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	MegaCorpOne LLC
Contact Name	Muhammad Hussain
Contact Title	Lead Penetration Tester

Document History

Version	Date	Author(s)	Comments
001	08/14/2024	Muhammad Hussain	

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.

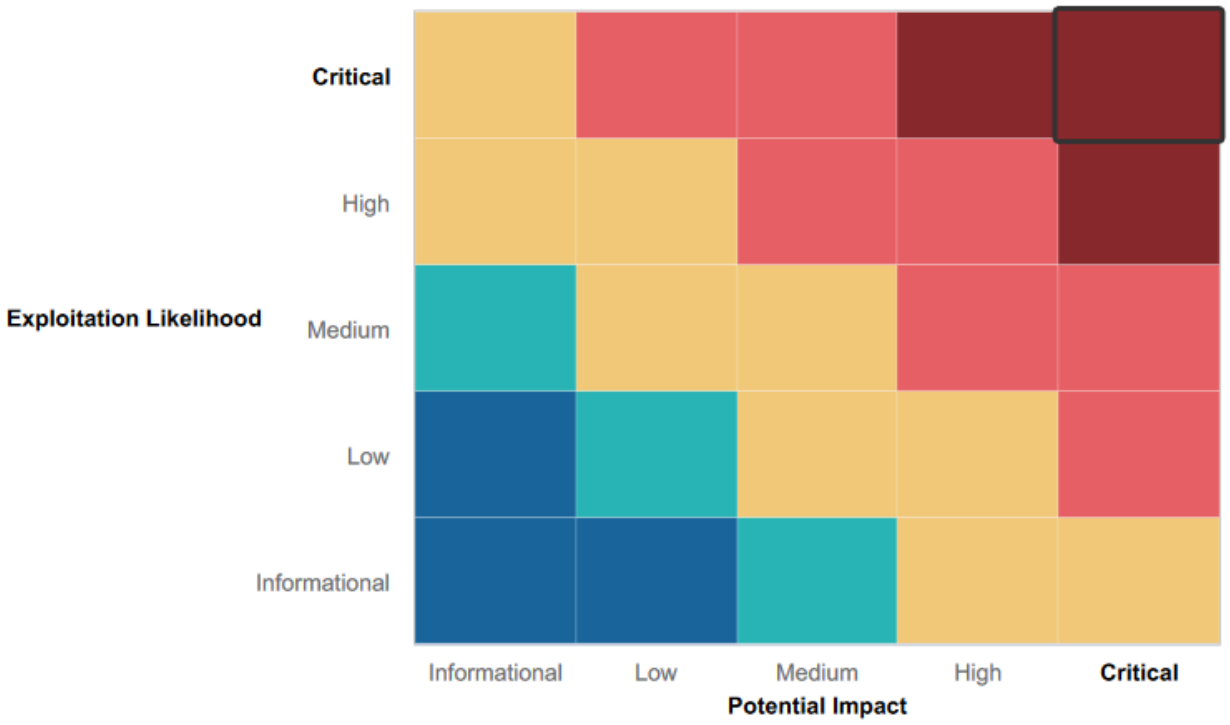
Executive Summary of Findings

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.

- Mitigation strategy in place for denial of DDOS Attacks to ensure network availability
- No vulnerable open source data penetration due to mapping network architecture
- Tools like Metasploit/John the Ripper/Nmap are utilized to prevent unauthorized access
- Forward-thinking defensive and offensive strategy
- Current and continuing penetration testing to identify vulnerabilities for mitigation

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.

- XSS vulnerabilities
- Sensitive data exposure
- Local file inclusion
- SQL Injection
- Command Injection
- Brute Force Attacks
- PHP Injection
- Directory traversal
- Shellshock

Executive Summary

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.

Each threat is assessed in terms of both its potential impact on the business and the likelihood of exploitation.

Summary Vulnerability Overview

Vulnerability	Severity
XSS Reflected Vulnerability	High
XSS Reflected (Advanced) Vulnerability	High
XSS Stored Vulnerability	High
Sensitive Data Exposure Vulnerability	Low
Local File Inclusion Vulnerability	High
Local File Inclusion (Advanced) Vulnerability	Medium
SQL Injection Vulnerability	Critical
Sensitive Data Exposure Vulnerability	Critical
Sensitive Data Exposure Vulnerability	High
Command Injection Vulnerability	Critical
Command Injection (Advanced) Vulnerability	High
Brute Force Attack Vulnerability	Critical
PHP Injection Vulnerability	Medium
Session Management Vulnerability	High
Directory Traversal Vulnerability	Critical
Open Source Exposed Data	Low
Pinging totalrekall.xyz	Low
Open Source Exposed Data	Low
Number of Hosts on the Network	Medium
Scan Results	High
Nessus Scan Results	Critical
Apache Tomcat Remote Code Execute Vulnerability (CVE-2017-126-17)	Critical
Shellshock	High
Other Vulnerabilities on the affected Host	Critical
Struts – CVE-2017-5638	High
Drupal – CVE-2019-6340	High
CVE-2019-14287	High

Tanya4life	Low
Nmap Scan	Medium
FTP Anonymous	Medium
SLMail (SMTP on P25 and POP3 on P110)	Medium
Scheduled Task Vulnerability	Medium
Kiwi Flag 6	Critical
Lateral Movement	Critical
MsCacheV2 vis LSADump	Critical
Navigating the C:\ directory	Critical
Default Administrator Credentials	High

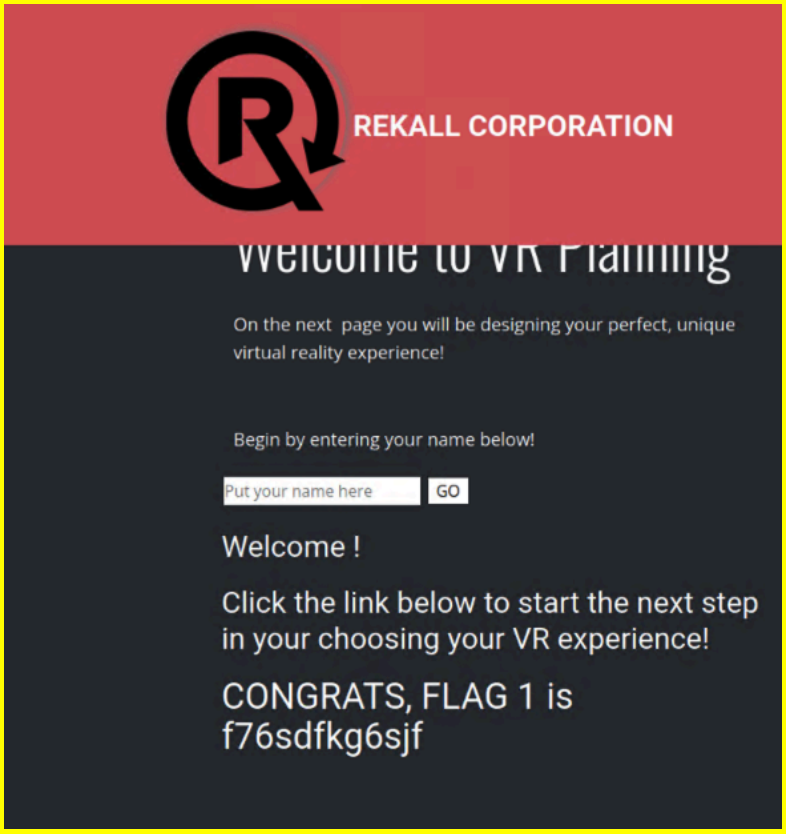
The following summary tables represent an overview of the assessment findings for this penetration test:

Scan Type	Total
Hosts	Web App 34.102.136.180 Linux OS 192.168.13.10 192.168.13.11 192.168.13.12 192.168.13.13 192.168.13.14 Windows OS 172.22.117.10 172.22.117.20
Ports	21-FTP 25-SMTP 80-HTTP 106-POP3PW 110-POP3 135-MSRPC 139-NETBIOS-SSN 443-SSL/HTTP

Exploitation Risk	Total
Critical	12
High	13
Medium	7

Low	6

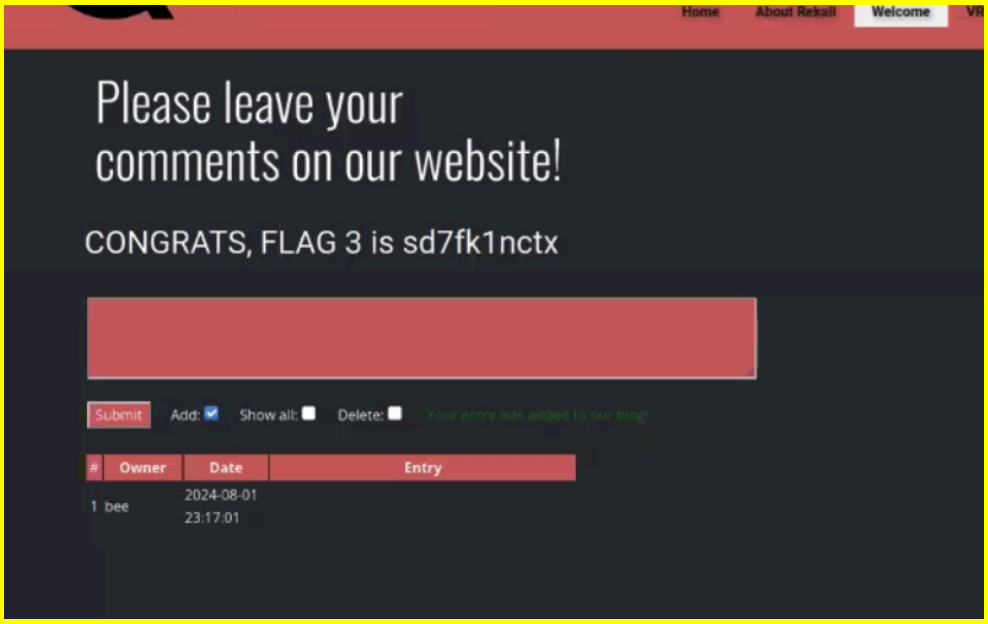
Vulnerability Findings

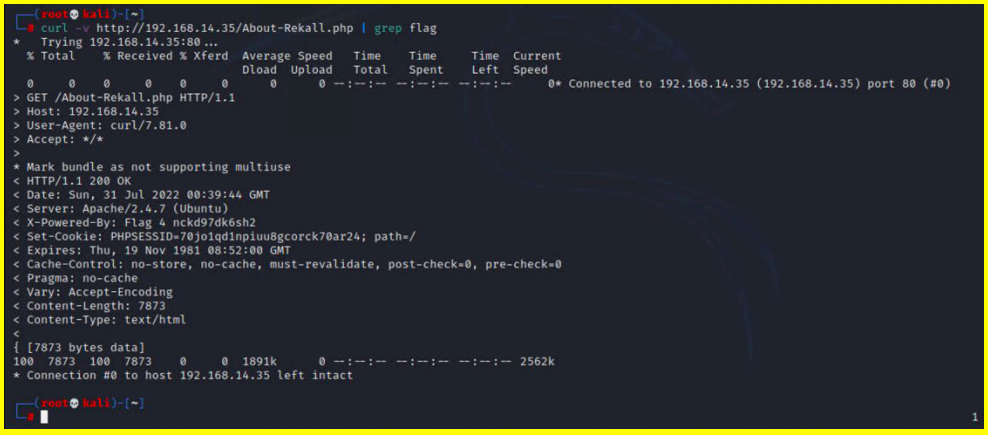
Vulnerability 1	Findings
Title	XSS Reflected Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	Vulnerability was located in the “Welcome.php” page of the Rekal Corp. website. I used “<script>alert(‘1’)</script>” to reveal the vulnerability
Images	
Affected Hosts	Welcome.php
Remediation	The implementation of better security awareness training, the utilization of output encoding libraries, and implementing the practice of always sanitizing and validating user data.

