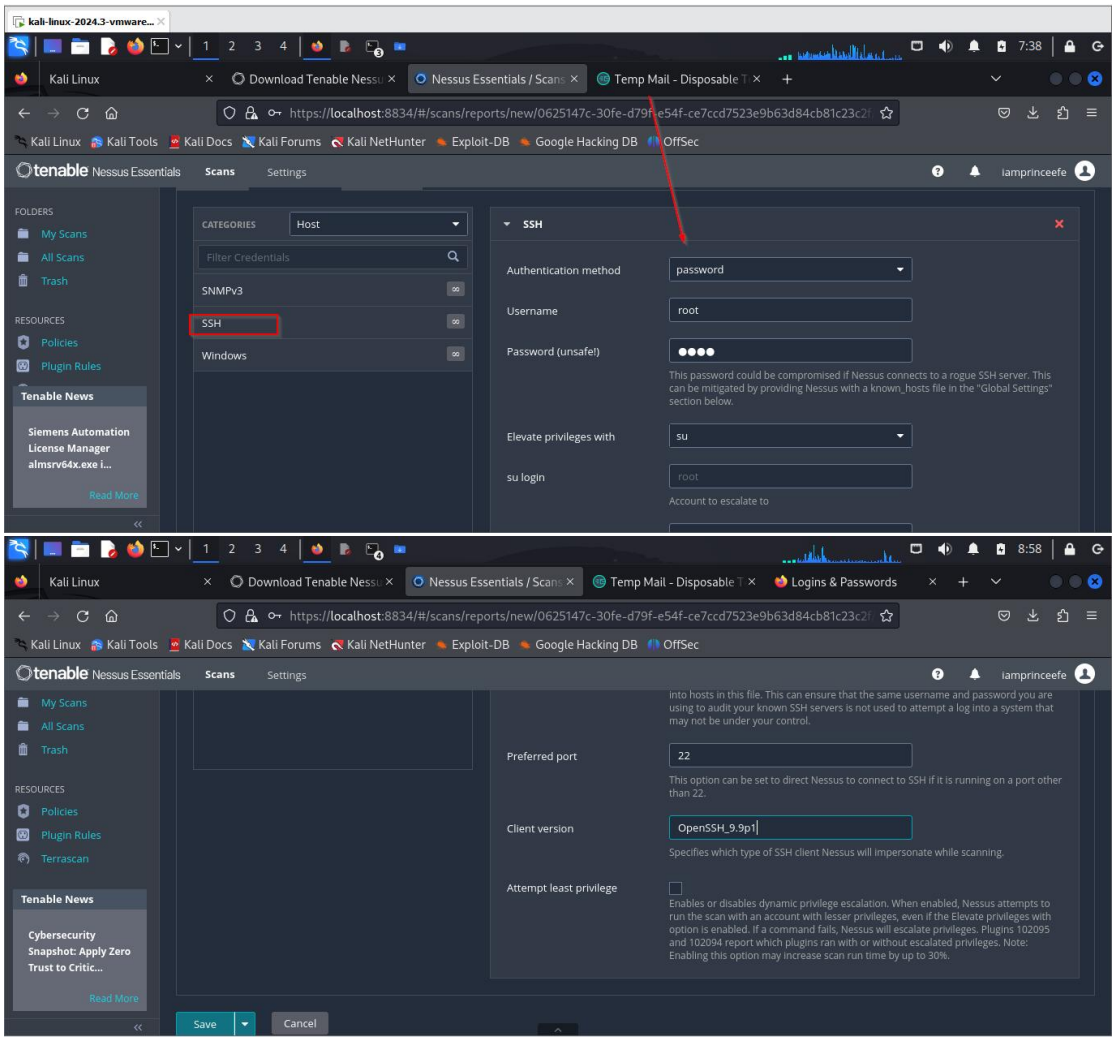
After downloading and configuring Nessus, the team conducted a credentialed scan to assess the security posture of the Linux server, in accordance with the company's security policies. To set up the scan, the necessary credentials were entered by selecting SSH in the Credentials tab. The username was configured as "root," and the appropriate root password was provided. For privilege escalation, "su" was selected, ensuring that the scan had sufficient permissions to perform a comprehensive audit.

Additionally, the **Custom Password Prompt** was configured to "password" to align with the system's authentication requirements. The team ran the command ssh -V to check the version of OpenSSH running on the server, which was then input into the scan configuration to ensure compatibility and accurate vulnerability assessment for the specific OpenSSH version. This credentialed scan forms part of the ongoing effort to ensure that the server remains compliant with security policies and is free from vulnerabilities.



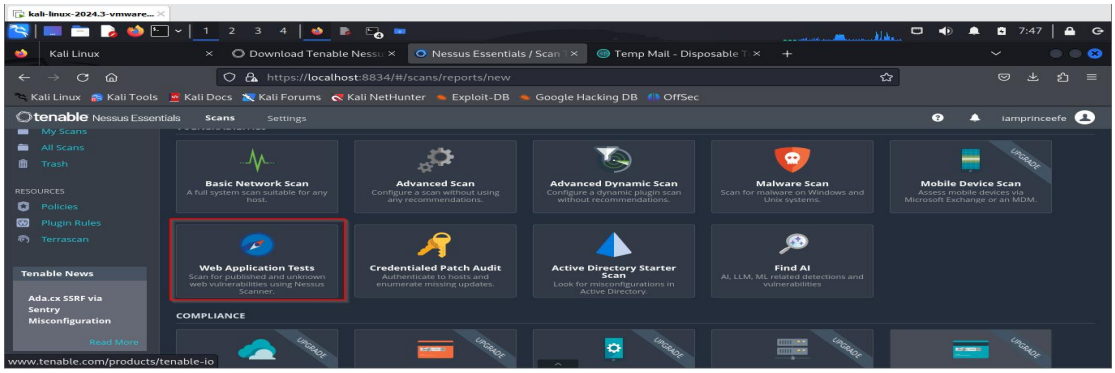
Select SSH:

