Cybersecurity

Penetration Test Report

Rekall Corporation

Penetration Test Report

Confidentiality Statement

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

Table of Contents

Contact Information

Company Name	LookingHackwards PTY LTD
Contact Name	Jeff Goldblum
Contact Title	Chief Hacking Officer

Document History

Version	Date	Author(s)	Comments
001	12/01/2024	Owen	Edits
002	18/01/2024	Owen	Edits
003	20/01/2024	Owen	Edits
004	20/01/2024	Owen	Edits
005	21/01/2024	Owen	Edits
005	21/01/2024	Owen	Edits
006	22/01/2024	Owen	Edits
007	22/01/2024	Owen	Edits
008	22/01/2024	Owen	Edits

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.

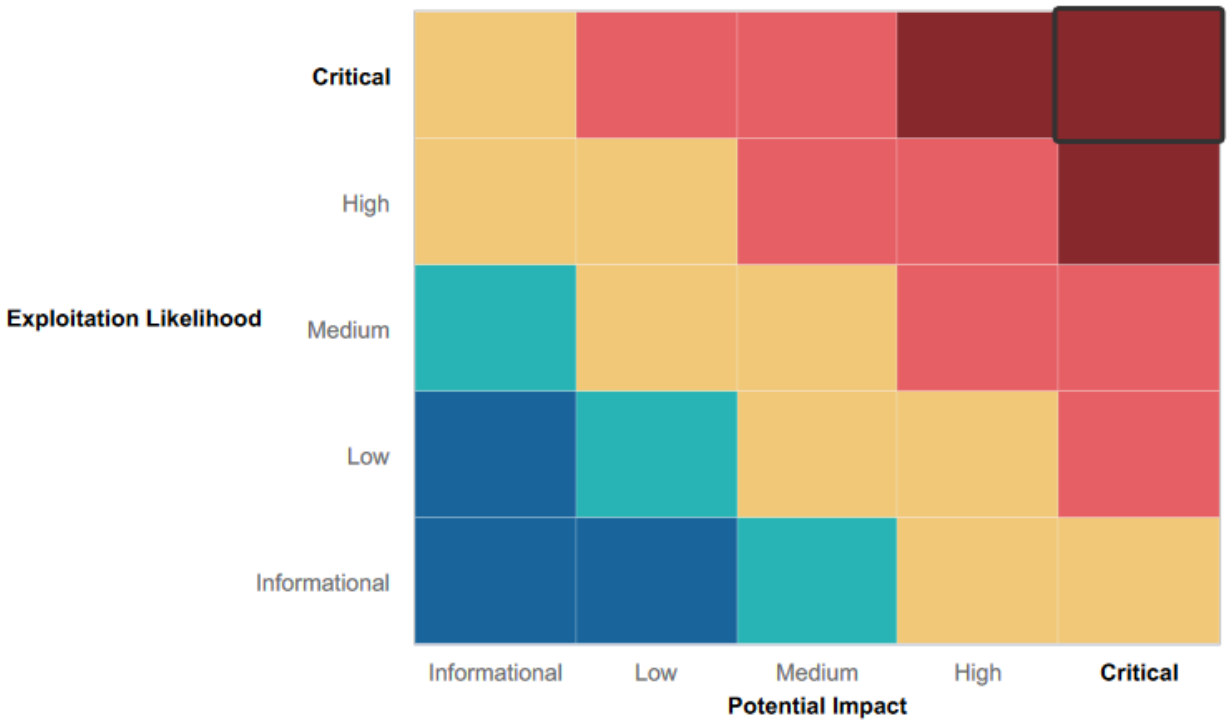
Executive Summary of Findings

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.

- Validation required for most inputs on the website
- Protection from many exploits such as LFI and XSS in many cases as it took several attempts to exploit

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.

- SQL Injections
- Exposed User Credentials
- PHP Injection
- Brute Force / Session Hijacking
- Nmap scan
- Remote Code Injection (RCE
- Shellshock exploit on web server
- Exposed Admin with weak password
- Weak password and Nmap
- SLMail Vulnerability
- dcsync (Domain Controller Impersonation)

Executive Summary

Through the scope of the outlined objectives, LOOKINGHACKWARDS was able to navigate Rekall's systems through various methods available to both the public or layman to much more advanced techniques. We were able to compromise at least three machines both Linux and Windows, as well as the Domain Controller.
LOOKINGHACKWARDSs tests revealed more than 25 vulnerabilities allowing us access to both the Linux and Windows10 systems enabling us to escalate our privileges to both root and System as well as move laterally through the systems thus allowing us to maintain persistence in compromising the various systems. Many of these vulnerabilities in the system are considered critical and could have catastrophic consequences for the Rekall corporation.
LOOKINGHACKWARDS found several vulnerabilities throughout the testing process. We started with the webapp and were able to find it vulnerable to several attacks. First was an XSS reflected attack running a malicious script on the home page. We also found it vulnerable to Local File Inclusion and PHP injection attacks were successful. We were able to run XSS stored scripting on the comment page and an SQL injection attack was able to reveal the login credentials of an Admin user.
Through a brute force attack using Burp Suite's Intruder tool we gained access to the Admin Legal Documents We found the robots.txt file to be accessible and contained sensitive information. There were several instances of sensitive data found to be exposed through Open Source and view through OSINT. We were able to find a stored certificate and user login credentials in the HTML code.
Navigating the Linux environment we found through scans that 5 IPs were exposed and potentially vulnerable to attack. One host was running Drupal allowing us to run an exploit opening a meterpreter session. Using a Struts vulnerability we accessed the root directory. We were also able to use a shellshock exploit to access the sudoers and passwd files. Finally we were able to find a username that allowed us to enter the system and escalate our privileges to root.
In the Windows environment we were able to use an SLMail exploit to gain access to the system. Navigating through we discovered that we could find users and password hashes for administrators across the system as well as open reverse shell in meterpreter. We were able to access the Task Scheduler and were also able to crack the password hashes without any difficulty. We were able to view and access all directories by the completion of our scoped tasks.

Primarily username and password strength and password policies all need immediate updating as many of the critical risks stem from this.
Many of the vulnerabilities we found could be used in a malicious fashion to devastating results for the Rekall corporation.