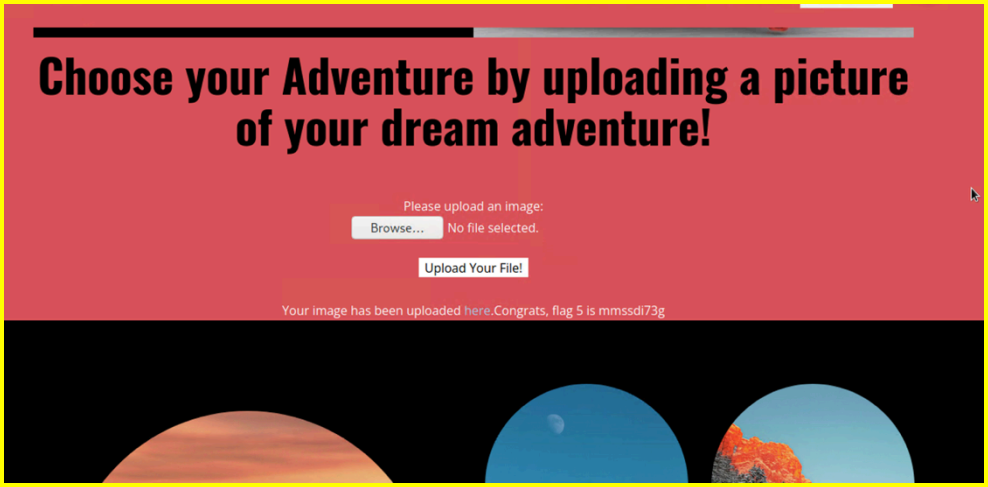
Vulnerability 2	Findings

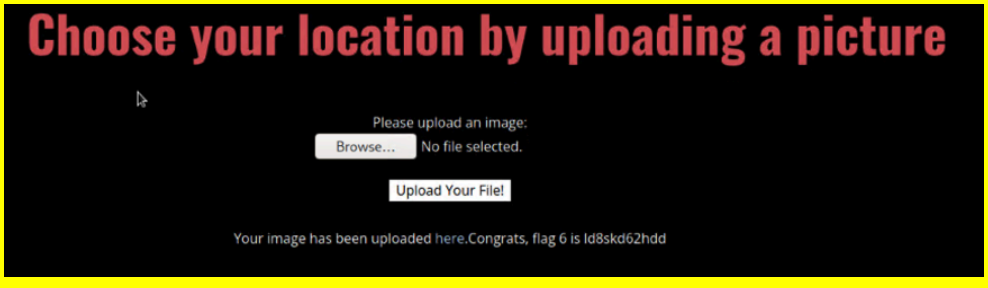
Title	XSS Reflected (Advanced) Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	Vulnerability was located on the “Who do you want to be?” Memory Planner page of the Rekall Corp website. I used <SCRscriptIPT>alert(‘Hello’)</SCRscriptIPT> to reveal the vulnerability due to the input validation removing the word “script”.
Images	
Affected Hosts	Memory-Planner.php
Remediation	The implementation of better security awareness training, the utilization of output encoding libraries, and implementing the practice of always sanitizing and validating user data.

Vulnerability 3	**Findings**
Title	XSS Stored Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	Vulnerability was located on the comments.php page of the Rekall Corp. website. I used <script>alert(‘1’)</script> to find the stored user data. The data is stored on the server and displayed to users without proper encoding or sanitization.

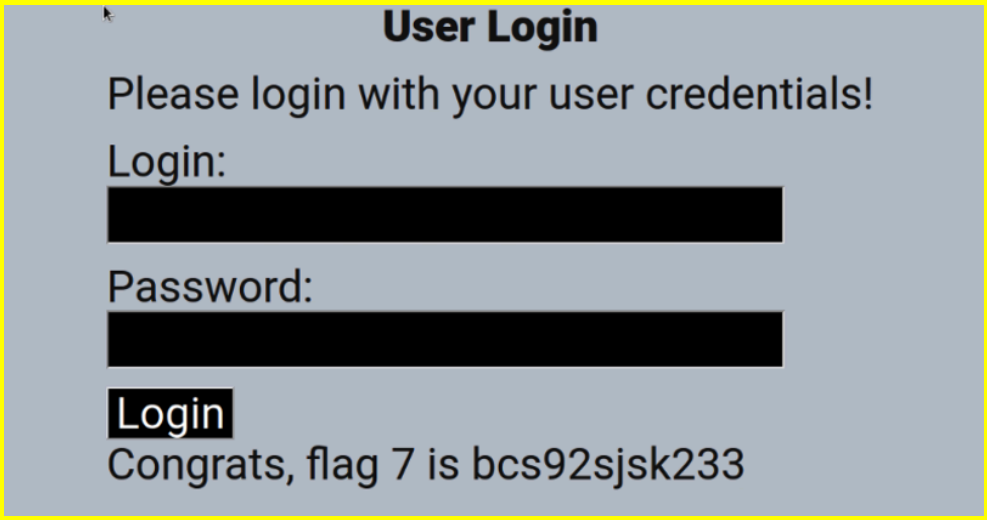
Images	
Affected Hosts	Comments.php
Remediation	The implementation of better security awareness training, the utilization of output encoding libraries, and implementing the practice of always sanitizing and validating user data.

Vulnerability 4	Findings
Title	Sensitive Data Exposure Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Low
Description	Utilizing Curl, the flag is found in the HTTP response header.
Images	
Affected Hosts	About-Rekall.php
Remediation	N/A

Vulnerability 5	Findings
Title	Local File Inclusion Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	Vulnerability is located in Memory-Planner.php and is revealed by uploading any php file.
Images	
Affected Hosts	Memory-Planner.php
Remediation	Validating user inputs and

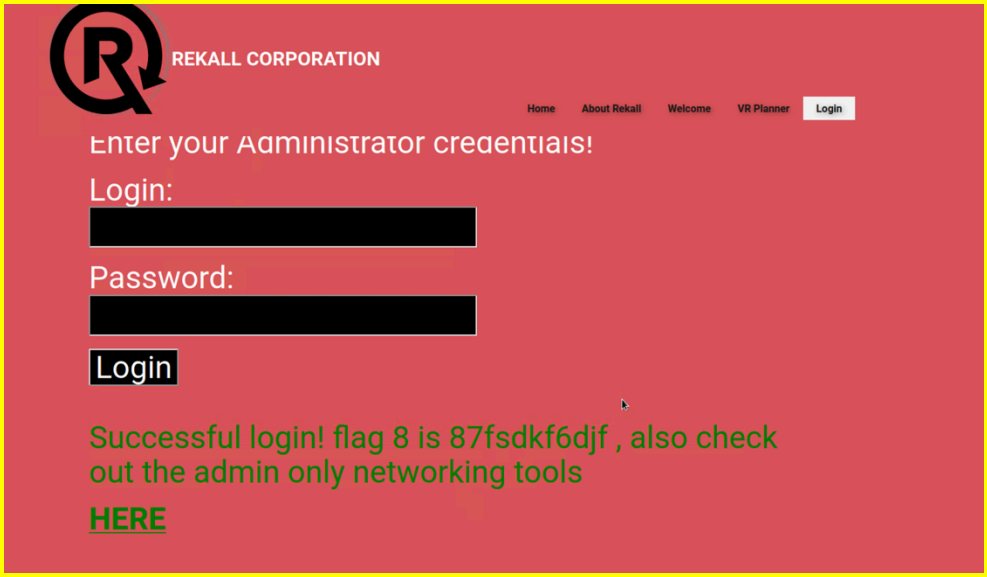
Vulnerability 6	Findings
Title	Local File Inclusion (Advanced) Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Medium
Description	Vulnerability is located in Memory-Planner.php and filters for .jpg. To bypass this, I renamed my script to script.jpg.php.
Images	

Affected Hosts	Memory-Planner.php
Remediation	Strong input validation, limiting file types for uploads, and restricting file paths to prevent execution of unintended files.

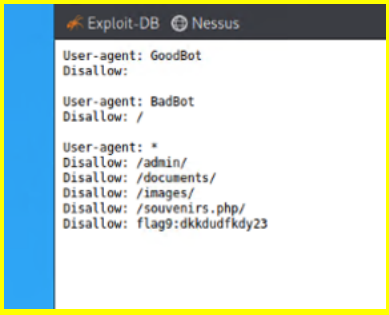
Vulnerability 7	**Findings**
Title	SQL Injection Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	Vulnerability is located in the “Admin Login” on the Login.php page. I used “1’ OR ‘1’ = ‘1” in the login field
Images	
Affected Hosts	Login.php
Remediation	Implement input validation and sanitization of all input prior to it being processed on the backend.

Vulnerability 8	**Findings**
Title	Sensitive Data Exposure Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	Username and Password are in the HTML
Images	

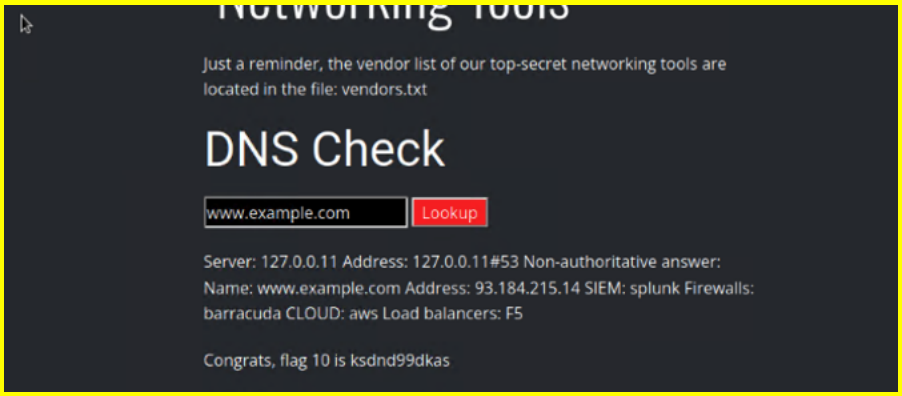
```
    <form action="/Login.php" method="POST">

        <p><label for="login">Login:</label><font color="#DB545A">dougquaid</font><br />
        <input type="text" id="login" name="login" size="20" /></p>

        <p><label for="password">Password:</label><font color="#DB545A">kuato</font><br />
        <input type="password" id="password" name="password" size="20" /></p>

        <button type="submit" name="form" value="submit" background-color="black">Login</button>

    </form>

    </br >

</div>
```