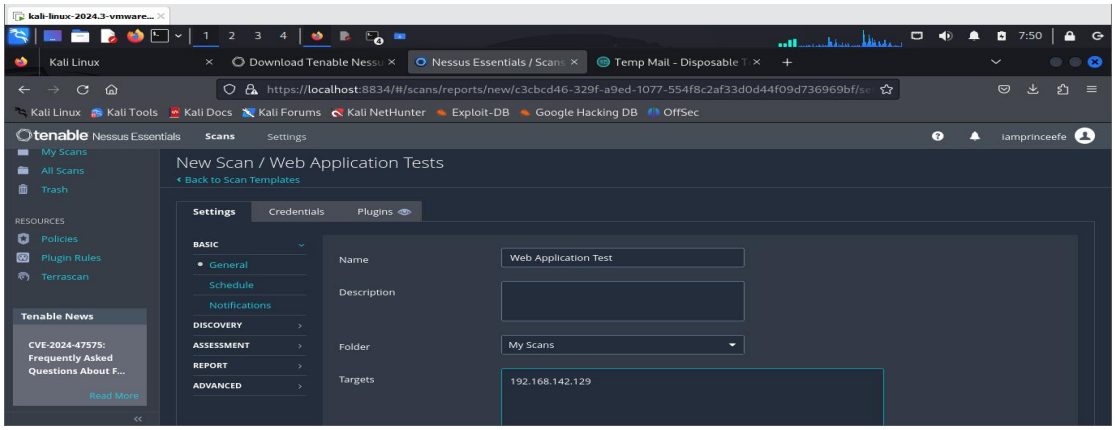




4.1 WEB APPLICATION SCAN

In order to assess the security of a web application running on the Linux server, the IT team conducted a web application scan using Nessus. The process began by creating a Web Application Test scan within Nessus, using the default scan settings without the need for authentication configuration. Once the scan was initiated, the team ran it and thoroughly analyzed the results to identify any vulnerabilities present within the web application. This assessment is part of the ongoing effort to ensure that the web application is secure and compliant with the organization's security standards.



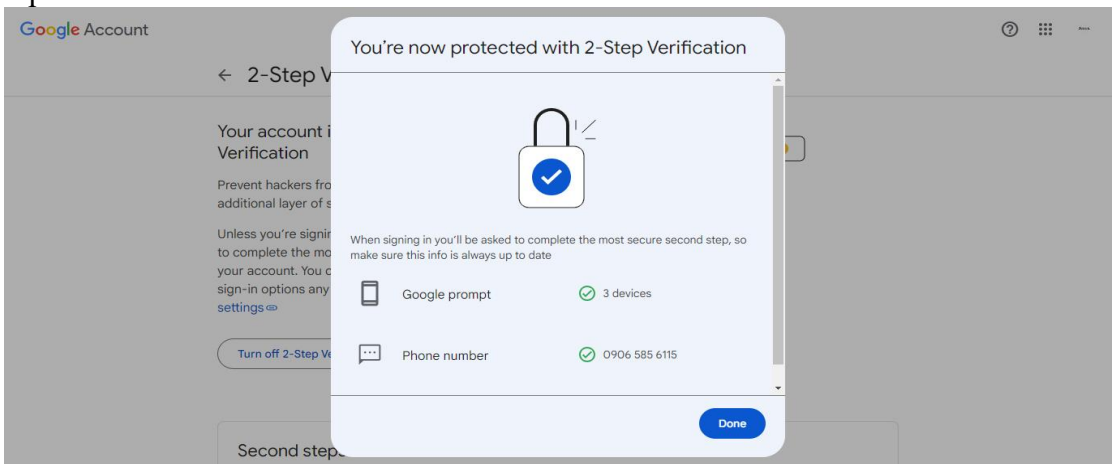


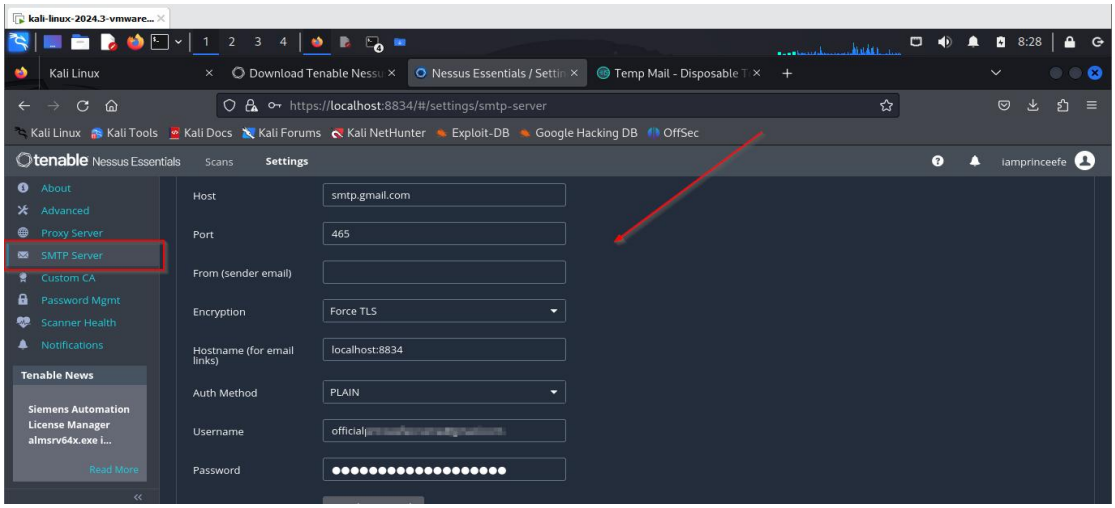
6.0: Email Configuration for Automated Reporting