We've included screenshots of our exploits in the Vulnerabilities section so they can be reproduced and/or remediated.

.

Summary Vulnerability Overview

Vulnerability	Severity
XSS Reflected	Medium
Sensitive Data Exposure in Response Header	High
Local File Inclusion	High
SQL Injections	Critical
Exposed User Credentials	Critical
Sensitive Data Exposure in Robots.txt file	High
Command Injection	High
PHP Injection	Critical
Brute Force / Session Hijacking	Critical
Directory / Path Transversal	High
Open Source Exposed Data - a	Medium
Open Source Exposed Data - b	Medium
Open Source Exposed Data - c	Medium
Nmap Reconnaissance Scan	High
Nmap scan	Critical
Nessus Scan	High
Remote Code Injection (RCE)	Critical
Shellshock exploit on web server	Critical
Shellshock	Critical
Nessus Scan	High
Drupal Exploit	High
Exposed Admin with weak password	Critical
Nmap scan	High
Weak password and Nmap	Critical
SLMail Vulnerability	Critical
Task Scheduler	Medium
Credentials Dump	High

Directory search	Medium
Meterpreter	Critical
Meterpreter	Critical
dcsync	Critical

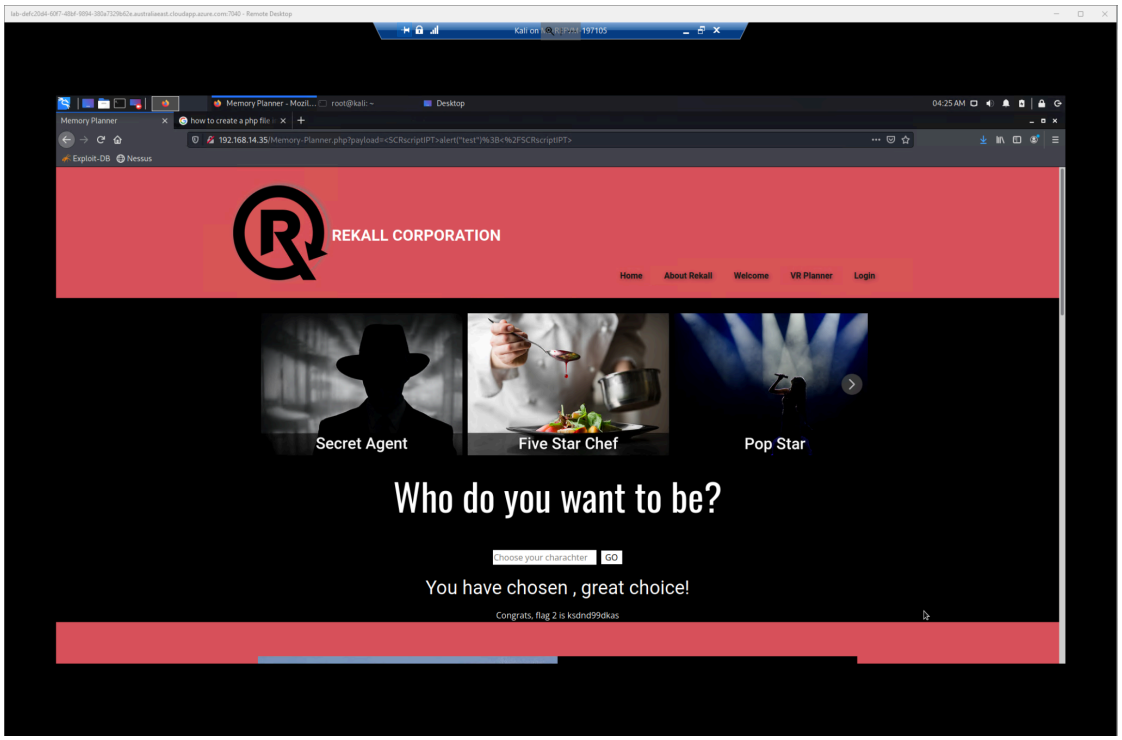
The following summary tables represent an overview of the assessment findings for this penetration test:

Scan Type	Total
Hosts	172.22.117.10 172.22.117.20 192.168.13.1 192.168.13.10 192.168.13.11 192.168.13.12 192.168.13.13 192.168.13.14 192.168.14.35
Ports	21 22 80 8080 106 110

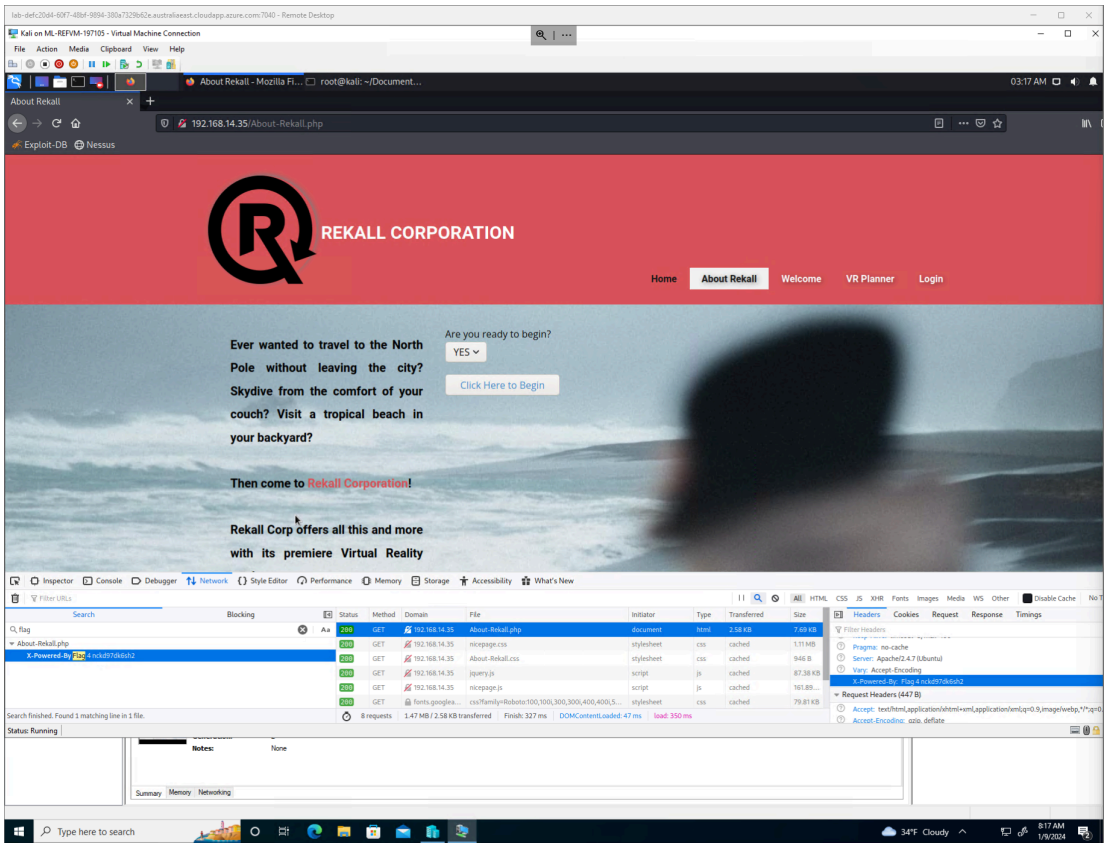
Exploitation Risk	Total
Critical	14
High	11
Medium	6
Low	0