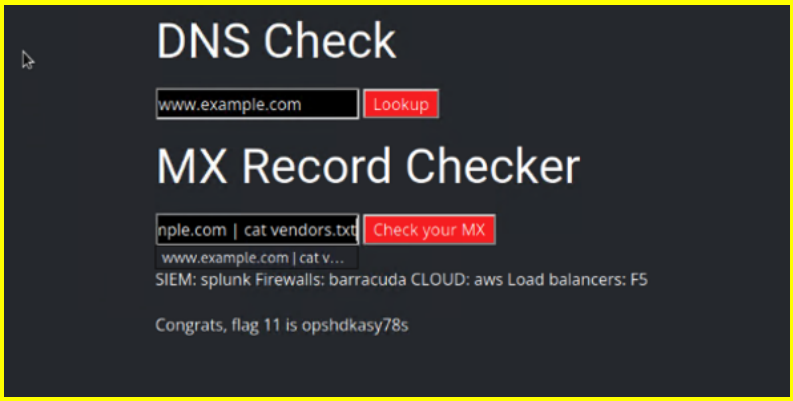
Affected Hosts	Login.php
Remediation	Not storing sensitive information in HTML or publicly accessible and encrypting sensitive data

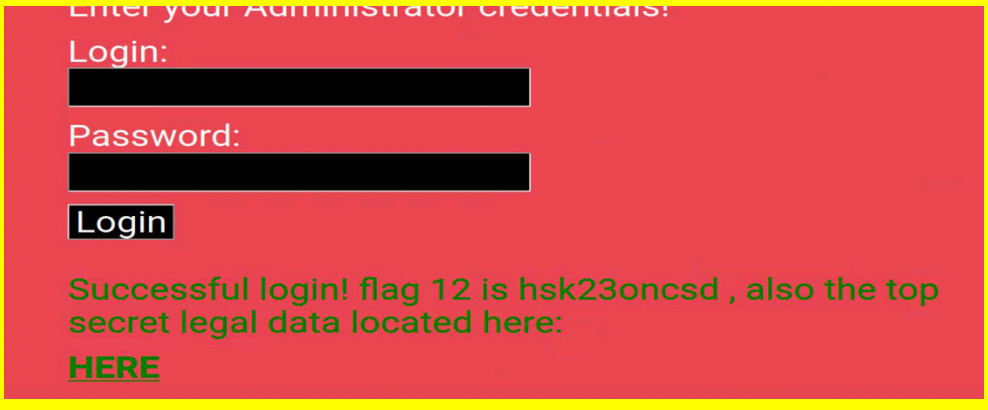
Vulnerability 9	Findings
Title	Sensitive Data Exposure Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	Data exposure is available on the robots.txt page
Images	

Affected Hosts	Robots.txt
Remediation	Double checking data that is entered

Vulnerability 10	Findings
Title	Command Injection Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	Vulnerability is on the networking.php page in the DNS Check field. I used "www.exammple.com; cat vendors.txt"
Images	
Affected Hosts	Networking.php
Remediation	Implementing input validation and strong access controls

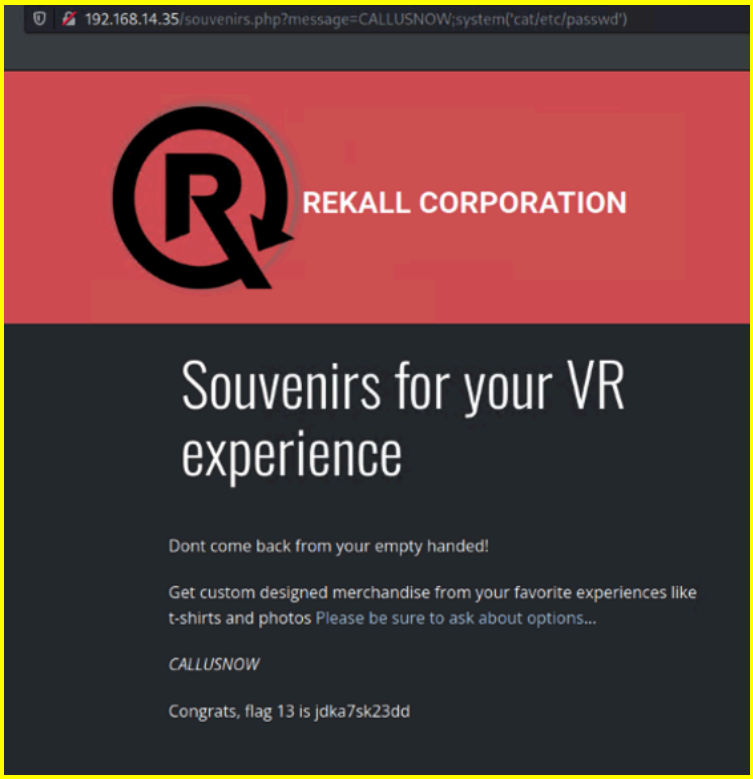
Vulnerability 11	Findings
Title	Command Injection (Advanced) Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	Vulnerability is on the networking.php page in the MX Record Checker. Input validation doesn't allow "&" and ";", so I used "www.example.com | cat vendors.txt"

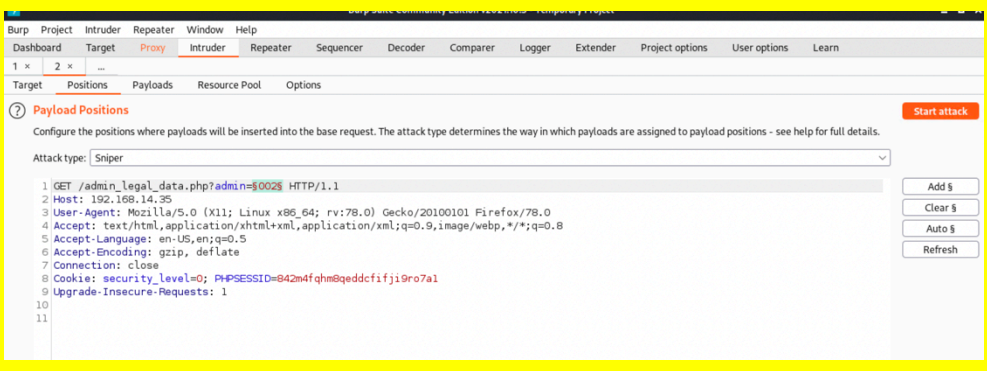
Images	
Affected Hosts	Networking.php
Remediation	Implementing input validation and strong access controls

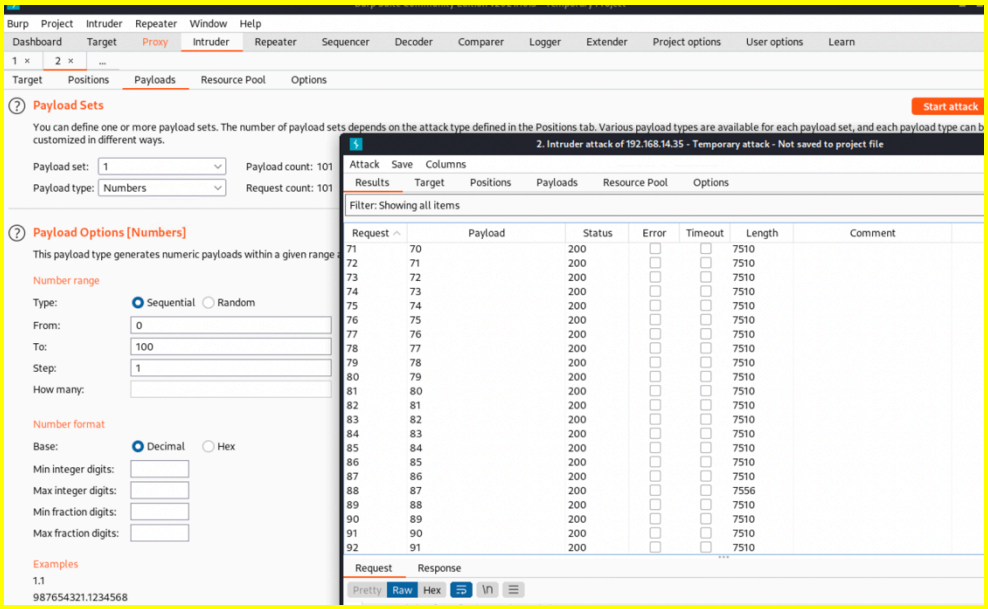
Vulnerability 12	Findings
Title	Brute Force Attack Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	Vulnerability is on the networking.php page, utilizing the same vulnerability in flag 10 and 11, /etc/passwd file can be used to see a login credential (melina:melina)
Images	

	
Affected Hosts	Login.php
Remediation	Implementing account lockouts and complex password requirements

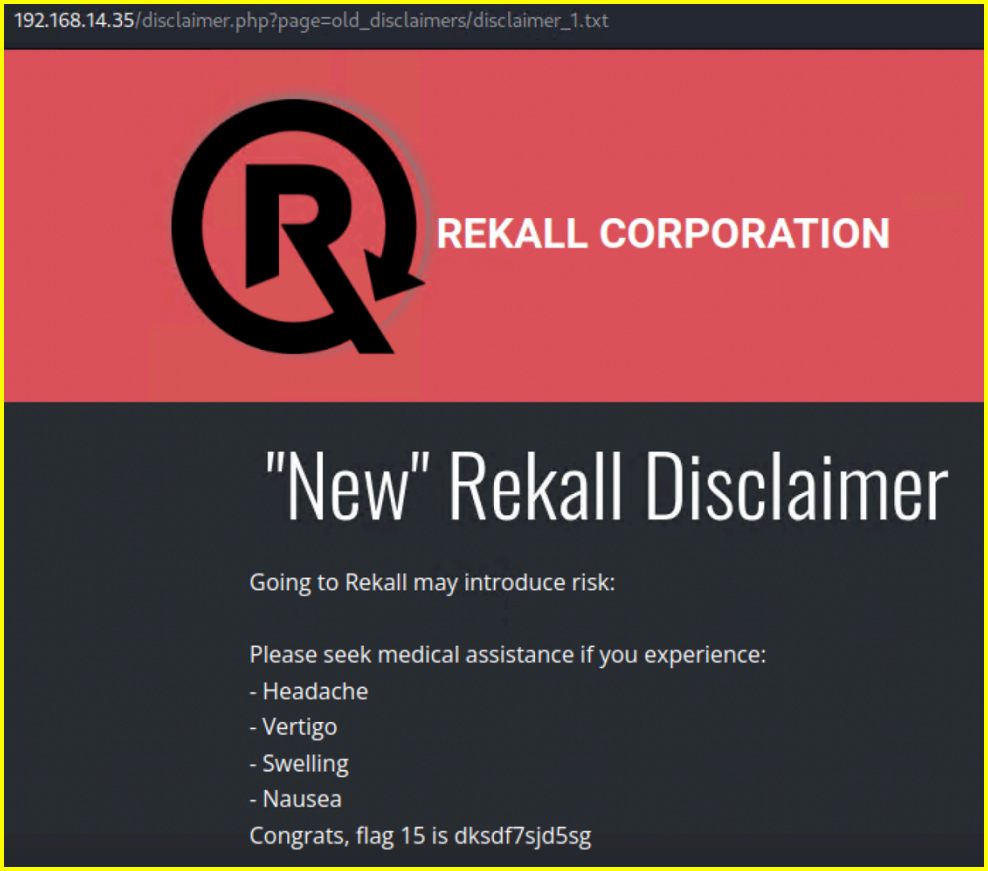
Vulnerability 13	Findings
Title	PHP Injection Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Medium
Description	The souvenirs.php webpage with the vulnerability is shown in the robots.txt file. This page can be accessed by modifying the URL to the following. http://192.168.13.35/souvenirs.php?message=""; system('cat /etc/passwd')

Images	
Affected Hosts	Souvenirs.php
Remediation	Employing validation, filtering and sanitization of user inputs.