To configure SMTP for email reports in Nessus, the team followed a series of steps to ensure secure email communication. First, the SMTP Server settings were configured within Nessus by navigating to Nessus Settings > SMTP Server. The SMTP Host was set to `smtp.gmail.com`, with TLS encryption enabled and the Port set to `587` for secure transmission.

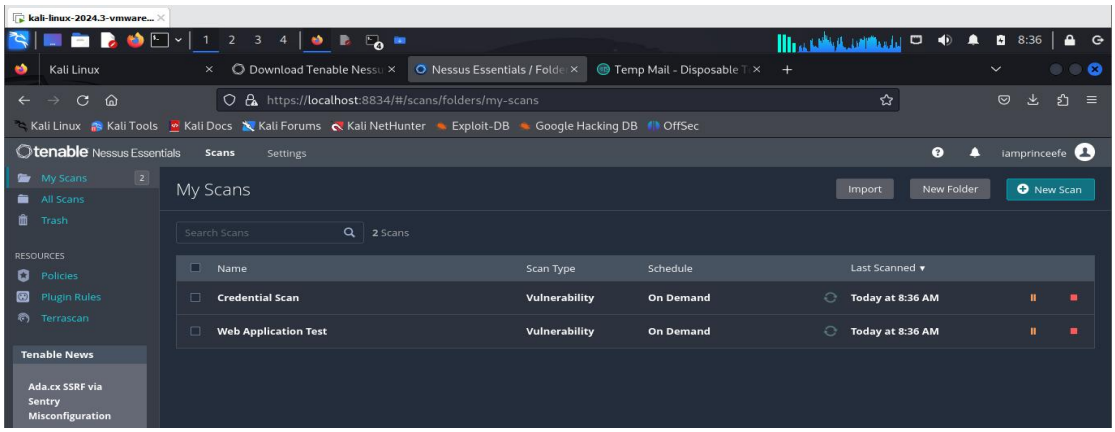
Next, to securely send email reports, the team generated a Gmail App Password. This was done by logging into the Gmail account, navigating to the Google Account Settings, and ensuring that 2-Step Verification was enabled. The team then accessed App Passwords under the Security section, created a unique app password specifically for Nessus, and copied the generated password for use in the configuration.

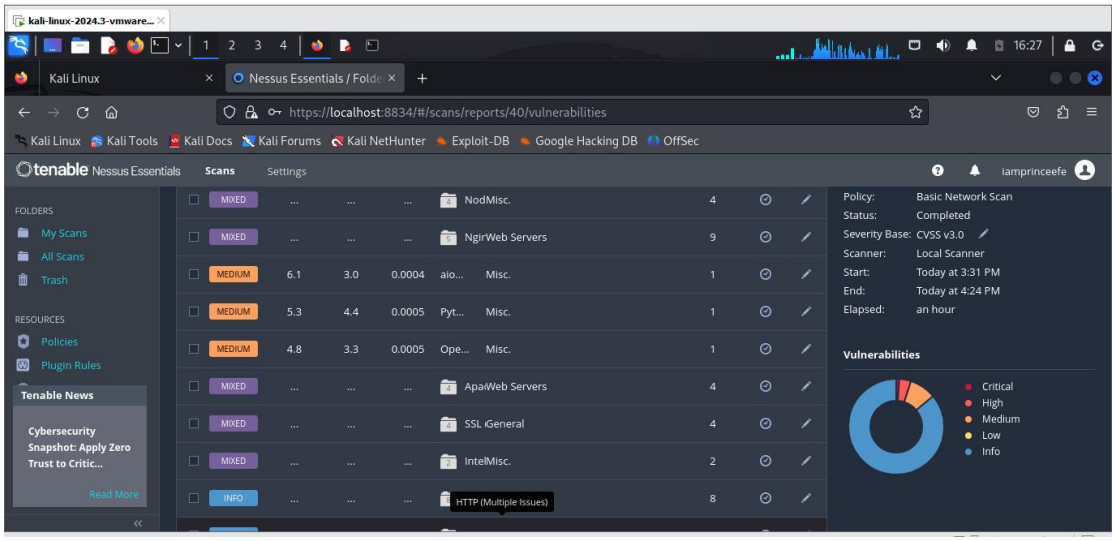
Finally, the SMTP credentials were entered into Nessus, using the Gmail address as the username and the app password generated earlier as the password. Both the From Address and To Address fields were set to the desired email addresses for sending and receiving the reports. This configuration ensures that Nessus can securely send email reports via Gmail's SMTP server.