Vulnerability Findings

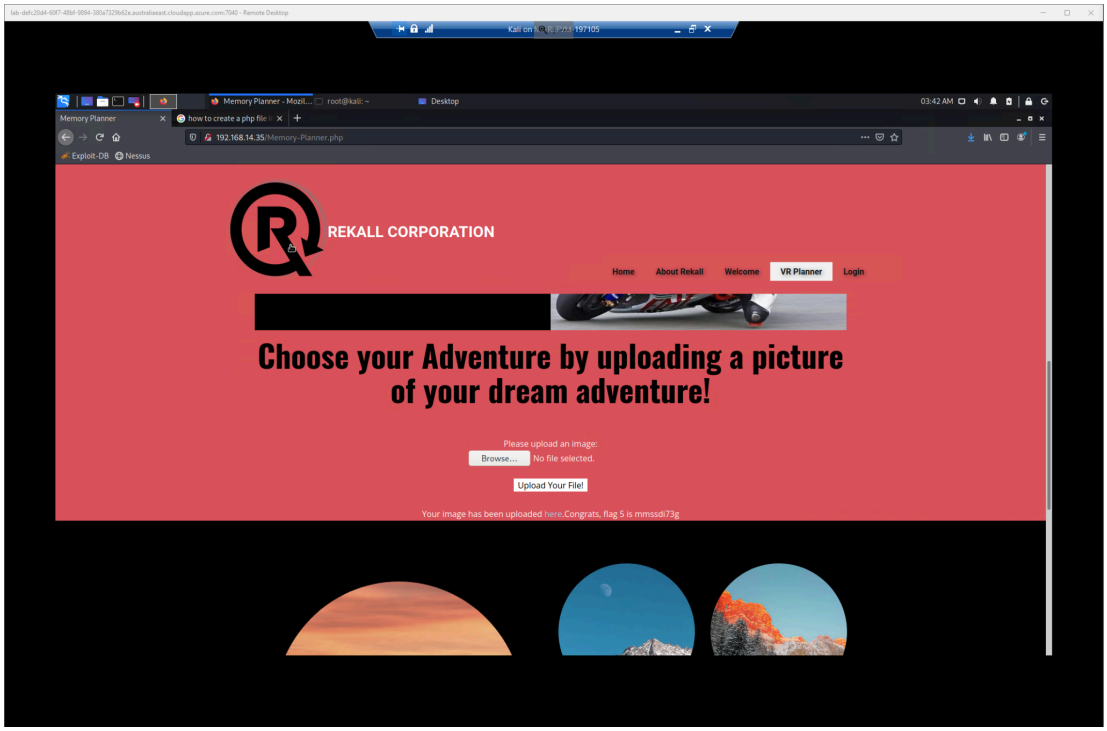
Vulnerability 1	Findings
Title	XSS Reflected
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Medium

Description	Malicious script to bypass input validation was created with the assistance of online research. The command “<SCRscriptIPT>alert(“test”);</SCRscriptIPT> was entered into ‘choose your character field’ which returned the flag.
Images	
Affected Hosts	192.168.14.35/Memory-Planner
Remediation	• Always Validate user input to ensure it adheres to the expected format • Encode user-generated data before rendering it in HTML templates or sending to the client • Properly sanitize output data

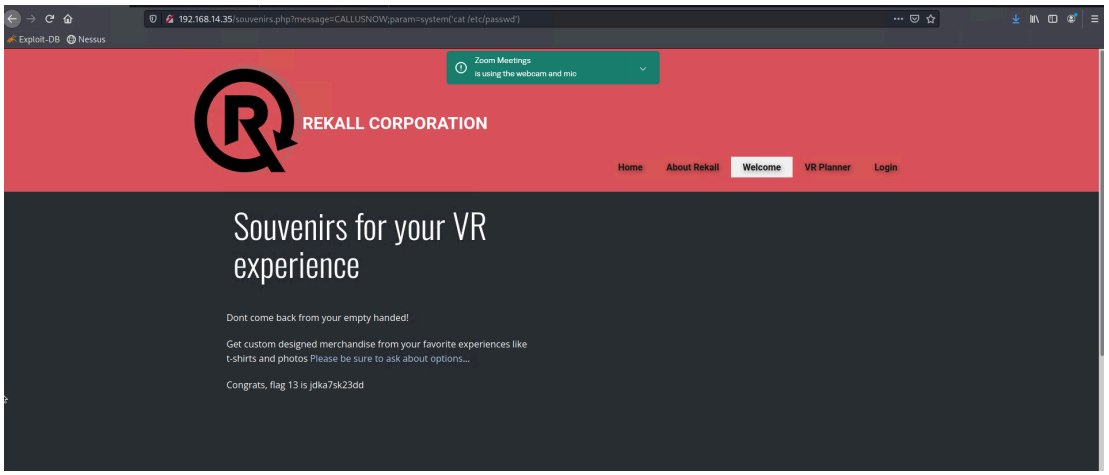
Vulnerability 2	Findings
Title	Sensitive Data Exposure on Response Header
Type (Web app / Linux OS / Wlndows OS)	Web App
Risk Rating	High
Description	Sensitive data was found while exploring through HTTP responses