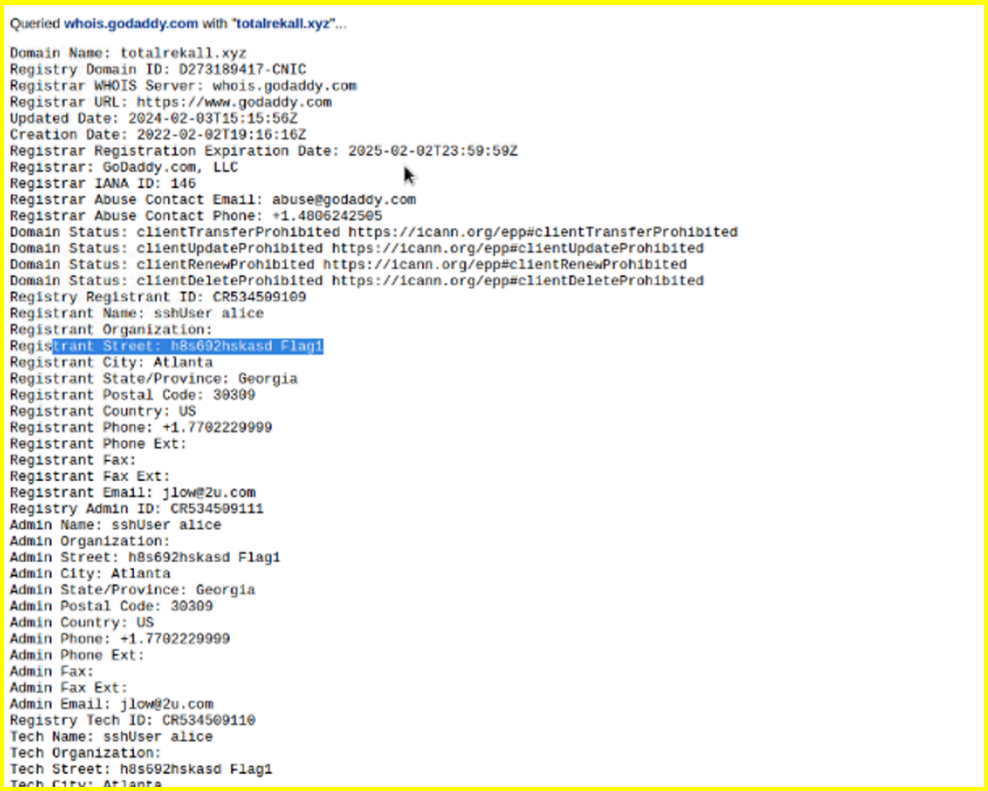
Vulnerability 14	Findings
Title	Session Management Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	The vulnerable page was identified via the Burpsuite intruder to brute force the Session ID. Many were tested but 87 was the secret session ID that led to the flag.
Images	

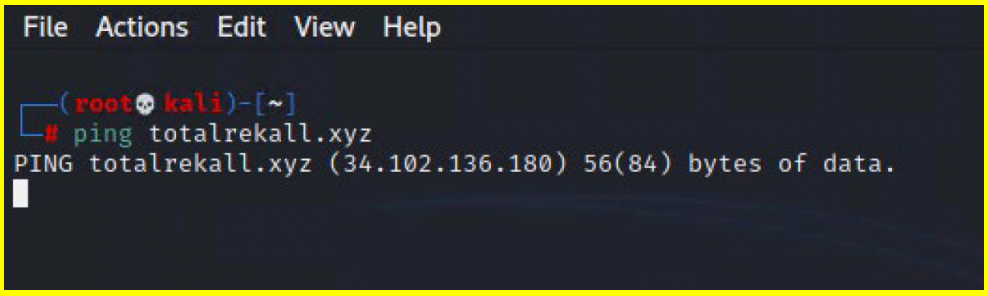
	 
Affected Hosts	Disclaimer.php
Remediation	Utilizing secure and random session ID's with session timeouts. Enforcing SSL for all connections

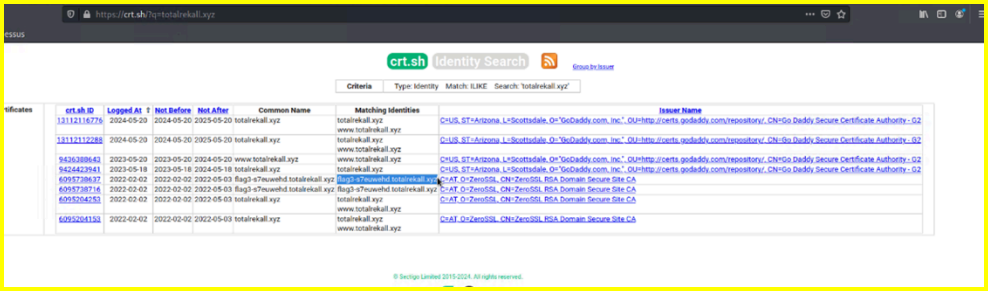
Vulnerability 15	Findings
Title	Directory Traversal Vulnerability
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical

Description	Using the vulnerabilities of Flag 10 and 11, I utilized the ls command to see the directory of old_disclaimers. The URL was modified to http://192.168.13.35/disclaimer.php?page=old_disclaimers/disclaimer_1.txt to find the “New” Rekall Disclaimer
Images	
Affected Hosts	Disclaimer.php
Remediation	Removing older code and pages and performing code cleanup.

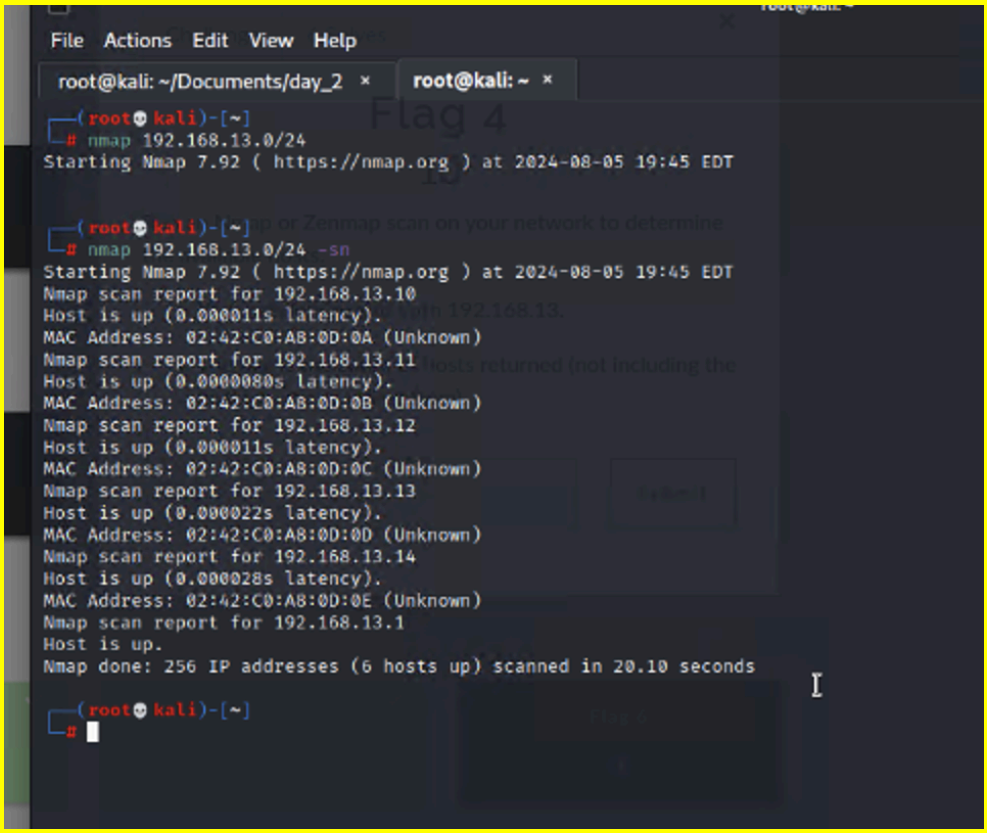
Vulnerability 1	**Findings**
Title	Open Source Exposed Data
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Low
Description	In the WHOIS data for totalrekall.xyz, the registrant street was revealed

Images	
Affected Hosts	https://centralops.net/co/DomainDossier/aspx
Remediation	N/A

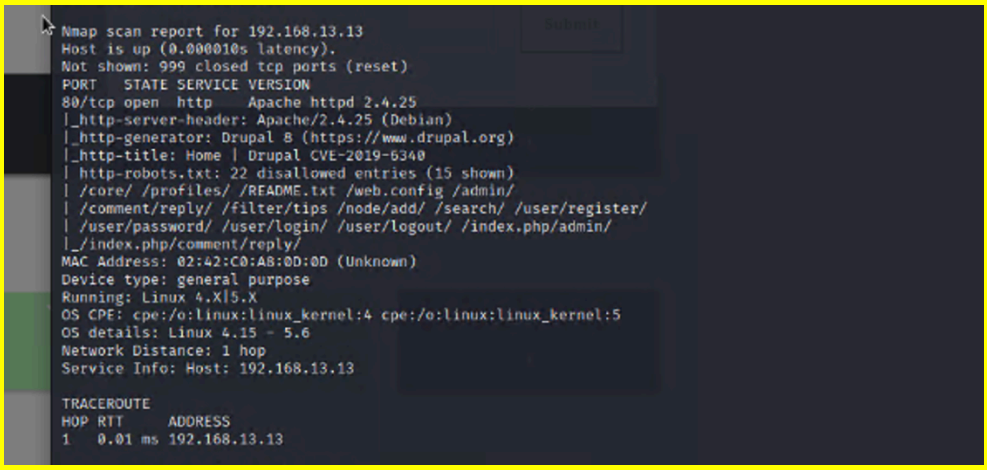
Vulnerability 2	Findings
Title	Pinging totalrekall.xyz
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Low
Description	pinging
Images	
Affected Hosts	34.102.136.180
Remediation	N/A

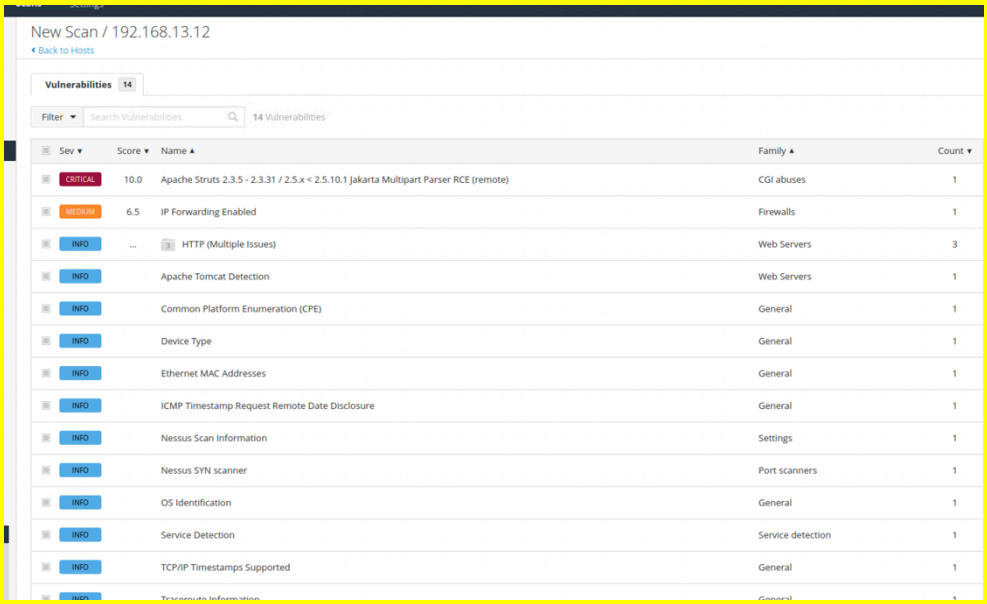
Vulnerability 3	Findings
Title	Open Source Exposed Data
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Low
Description	Using crt.sh, search for totalrekall.xyz
Images	
Affected Hosts	N/A
Remediation	N/A