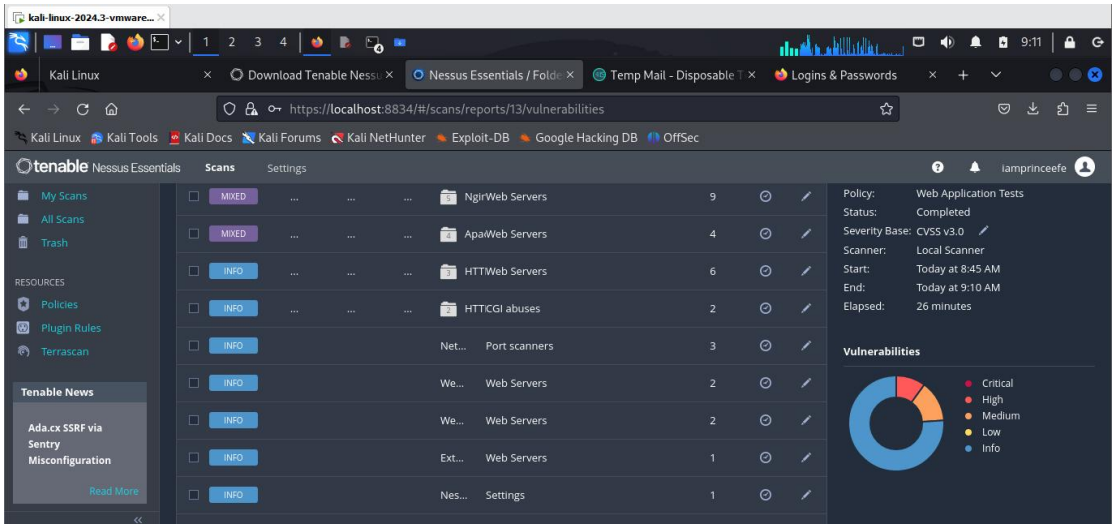


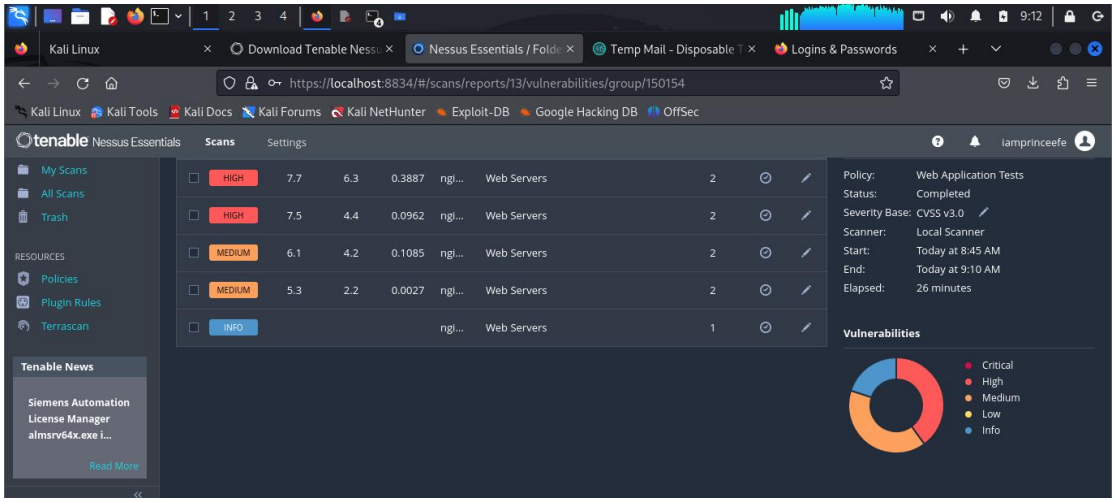


SCAN VULNERABILITIES:

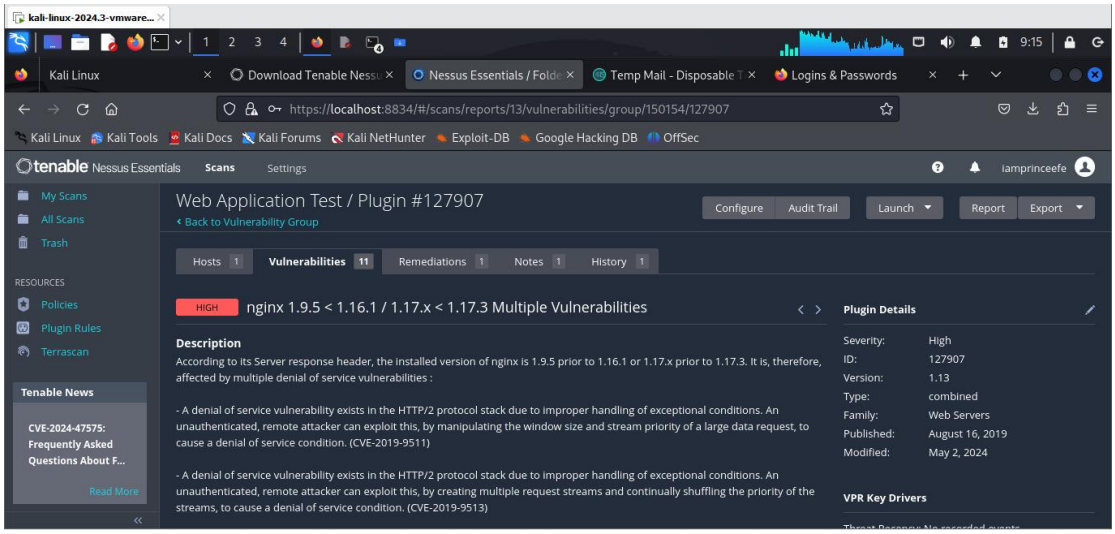


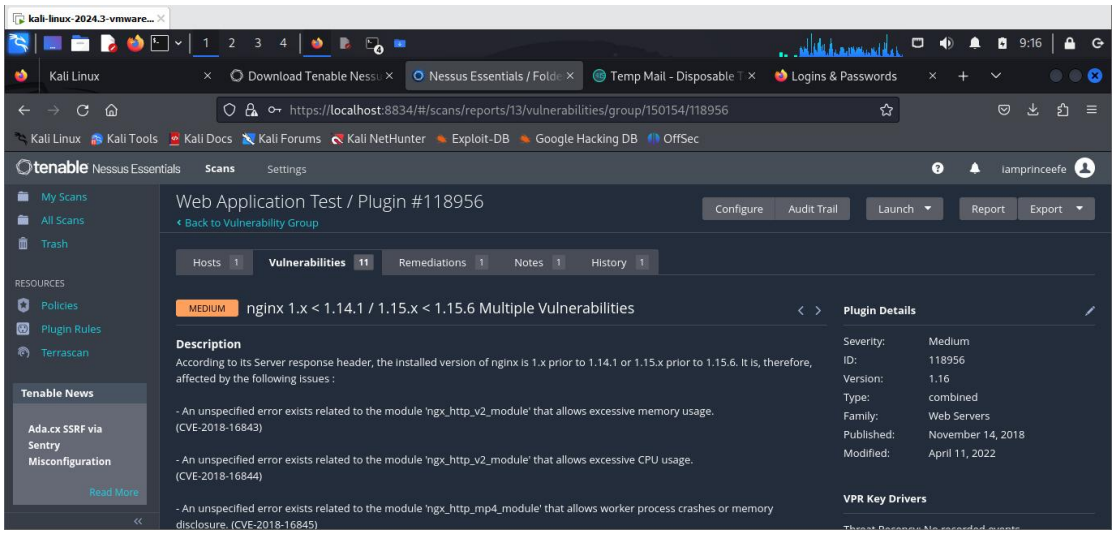


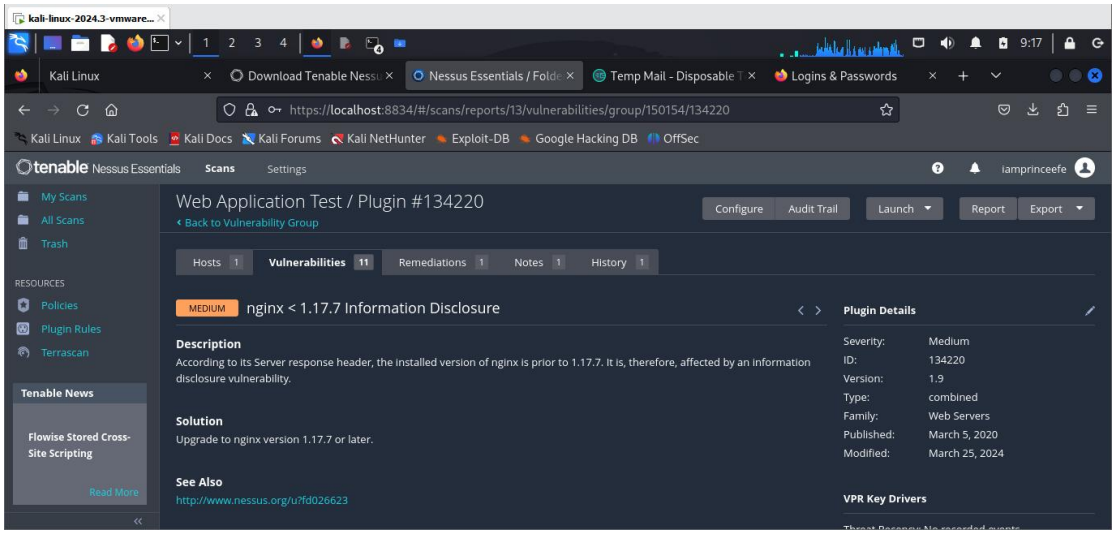

kali-linux-2024.3-vmware...
Kali Linux
Download Tenable Nessu
Nessus Essentials / Folde
Temp Mail - Disposable
Logins & Passwords
https://localhost:8834/#/scans/reports/13/vulnerabilities
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
tenable Nessus Essentials
Scans
Settings
iamprinceefe
My Scans
All Scans
Trash
RESOURCES
Policies
Plugin Rules
Terrascan
Tenable News
Ada.cx SSRF via Sentry Misconfiguration
Read More
MIXED NgirWeb Servers 9
MIXED ApaWeb Servers 4
INFO HTTWeb Servers 6
INFO HTTCGI abuses 2
INFO Net... Port scanners 3
INFO We... Web Servers 2
INFO We... Web Servers 2
INFO Ext... Web Servers 1
INFO Nes... Settings 1
Policy: Web Application Tests
Status: Completed
Severity Base: CVSS v3.0
Scanner: Local Scanner
Start: Today at 8:45 AM
End: Today at 9:10 AM
Elapsed: 26 minutes
Vulnerabilities
Critical
High
Medium
Low
Info