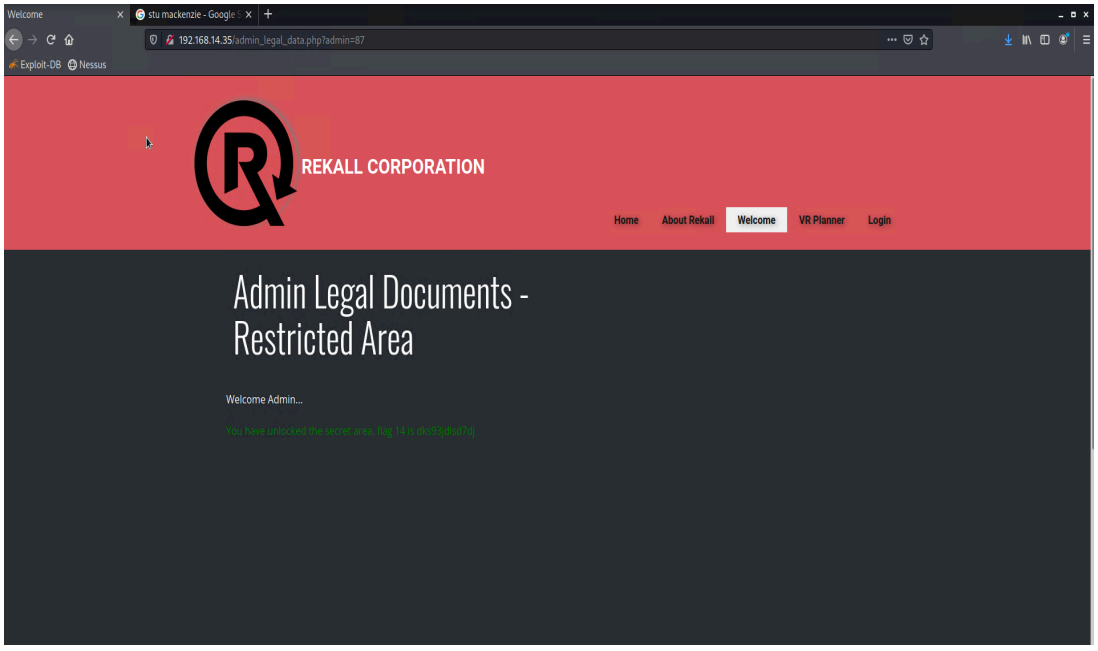
Images	
Affected Hosts	192.168.14.35/About-Rekall
Remediation	• Mask or remove sensitive data in all responses before they are returned from the server • Secure HTTP Headers by using HTTP headers like CSP and HTTP Strict Transport Security

Vulnerability 3	Findings
Title	Local File Inclusion
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	High
Description	A PHP file was created through our local system which was successfully uploaded onto the ‘Memory-Planner’ page

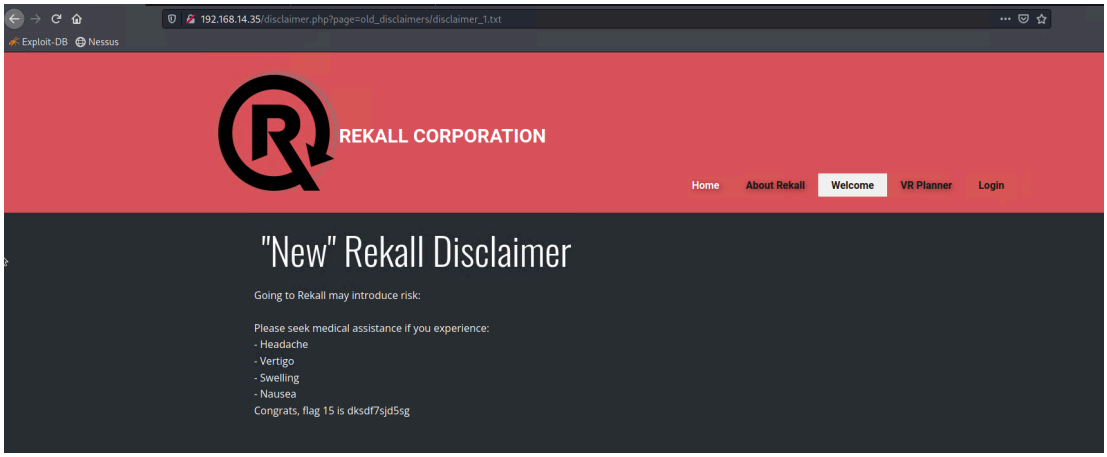
Images	
Affected Hosts	192.168.14.35/Memory-Planner
Remediation	• Apply server-side validation not on the client side • Use a whitlest of files and ignore every other filename and path • Disable PHP execution in Upload directories

Vulnerability 4	Findings
Title	PHP injection
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	High
Description	On a previous flag we looked at Robots.txt and found a hidden subdirectory called souvenirs.php. Loaded it up to try the PHP inject querying the system that I attempted on the disclaimers directory and a few others. I tried a simple “192.168.14.35/souvenirs/php?param=system(‘cat /etc/passwd’)” (i did consult ChatGPT for this injection). This didnt work. When you click on options it changes the URL and includes “message=CALLUSNOW” so i tried the same command but left that in resulting in “192.168.14.35/souvenirs.php?message=CALLUSNOW;param=system(‘cat /etc/passwd’)”