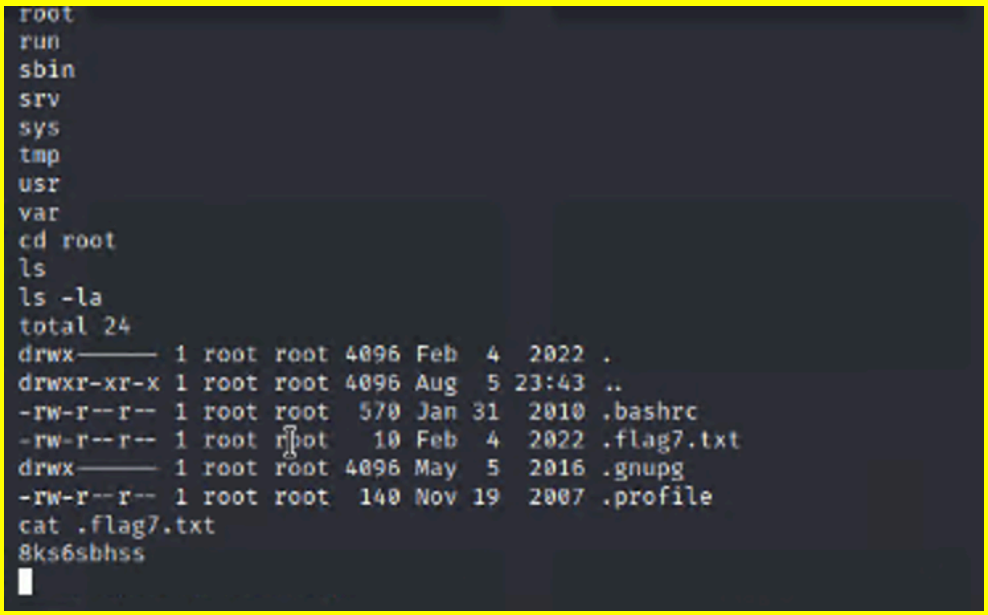
Vulnerability 4	Findings
Title	Number of Hosts on the Network
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Medium
Description	Utilizing Nmap, scan the network and determine the number of hosts, excluding the host. (5)

Images	
Affected Hosts	192.168.13.0/24
Remediation	N/A

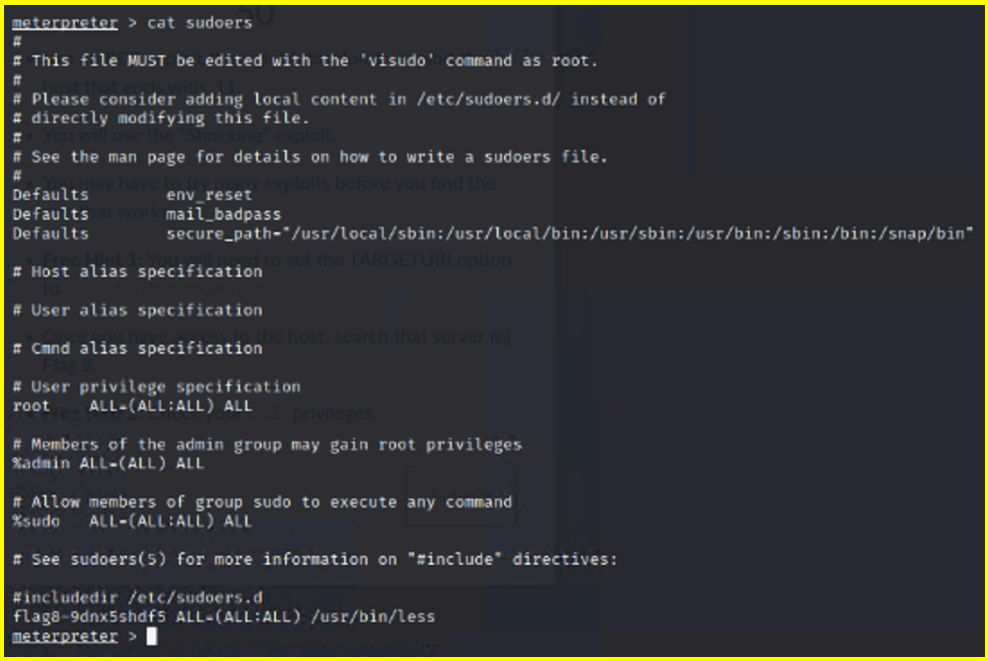
Vulnerability 5	Findings
Title	Scan Results
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	Utilize an aggressive Nmap Scan (nmap 192.168.13.0/24 -A) to find the host running Drupal (192.168.13.13)

Images	
Affected Hosts	192.168.13.13
Remediation	N/A

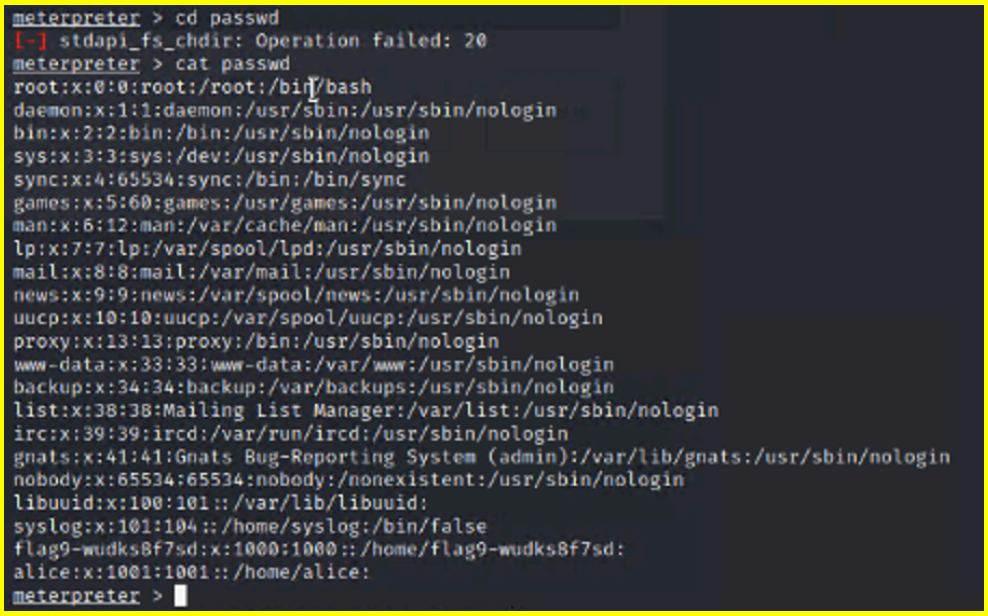
Vulnerability 6	Findings
Title	Nessus Scan Results
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Critical
Description	Utilizing Nessus, scan 192.168.13.12 to find a critical vulnerability for Apache Struts.
Images	
Affected Hosts	192.168.13.12
Remediation	Keeping software up to date with the most recent security updates

Vulnerability 7	Findings
Title	Apache Tomcat Remote Code Execute Vulnerability (CVE-2017-126-17)
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Critical
Description	Utilizing MSFconsole, searched for the correct RCE exploit to apply to Apache Tomcat. Created a shell and navigated to find the flag.
Images	
Affected Hosts	192.168.13.10
Remediation	Keeping software up to date with the most recent security updates

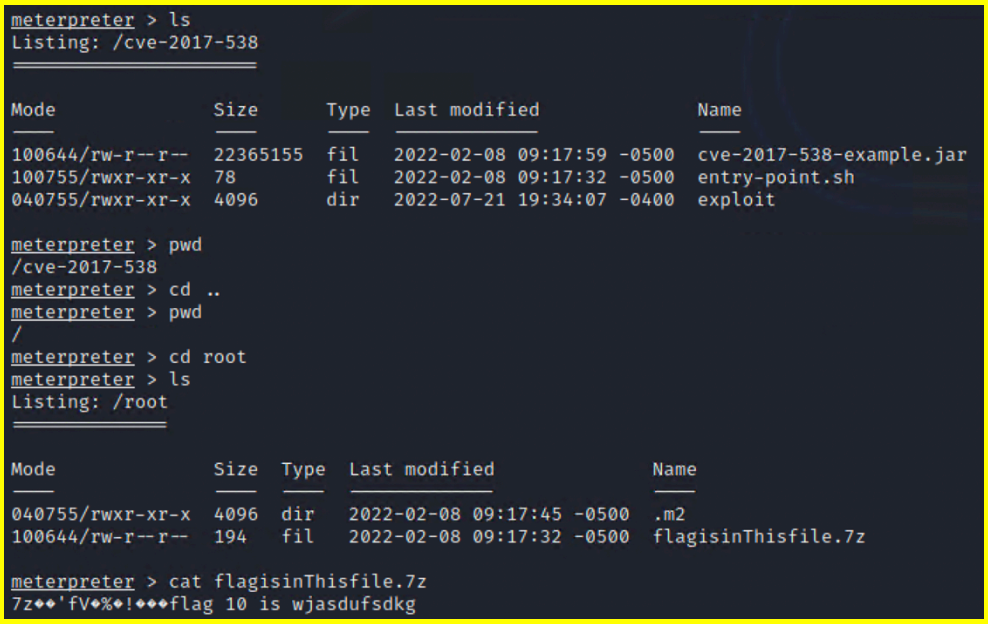
Vulnerability 8	Findings
Title	Shellshock
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	Utilizing MSFconsole, search for the shellshock exploit (exploit/multi/http/apahce_mod_cgi_bash_env_exec). Use the following options - Target URI: /cgi-bin/shockme.cgi - RHOST: 192.168.13.11 The flag was found in the /etc/sudoers folder

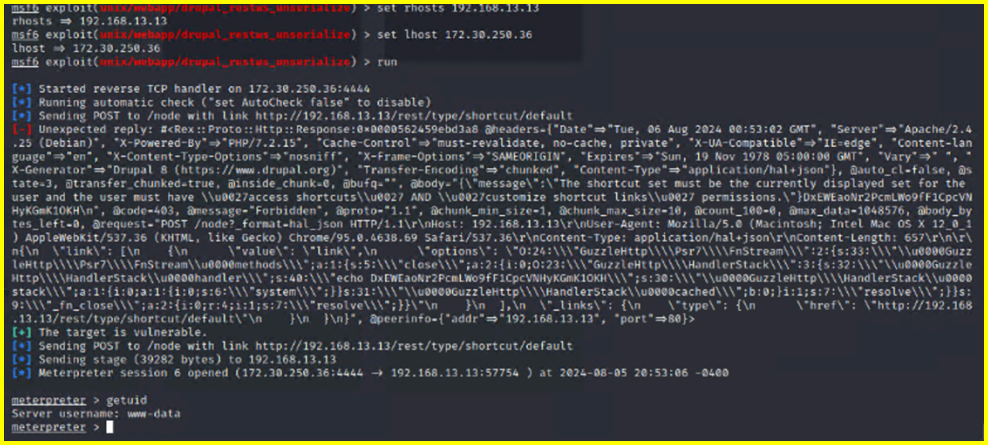
Images	
Affected Hosts	192.168.13.11
Remediation	Keeping software up to date with the most recent security updates