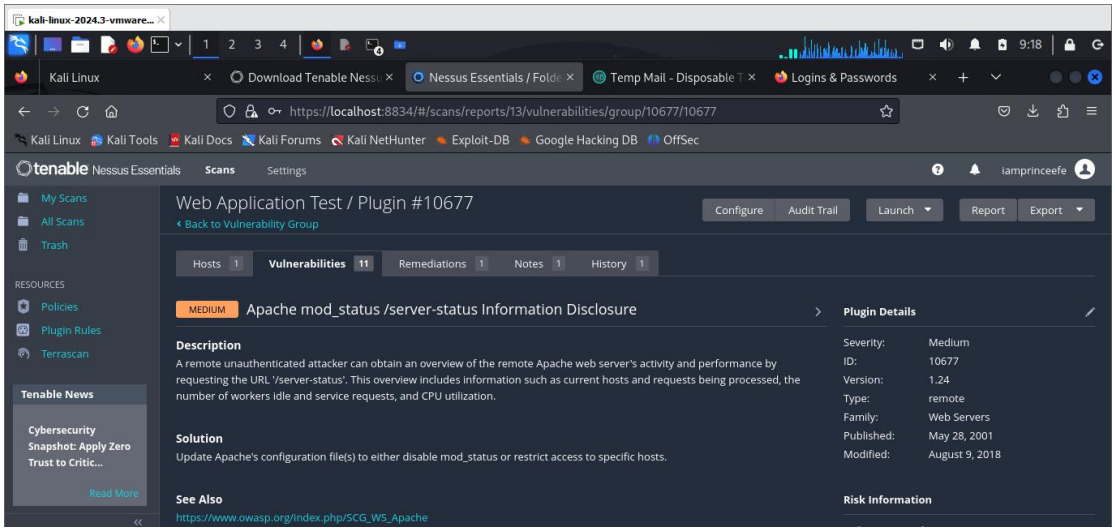

Kali Linux
Download Tenable Nessu
Nessus Essentials / Folde
Temp Mail - Disposable
Logins & Passwords
https://localhost:8834/#/scans/reports/13/vulnerabilities/group/150154
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
tenable Nessus Essentials
Scans
Settings
iamprinceefe
My Scans
All Scans
Trash
RESOURCES
Policies
Plugin Rules
Terrascan
Tenable News
Siemens Automation License Manager almsrv64x.exe i...
Read More
HIGH 7.7 6.3 0.3887 ngi... Web Servers 2
HIGH 7.5 4.4 0.0962 ngi... Web Servers 2
MEDIUM 6.1 4.2 0.1085 ngi... Web Servers 2
MEDIUM 5.3 2.2 0.0027 ngi... Web Servers 2
INFO ngi... Web Servers 1
Policy: Web Application Tests
Status: Completed
Severity Base: CVSS v3.0
Scanner: Local Scanner
Start: Today at 8:45 AM
End: Today at 9:10 AM
Elapsed: 26 minutes
Vulnerabilities
Critical
High
Medium
Low
Info


kali-linux-2024.3-vmware...
Kali Linux
Download Tenable Nessu
Nessus Essentials / Folde
Temp Mail - Disposable
Logins & Passwords
https://localhost:8834/#/scans/reports/13/vulnerabilities/group/150154/150154
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
tenable Nessus Essentials
Scans
Settings
iamprinceefe
My Scans
All Scans
Trash
RESOURCES
Policies
Plugin Rules
Terrascan
Tenable News
FY 2024 State and Local Cybersecurity Grant Progra...
Read More
Web Application Test / Plugin #150154
Back to Vulnerability Group
Configure
Audit Trail
Launch
Report
Export
Hosts 1
Vulnerabilities 11
Remediations 1
Notes 1
History 1
HIGH nginx 0.6.x < 1.20.1 1-Byte Memory Overwrite RCE
Description
According to its Server response header, the installed version of nginx is 0.6.18 prior to 1.20.1. It is, therefore, affected by a remote code execution vulnerability. A security issue in nginx resolver was identified, which might allow an unauthenticated remote attacker to cause 1-byte memory overwrite by using a specially crafted DNS response, resulting in worker process crash or, potentially, in arbitrary code execution.
Note that Nessus has not tested for this issue but has instead relied only on the application's self-reported version number.
Solution
Upgrade to nginx 1.20.1 or later.
Plugin Details
Severity: High
ID: 150154
Version: 1.9
Type: combined
Family: Web Servers
Published: June 3, 2021
Modified: September 15, 2022
VPR Key Drivers


kali-linux-2024.3-vmware...
Kali Linux
Download Tenable Ness
Nessus Essentials / Folde
Temp Mail - Disposable
Logins & Passwords
https://localhost:8834/#/scans/reports/13/vulnerabilities/group/150154/127907
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
tenable Nessus Essentials
Scans
Settings
iamprinceefe
My Scans
All Scans
Trash
RESOURCES
Policies
Plugin Rules
Terrascan
Tenable News
CVE-2024-47575: Frequently Asked Questions About F...
Read More
Web Application Test / Plugin #127907
Back to Vulnerability Group
Configure
Audit Trail
Launch
Report
Export
Hosts 1
Vulnerabilities 11
Remediations 1
Notes 1
History 1
HIGH
nginx 1.9.5 < 1.16.1 / 1.17.x < 1.17.3 Multiple Vulnerabilities
Description
According to its Server response header, the installed version of nginx is 1.9.5 prior to 1.16.1 or 1.17.x prior to 1.17.3. It is, therefore, affected by multiple denial of service vulnerabilities :
- A denial of service vulnerability exists in the HTTP/2 protocol stack due to improper handling of exceptional conditions. An unauthenticated, remote attacker can exploit this, by manipulating the window size and stream priority of a large data request, to cause a denial of service condition. (CVE-2019-9511)
- A denial of service vulnerability exists in the HTTP/2 protocol stack due to improper handling of exceptional conditions. An unauthenticated, remote attacker can exploit this, by creating multiple request streams and continually shuffling the priority of the streams, to cause a denial of service condition. (CVE-2019-9513)
Plugin Details
Severity: High
ID: 127907
Version: 1.13
Type: combined
Family: Web Servers
Published: August 16, 2019
Modified: May 2, 2024
VPR Key Drivers