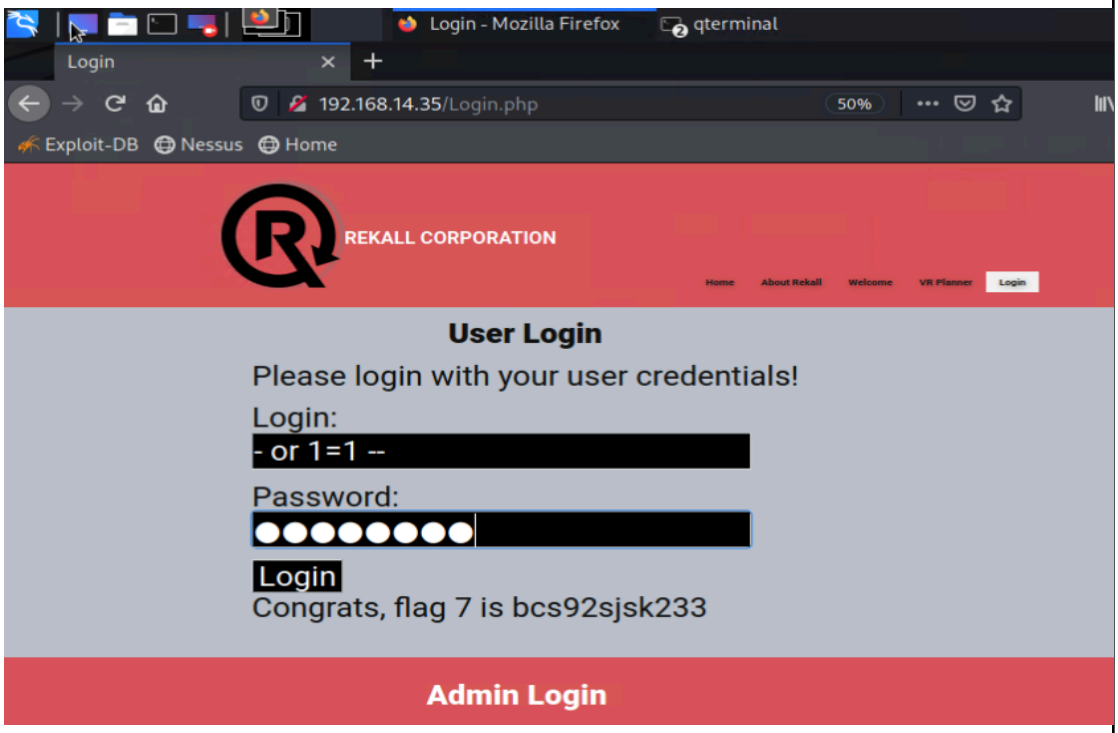
Images	
Affected Hosts	192.168.14.35
Remediation	• Validate and sanitise all user inputs • Use output encoding • Set correct permissions

Vulnerability 5	Findings
Title	Brute Force
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	Using the Burp Suite Intruder tool i was able to brute force session ID's and this was able to allow me to gain the admin session token and get the 14th flag
Images	

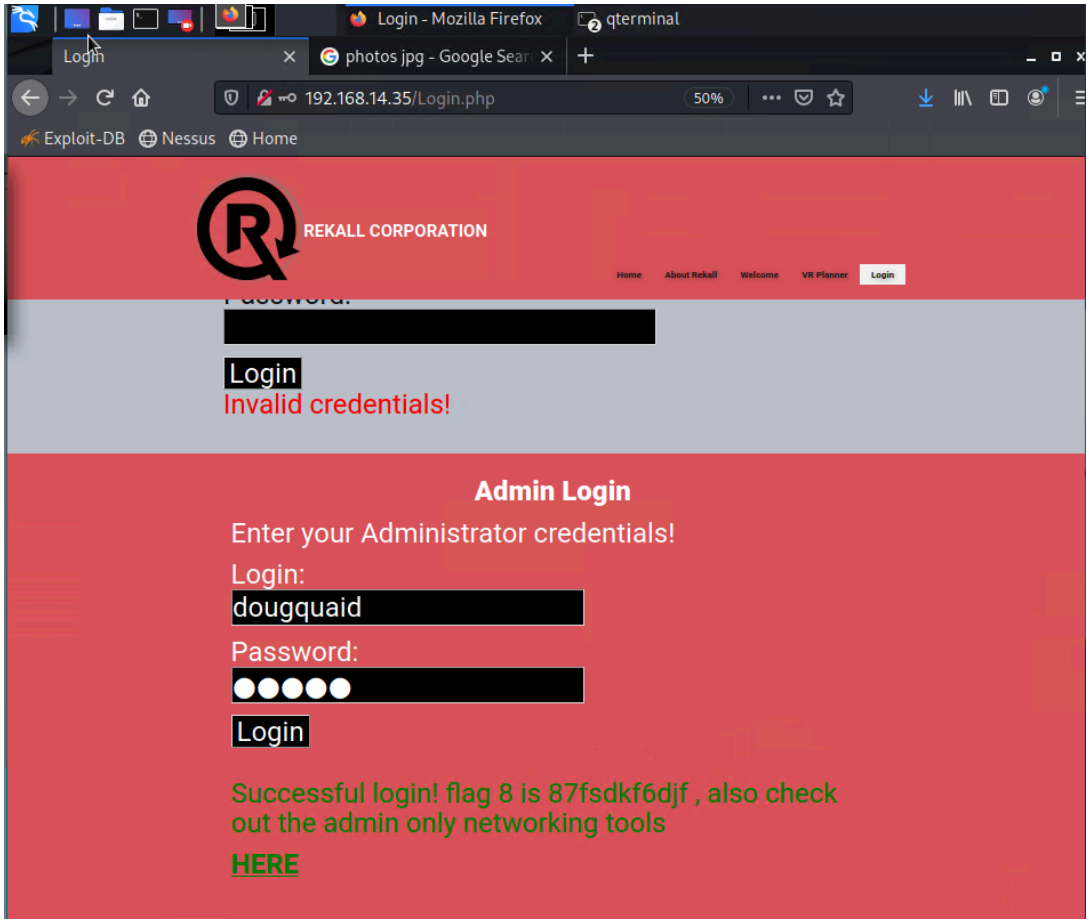
Affected Hosts	192.168.14.35
Remediation	• Use secure, random session IDs • Rotate session ID after login and SSL enforce for all connections • Implement timeouts for sessions