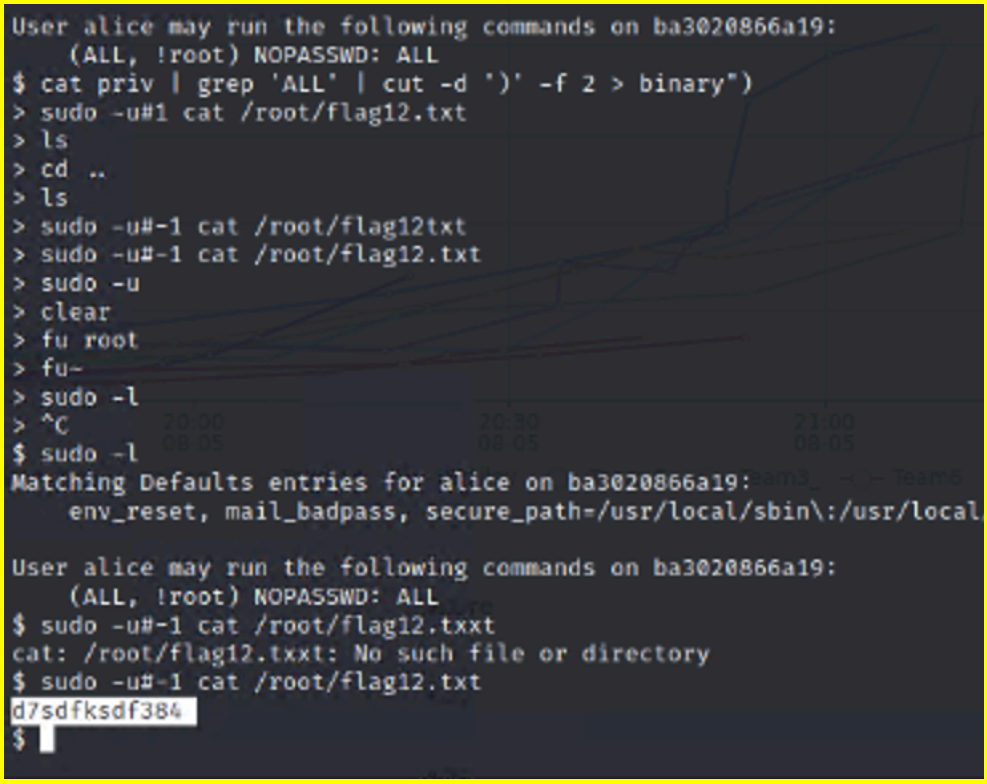
Vulnerability 9	Findings
Title	Other Vulnerabilities on the affected Host
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Critical
Description	In the same host machine, find flag 9 with the following command, ”cat /etc/passwd”

Images	
Affected Hosts	192.168.13.11
Remediation	Keeping software up to date with the most recent security updates

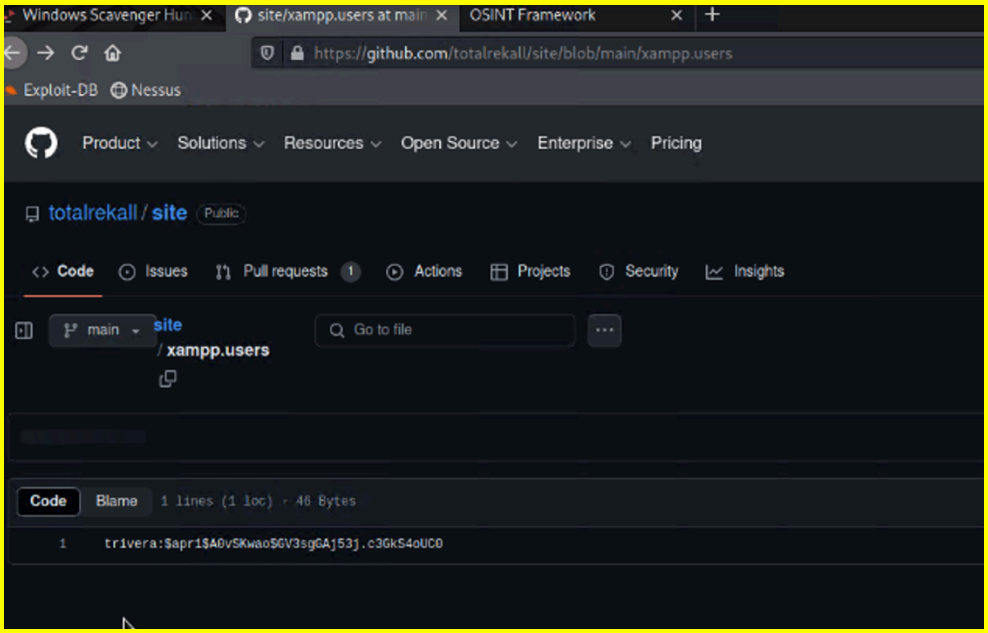
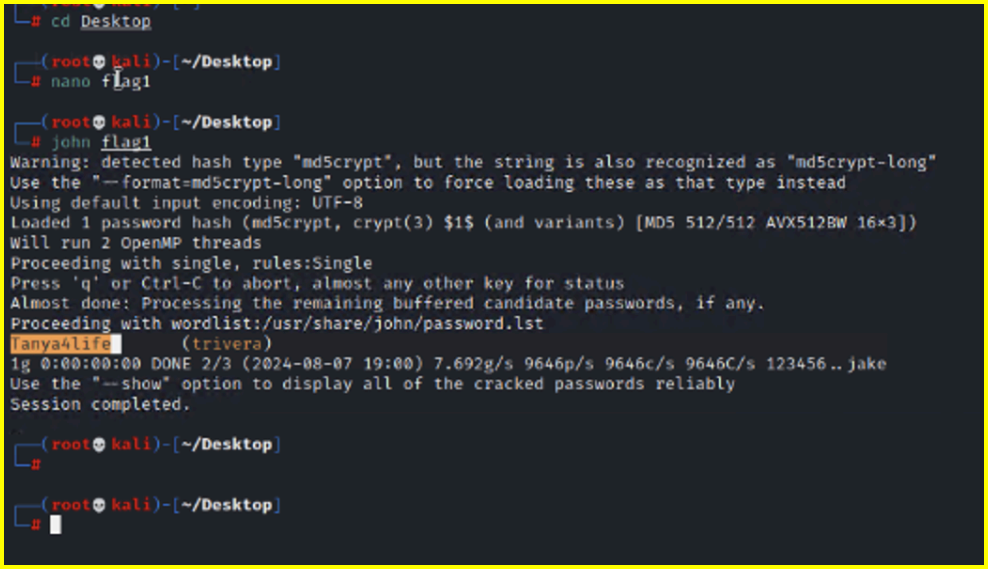
Vulnerability 10	Findings
Title	Struts – CVE-2017-5638
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	Utilizing Nessus, determine that this host (192.168.13.12) is vulnerable to a struts exploit. Via MSFconsole, utilize the struts2_content_type_ognl exploit. Once a shell was made, found flag in the “flagisinThisfile.7z” file.

Images	
Affected Hosts	192.168.13.13
Remediation	Keeping software up to date with the most recent security updates

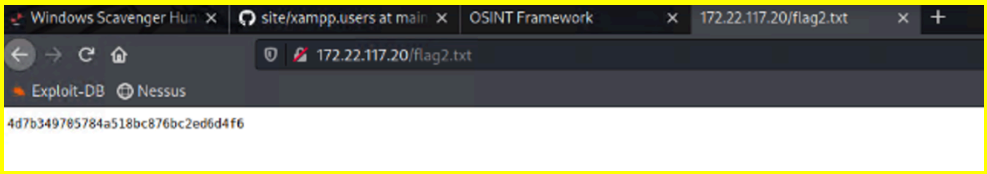
Vulnerability 11	Findings
Title	Drupal – CVE-2019-6340
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	Via Nmap, locate a host with Drupal. Utilizing MSFconsole, use unix/webapp/drupal_restws_unserialize exploit. Set the RHOSTS to 192.168.13.13. When a shell is formed, use “getuid”.
Images	
Affected Hosts	192.168.13.13
Remediation	Keeping software up to date with the most recent security updates

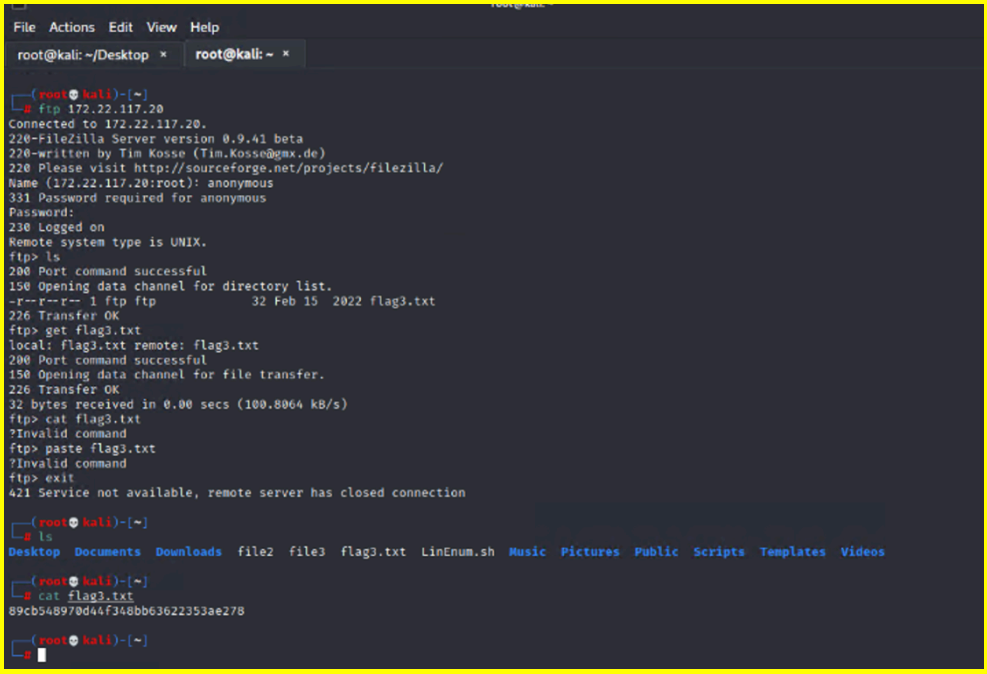
Vulnerability 12	Findings
Title	CVE-2019-14287
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	SSH into the user Alice, perform a privlage escalation. Use command sudo -u#-1 cat /root/flag12.txt to get the flag.
Images	
Affected Hosts	192.168.13.14
Remediation	Keeping software up to date with the most recent security updates