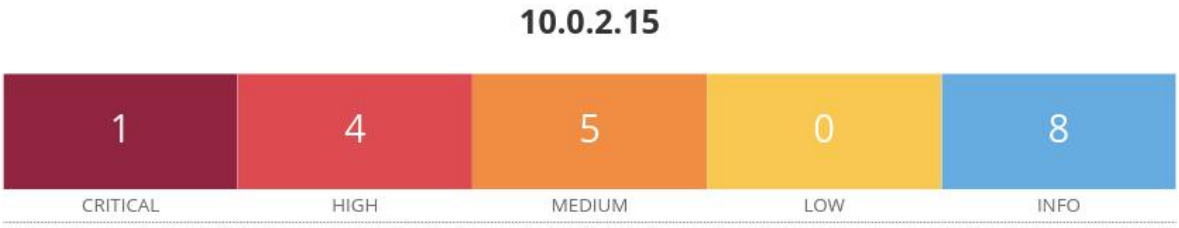

kali-linux-2024.3-vmware...
https://localhost:8834/#/scans/reports/13/vulnerabilities/group/150154/118956
Web Application Test / Plugin #118956
Back to Vulnerability Group
MEDIUM
nginx 1.x < 1.14.1 / 1.15.x < 1.15.6 Multiple Vulnerabilities
Description
According to its Server response header, the installed version of nginx is 1.x prior to 1.14.1 or 1.15.x prior to 1.15.6. It is, therefore, affected by the following issues :
- An unspecified error exists related to the module 'ngx_http_v2_module' that allows excessive memory usage. (CVE-2018-16843)
- An unspecified error exists related to the module 'ngx_http_v2_module' that allows excessive CPU usage. (CVE-2018-16844)
- An unspecified error exists related to the module 'ngx_http_mp4_module' that allows worker process crashes or memory disclosure. (CVE-2018-16845)
Plugin Details
Severity: Medium
ID: 118956
Version: 1.16
Type: combined
Family: Web Servers
Published: November 14, 2018
Modified: April 11, 2022
VPR Key Drivers
Tenable News
Ada.cx SSRF via Sentry Misconfiguration
Read More


kali-linux-2024.3-vmware...
https://localhost:8834/#/scans/reports/13/vulnerabilities/group/150154/134220
Web Application Test / Plugin #134220
Back to Vulnerability Group
MEDIUM
nginx < 1.17.7 Information Disclosure
Description
According to its Server response header, the installed version of nginx is prior to 1.17.7. It is, therefore, affected by an information disclosure vulnerability.
Solution
Upgrade to nginx version 1.17.7 or later.
See Also
http://www.nessus.org/u?fd026623
Plugin Details
Severity: Medium
ID: 134220
Version: 1.9
Type: combined
Family: Web Servers
Published: March 5, 2020
Modified: March 25, 2024
VPR Key Drivers
Tenable News
Flowise Stored Cross-Site Scripting
Read More



6.1: CVE RESEARCH AND ANALYSIS

CREDENTIALS SCAN:
Here's an analysis of each vulnerability and recommendations for addressing them. These vulnerabilities are listed with their criticality and potential impact, focusing on mitigation measures you can implement.



Critical Vulnerability:

1. Node.js (CVE-2024): Multiple Vulnerabilities
Severity: Critical (CVSS 9.8)

Description: This critical vulnerability in Node.js could allow attackers to exploit the application remotely, potentially leading to a system compromise.

Recommendation: Update Node.js to at least version 18.20.4, 20.15.1, or 22.4.1, depending on the version in use. Regularly monitor for security patches for Node.js to stay up-to-date.

High Vulnerabilities:

1. Node.js: Multiple Vulnerabilities
Severity: High (CVSS 8.2 and 7.9)

Description: These vulnerabilities impact earlier versions of Node.js, potentially allowing remote code execution or causing service disruptions.

Recommendation: Ensure Node.js is updated to at least 18.20.1, 20.12.1, or 21.7.2 as per the April security release.

2. nginx 1-Byte Memory Overwrite RCE

Severity: High (CVSS 7.7)
Description: This vulnerability allows remote code execution, potentially letting an attacker execute arbitrary code.
Recommendation: Upgrade nginx to version 1.20.1 or higher, and confirm that your web server environment follows the best practices for server hardening.

3.nginx Multiple Vulnerabilities
Severity: High (CVSS 7.5)

Description: Multiple vulnerabilities affecting nginx versions 1.9.5 to 1.16.1 and 1.17.x to 1.17.3, including the potential for information disclosure.

Recommendation: Update nginx to the latest stable version. Additionally, review nginx configurations to ensure secure configurations.

Medium Vulnerabilities:

1. SSL Certificate Cannot Be Trusted

Severity: Medium (CVSS 6.5)

Description: This indicates that the SSL certificate in use might not be signed by a trusted authority, which could lead to user mistrust or man-in-the-middle attacks.

Recommendation: Use a trusted, valid SSL certificate from a well-known Certificate Authority (CA). Ensure that the certificate is correctly installed and up-to-date.

2. aioHTTP XSS (Cross-Site Scripting)

Severity: Medium (CVSS 6.1)

Description: This vulnerability in aioHTTP allows cross-site scripting, which could lead to data theft or session hijacking.