Vulnerability 6	**Findings**
Title	Directory Transversal
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	This vulnerability was found on the disclaimer.php page (like every website usually has a disclaimer so not uncommon to find). It said “”New” Rekall Disclaimer”, tried to bring up an old disclaimer in which i found disclaimer_1.txt. This contained the flag command used “192.168.14.35/disclaimer.php?page=old_disclaimers/disclaimer_1.txt”
Images	
Affected Hosts	192.168.14.35
Remediation	• Validate, filter and sanitise all user inputs • Never all for unsantised user input that leads to the file system operations • use an allow-list path validation

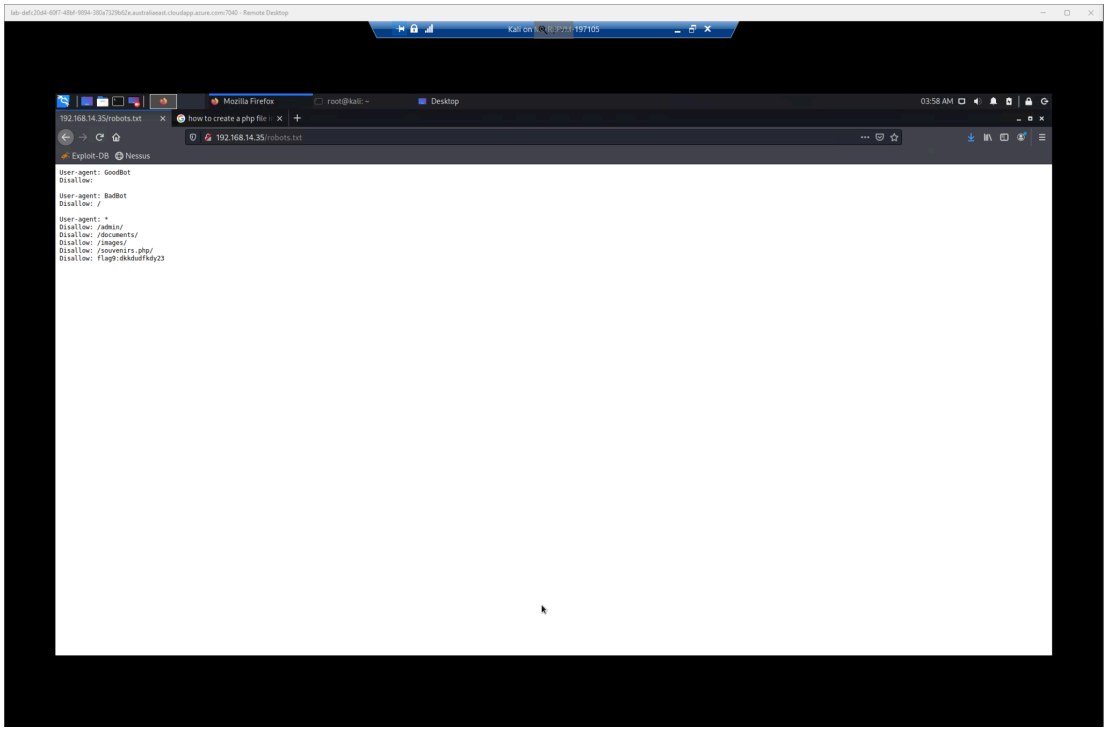
Vulnerability 7	**Findings**
Title	SQL Injection
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Critical
Description	We were able to login using SQL injection username: ‘- or 1=1 –’ and password:

	‘password’
Images	
Affected Hosts	192.168.14.35
Remediation	• don’t allow certain characters

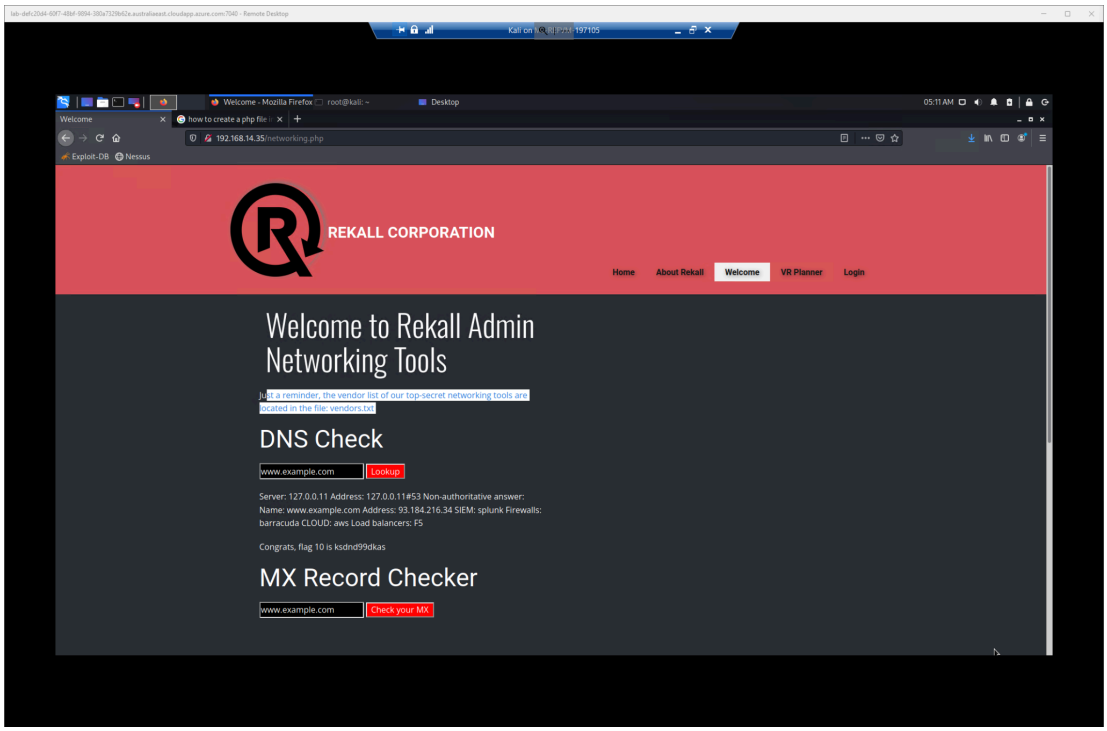
Vulnerability 8	Findings
Title	Exposed User Credentials
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Critical
Description	On the ‘login’ page if we right click on “login” box > click “inspect element” > on the left hand side of the inspector above the current selection we see “dougquaid” as the hidden text (username> inspect element for the “password” form > above the selected element you will see “kuato” (that is the password for the admin account) > login to the account with the username and password