Vulnerability 1	Findings
Title	Tanya4life
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Low
Description	Utilizing GitHub, and the repository “xampp.users”, found username and hashed password. Used John the Ripper to reveal credentials (trivera:Tanya4life)

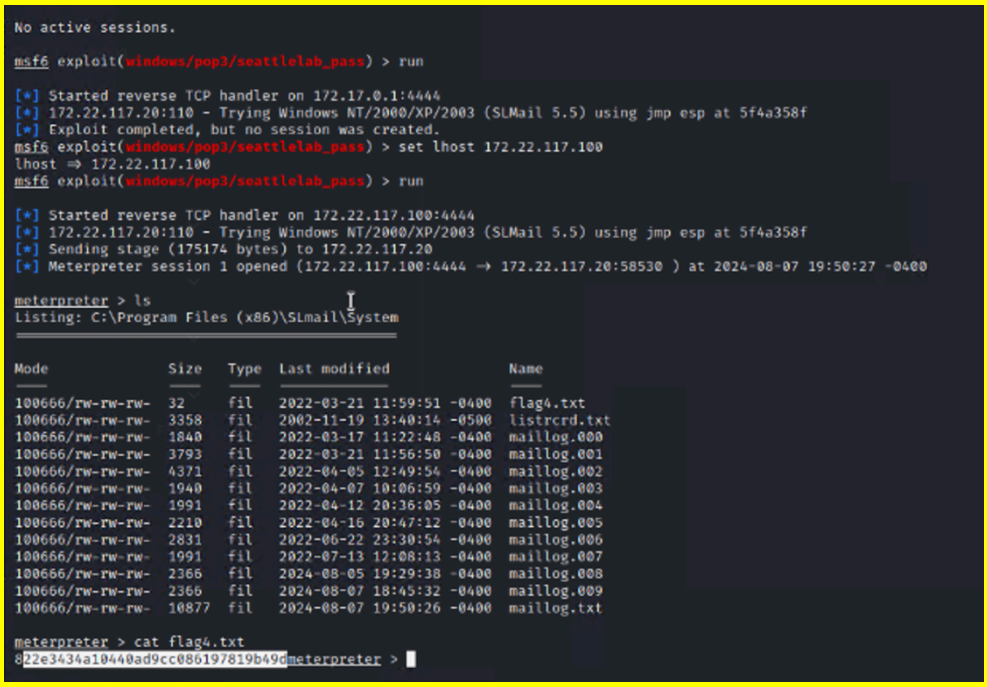
Images	 
Affected Hosts	N/A
Remediation	Not post credentials in an open forum

Vulnerability 2	Findings
Title	Nmap Scan
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Medium
Description	Utilizing Nmap, scan for available hosts. Then key in the IP's in the browsers URL.
Images	

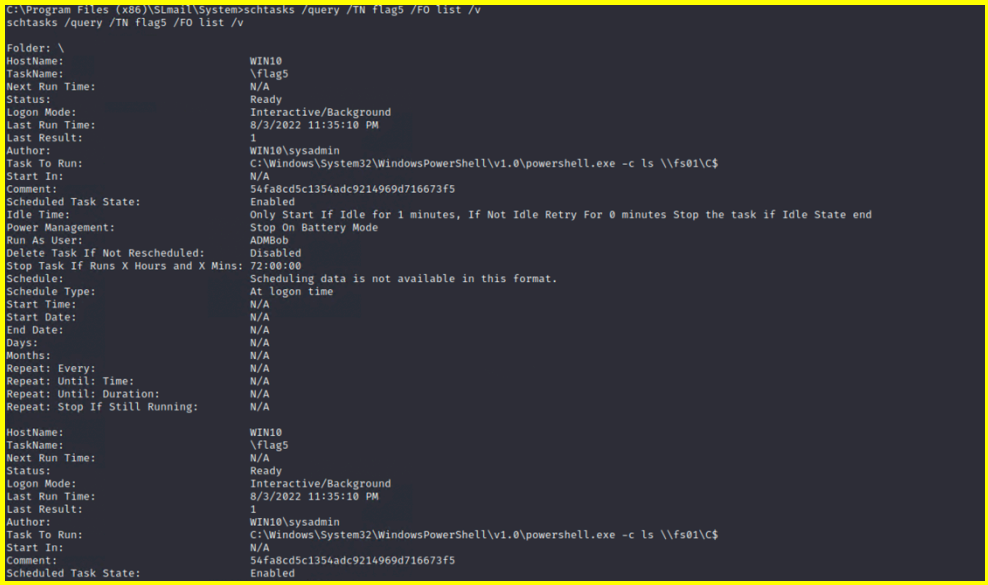
	
Affected Hosts	172.22.117.20
Remediation	N/A

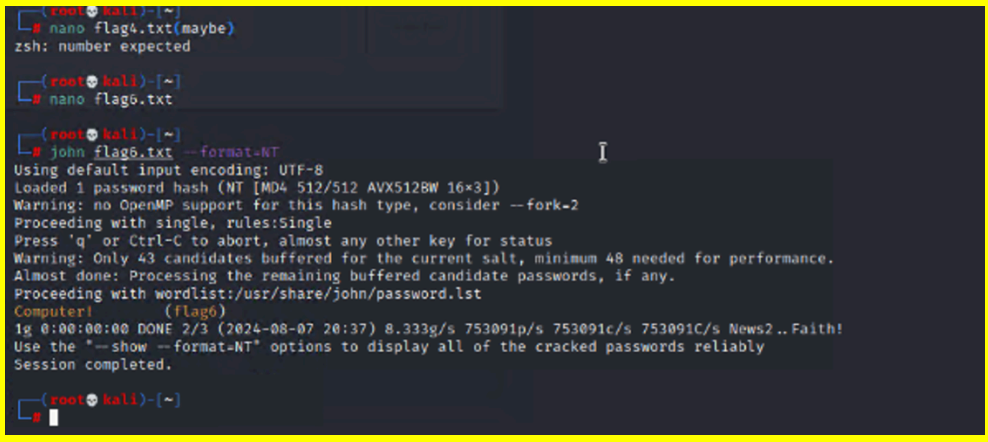
Vulnerability 3	Findings
Title	FTP Anonymous
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Medium
Description	Utilizing Nmap, notice FTP on port 21 is open.
Images	
Affected Hosts	172.22.117.20
Remediation	Unused ports should be closed. Whitelisting can be used to allow authorized users.

Vulnerability 4	Findings
Title	SLMail (SMTP on P25 and POP3 on P110)

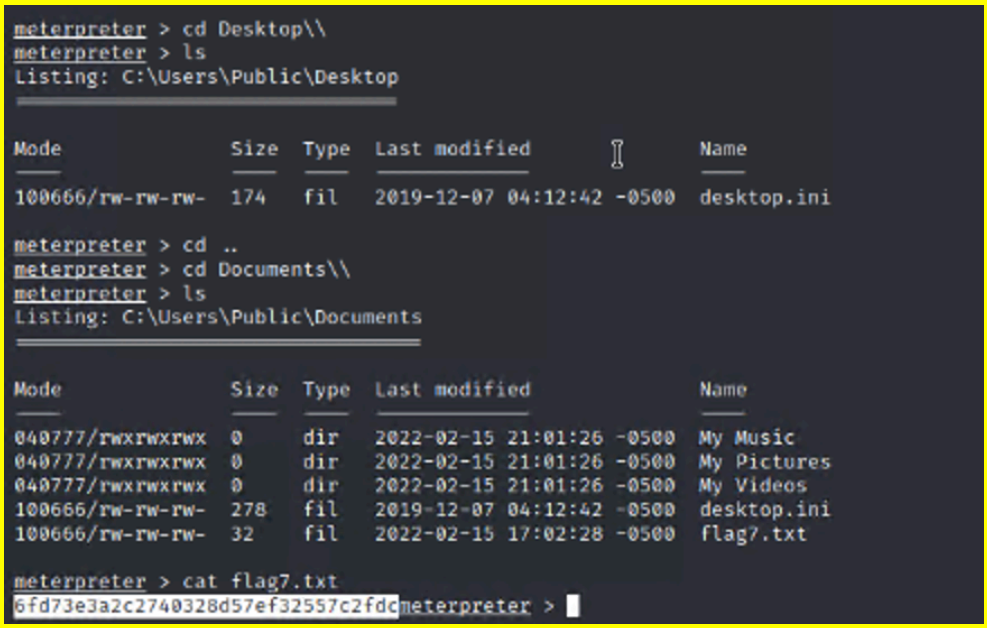
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Medium
Description	Utilizing Nmap, notice SLMail on port 25 and 110. Via MSFconsole, use “windows/pop3/seattlelab_pass” exploit. When a meterpreter session is created ls to find flag 4.
Images	
Affected Hosts	172.22.117.20
Remediation	Keeping software up to date with the most recent security updates

Vulnerability 5	Findings
Title	Scheduled Task Vulnerability
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Medium
Description	Utilizing the previous exploit, navigate to “schtasks /query” to see all the tasks and the flag.

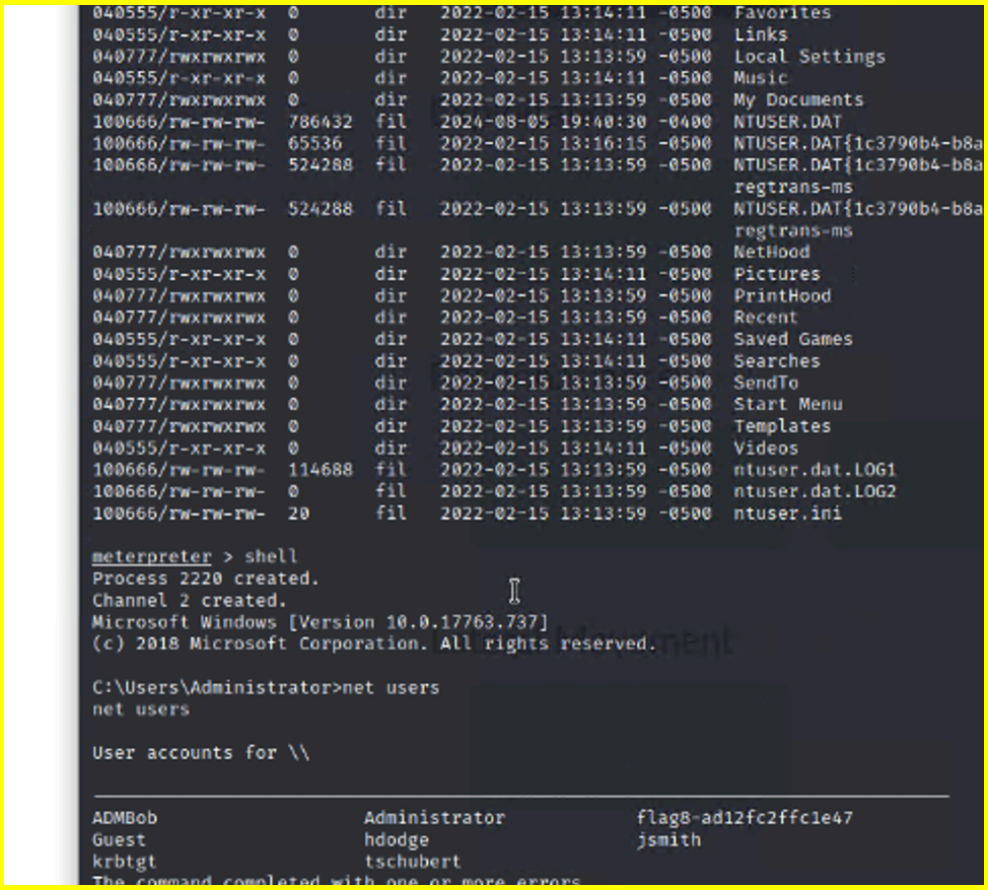
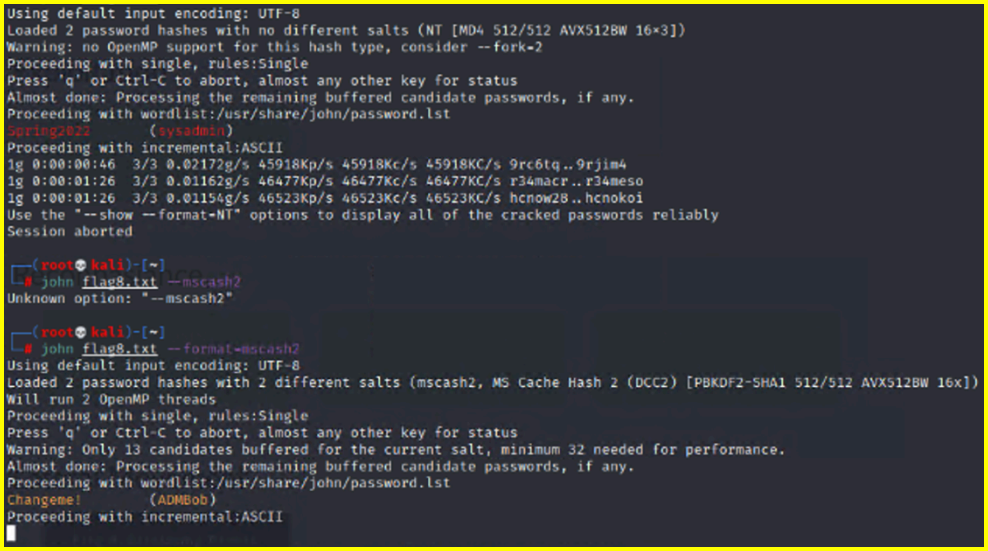
Images	
Affected Hosts	172.22.117.20
Remediation	Keeping software up to date with the most recent security updates