Recommendation: Update aioHTTP to version 3.9.4 or higher. If aioHTTP is used for web application development, implement input validation to prevent XSS.

3. nginx Information Disclosure

Severity: Medium (CVSS 5.3 and 6.1)

Description: This set of vulnerabilities exposes nginx information that could help an attacker gather server details.

Recommendation: Upgrade nginx to version 1.17.7 or higher, and disable any unnecessary information disclosures within nginx.

4. Apache mod_status Information Disclosure

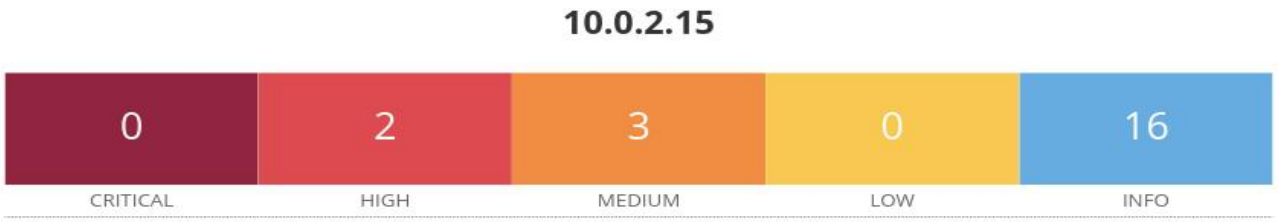
Severity: Medium (CVSS 5.3)

Description: The mod_status module in Apache can expose sensitive server information, which could aid attackers in crafting targeted attacks.

Recommendation: Disable mod_status or restrict access to /server-status to only trusted IP addresses. Review the Apache configuration to limit exposure of sensitive server information.

6.2: WEB APPLICATION VULNERABILITY SCAN

Here's a breakdown of the vulnerabilities discovered, along with recommendations to address them.



1. High Vulnerabilities

1. **nginx 1-Byte Memory Overwrite RCE**

Severity: High (CVSS 7.7)

Description: This vulnerability allows for remote code execution (RCE) due to a 1-byte memory overwrite. An attacker could exploit this to execute arbitrary code on the web server.

Recommendation: Update nginx to at least version 1.20.1 or the latest stable release. Regularly monitor for security patches and updates for nginx to prevent similar vulnerabilities.

2. **nginx Multiple Vulnerabilities**

Severity: High (CVSS 7.5)

Description: This set of vulnerabilities affects nginx versions from 1.9.5 to 1.16.1 and 1.17.x to 1.17.3. It includes various issues that could be exploited to compromise the web application.

Recommendation: Upgrade nginx to a more recent stable version that patches these vulnerabilities. Carefully review your nginx configuration settings to ensure secure web server operation.

2. Medium Vulnerabilities

1. nginx Multiple Vulnerabilities

Severity: Medium (CVSS 6.1)

Description: This includes vulnerabilities affecting nginx versions prior to 1.14.1 and 1.15.6. These vulnerabilities could lead to security weaknesses and compromise web application security.

Recommendation: Update nginx to at least version 1.14.1 or higher. Ensure that only necessary modules are enabled and verify that nginx configuration follows best security practices.

Apache mod_status Information Disclosure

Severity: Medium (CVSS 5.3)

Description: The mod_status module in Apache allows anyone with access to /server-status to view real-time server activity, potentially exposing sensitive information.

Recommendation: Disable mod_status in Apache if it's not necessary. If required, restrict access to /server-status to specific IP addresses or internal networks only. Review Apache configuration to limit data exposure.

nginx Information Disclosure

Severity: Medium (CVSS 5.3)

Description: This vulnerability affects nginx versions prior to 1.17.7 and allows attackers to gather potentially sensitive information from the server.

Recommendation: Update nginx to version 1.17.7 or a newer stable release. Additionally, review and harden nginx configurations to prevent unnecessary

information disclosure.

6.2: RECOMMENDATIONS
To strengthen the company's security posture, consider the following steps:

1. **Upgrade Software**: Update Node.js, nginx, and aioHTTP and Apache to the latest stable versions to resolve known vulnerabilities.

2. **Review Web Server Configuration**: Disable mod_status in Apache if not required, or limit its access. Configure nginx to minimize information disclosure and restrict access to server details.

3. **Regularly Monitor for Updates**: Keep track of newly disclosed vulnerabilities (CVEs) for nginx and Apache, applying patches promptly.

4. **Conduct Regular Scans**: Periodic vulnerability assessments can help identify new risks in the web application infrastructure.

5. **Secure Web Server Configurations**: Configure nginx and Apache to prevent information leaks and restrict access to potentially sensitive server status data.

6.Implement Strong SSL/TLS: Obtain and configure a trusted SSL certificate for secure communication.
Addressing these vulnerabilities will reduce the likelihood of exploitation and improve the resilience of your web application against potential attacks.

7.0: Conclusion:

In conclusion, the team has successfully identified and addressed several key security vulnerabilities across different systems and software packages. Through credentialed scans, web application assessments, and the configuration of SMTP for email reporting, we ensured that the systems were properly audited and that vulnerabilities were flagged for remediation. We also took proactive measures to mitigate risks associated with outdated software versions, including upgrading Node.js, Django, and OpenJDK to their latest secure versions, in accordance with the respective security advisories. These efforts are a crucial part of maintaining the integrity, security, and compliance of our infrastructure, ensuring that our systems are protected against known threats and vulnerabilities.