Images	
Affected Hosts	192.168.14.35/robots.txt
Remediation	• Remove information from the HTML • Utilize 2 FA

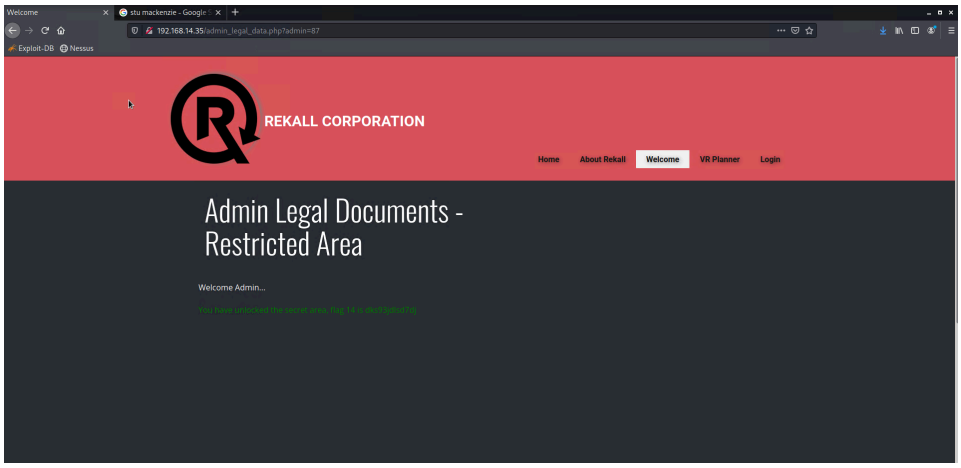
Vulnerability 9	Findings
Title	Sensitive Data Exposure in Robots.txt file
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	High
Description	A robots.txt file tells search engine crawlers which URLs the crawler can access on your site. This is used mainly to avoid overloading your site with requests. The discovery was easily located by adding /robots.txt to host mentioned below

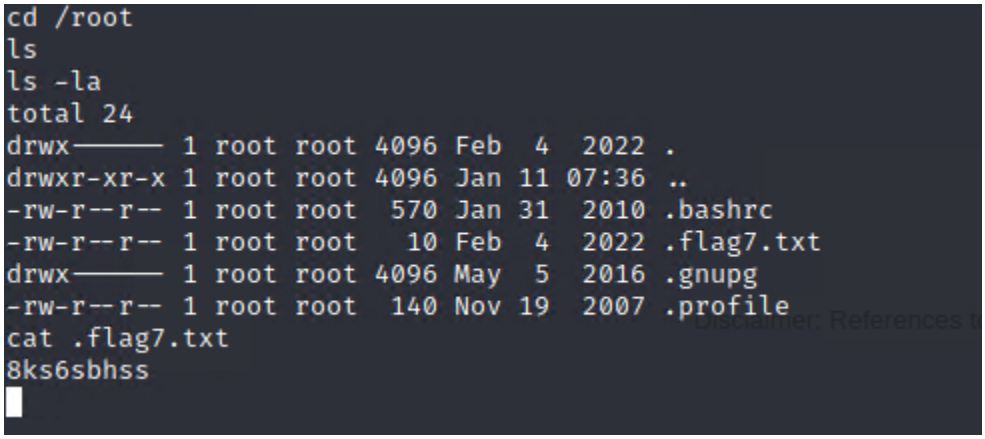
Images	
Affected Hosts	192.168.14.35/robots.txt
Remediation	• Don't store any sensitive information in robots.txt or any other publicly accessible files

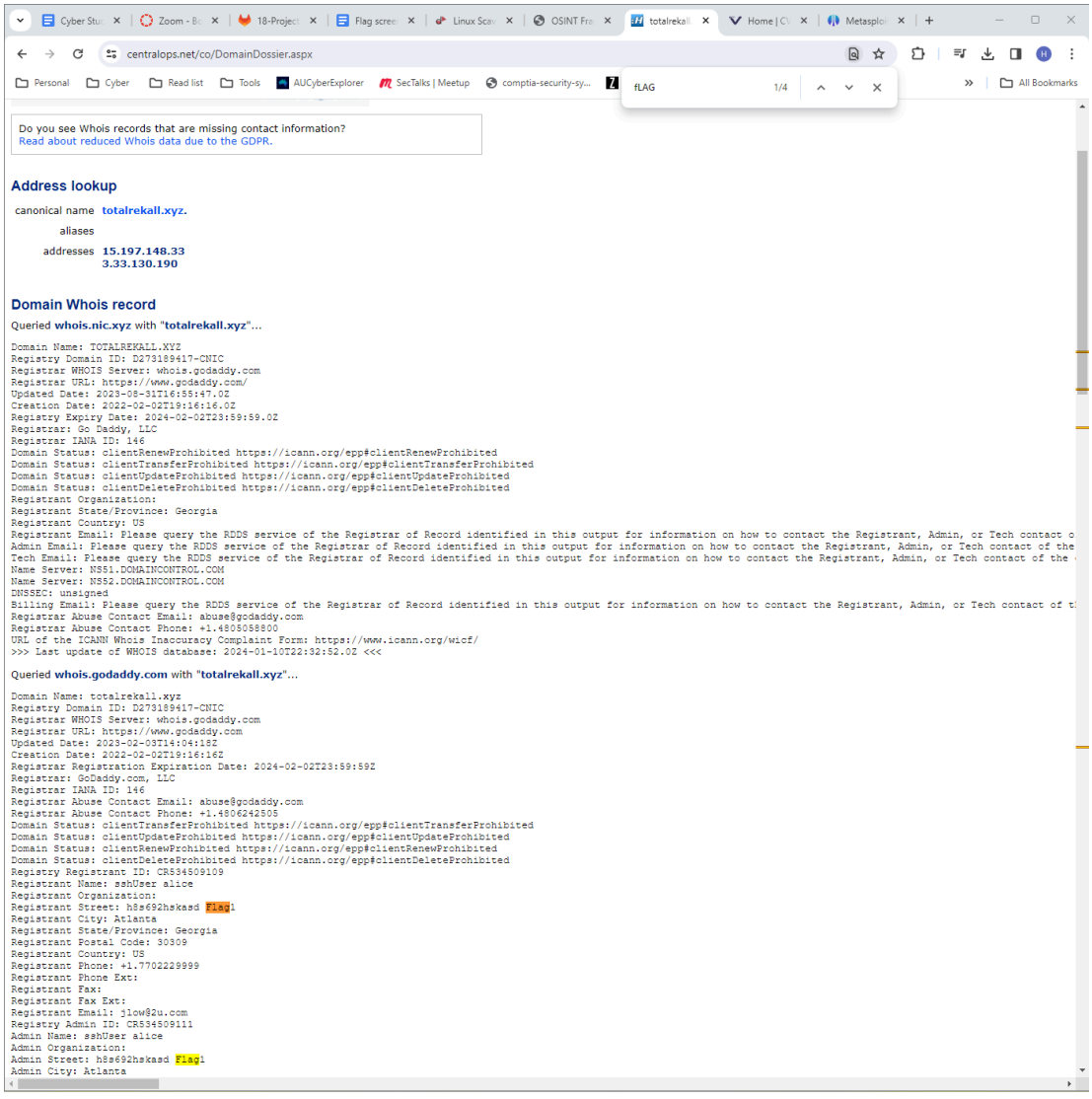
Vulnerability 10	Findings
Title	Command Injection
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	High
Description	Command injection was successfully executed allowing us to run arbitrary commands on the server by manipulating input fields designed to execute system commands. We were able to locate the page 192.168.14.35/networking.php and from there we could enter the command ;cat vendors.txt in the DNS check input field