Vulnerability 6	Findings
Title	Kiwi Flag 6
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Critical
Description	Utilizing Kiwi, dump the SAM files and crack the hash with John the Ripper
Images	
Affected Hosts	172.22.117.20
Remediation	Securely store all password hashes

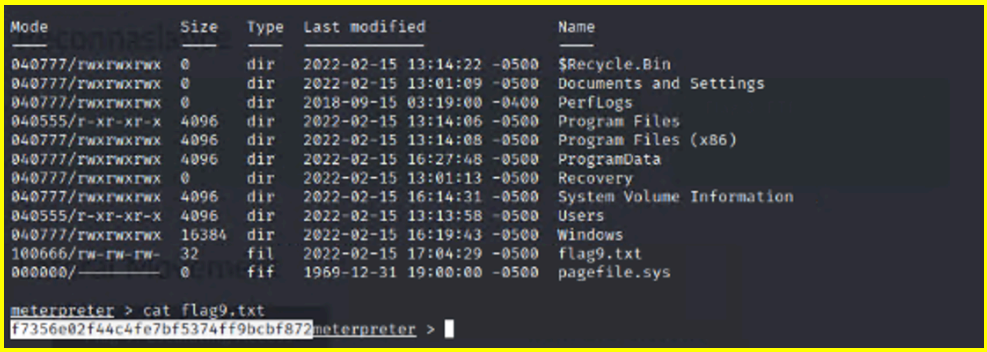
Vulnerability 7	Findings

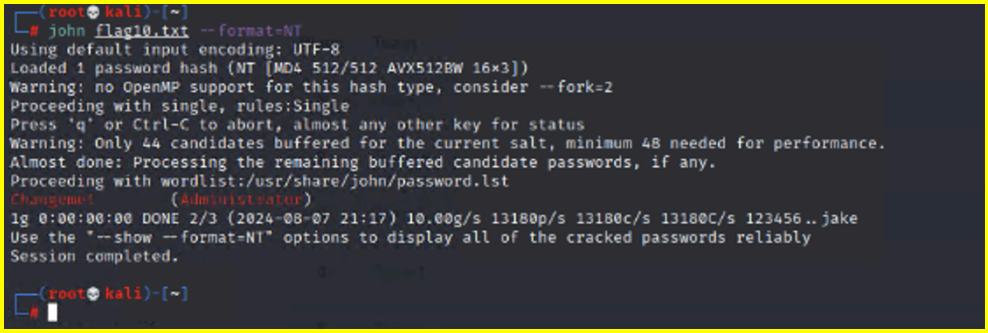
Title	Lateral Movement
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Critical
Description	The flag was located while searching the host
Images	
Affected Hosts	172.22.117.20
Remediation	Keeping software up to date with the most recent security updates

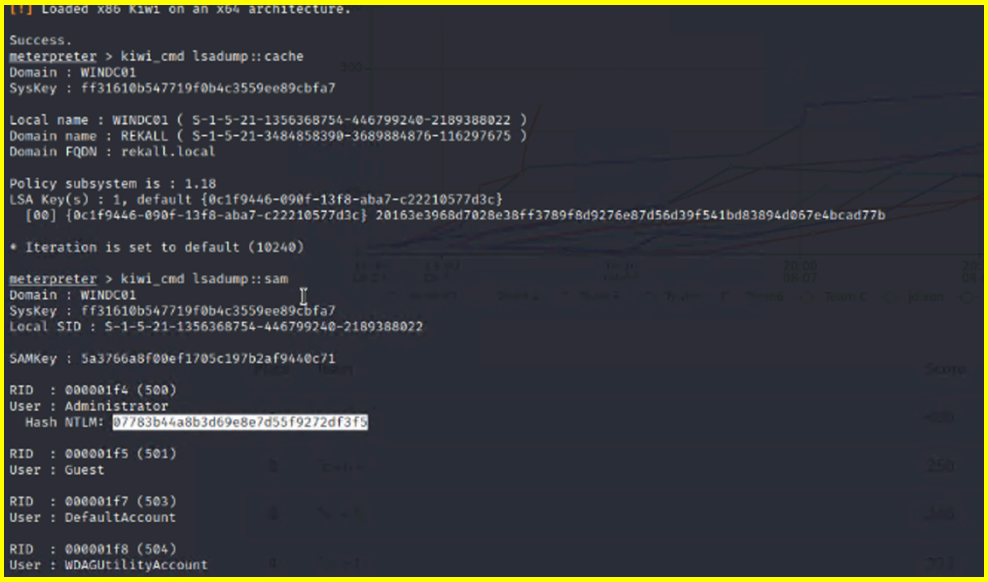
Vulnerability 8	Findings
Title	MsCacheV2 vis LSADump
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Critical
Description	Utilizing Kiwi, lsadump the cache (kiwi_cmd lsadump::cache). This provided the hashes for the administrator and ADMBob among some others

Images	 
Affected Hosts	172.22.117.20
Remediation	Not storing any credentials in the cache and implement multi-factor authentication

Vulnerability 9	Findings
Title	Navigating the C:\ directory

Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Critical
Description	Navigating to the root directory, flag 9 was found
Images	
Affected Hosts	172.22.117.20
Remediation	Emplement better access control to protect sensitive data.

Vulnerability 10	Findings
Title	Default Administrator Credentials
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	High
Description	Utilizing Kiwi, dump the DCSync to reveal the Administrator NTLM password Hash. Use John the Ripper to extract the password.
Images	



```
[!] Loaded x86 Kiwi on an x64 architecture.

Success.
meterpreter > kiwi_cmd lsadump::cache
Domain : WINDC01
SysKey : ff31610b547719f0b4c3559ee89cbfa7

Local name : WINDC01 ( S-1-5-21-1356368754-446799240-2189388022 )
Domain name : REKALL ( S-1-5-21-3484858390-3689884876-116297675 )
Domain FQDN : rekall.local

Policy subsystem is : 1.18
LSA Key(s) : 1, default {0c1f9446-090f-13f8-aba7-c22210577d3c}
  [00] {0c1f9446-090f-13f8-aba7-c22210577d3c} 20163e3968d7028e38ff3789f8d9276e87d56d39f541bd83894d067e4bcad77b

* Iteration is set to default (10240)

meterpreter > kiwi_cmd lsadump::sam
Domain : WINDC01
SysKey : ff31610b547719f0b4c3559ee89cbfa7
Local SID : S-1-5-21-1356368754-446799240-2189388022

SAMKey : 5a3766a8f00ef1705c197b2af9440c71

RID  : 000001f4 (500)
User : Administrator
  Hash NTLM: 07783b44a8b3d69e8e7d55f9272df3f5

RID  : 000001f5 (501)
User : Guest

RID  : 000001f7 (503)
User : DefaultAccount

RID  : 000001f8 (504)
User : WDAGUtilityAccount
```

```
Proceeding with wordlist:/usr/share/john/password.lst
Spring2022       (sysadmin)
Proceeding with incremental:ASCII
1g 0:00:00:46  3/3 0.02172g/s 45918Kp/s 45918Kc/s 45918KC/s 9rc6tq..9rjim4
1g 0:00:01:26  3/3 0.01162g/s 46477Kp/s 46477Kc/s 46477KC/s r34macr..r34meso
1g 0:00:01:26  3/3 0.01154g/s 46523Kp/s 46523Kc/s 46523KC/s hcnow28..hcnokoi
Use the "--show --format=NT" options to display all of the cracked passwords reliably
Session aborted

┌──(root💀kali)-[~]
└─# john flag8.txt --mscash2
Unknown option: "--mscash2"

┌──(root💀kali)-[~]
└─# john flag8.txt --format=mscash2
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (mscash2, MS Cache Hash 2 (DCC2) [PBK
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 13 candidates buffered for the current salt, minimum 32 needed for perf
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Changeme!        (ADMBob)
Proceeding with incremental:ASCII
1g 0:00:19:07  3/3 0.000871g/s 10122p/s 10122c/s 10122C/s mumanget..mumarito
1g 0:00:19:07  3/3 0.000871g/s 10122p/s 10123c/s 10123C/s purpen12..purpring
Use the "--show --format=mscash2" options to display all of the cracked passwords rel
Session aborted

┌──(root💀kali)-[~]
└─# nano flag10.txt

┌──(root💀kali)-[~]
└─# john flag10.txt --format=NT
Using default input encoding: UTF-8
Loaded 1 password hash (NT [MD4 512/512 AVX512BW 16×3])
Warning: no OpenMP support for this hash type, consider --fork=2
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 44 candidates buffered for the current salt, minimum 48 needed for perf
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Changeme!        (Administrator)
1g 0:00:00:00 DONE 2/3 (2024-08-07 21:17) 10.00g/s 13180p/s 13180c/s 13180C/s 123456.
Use the "--show --format=NT" options to display all of the cracked passwords reliably
Session completed.

┌──(root💀kali)-[~]
└─#
```

Affected Hosts	172.22.117.20
Remediation	Ensure passwords hashes are properly protected against tools like Kiwi.

	Employ better methods of authentication.