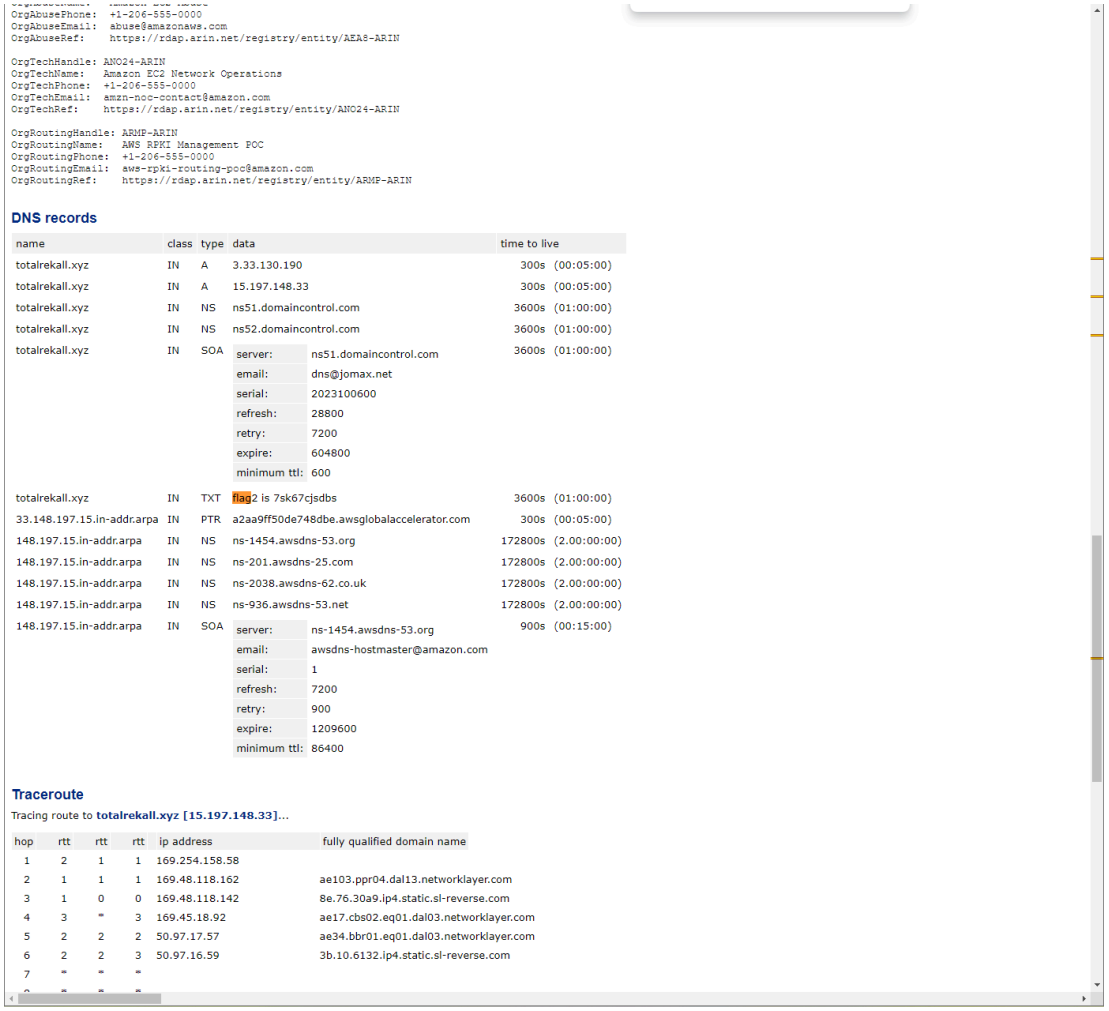
Images	
Affected Hosts	192.168.14.3/Networking.php
Remediation	• If possible remove the text that mentions the vendors.txt file is “top-secret” • Validate user input and only allow command needed for the task

Vulnerability 11	Findings
Title	Brute Force
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	Using the Burp Suite Intruder tool i was able to brute force session ID’s and this was able to allow me to gain the admin session token and get the 14th flag

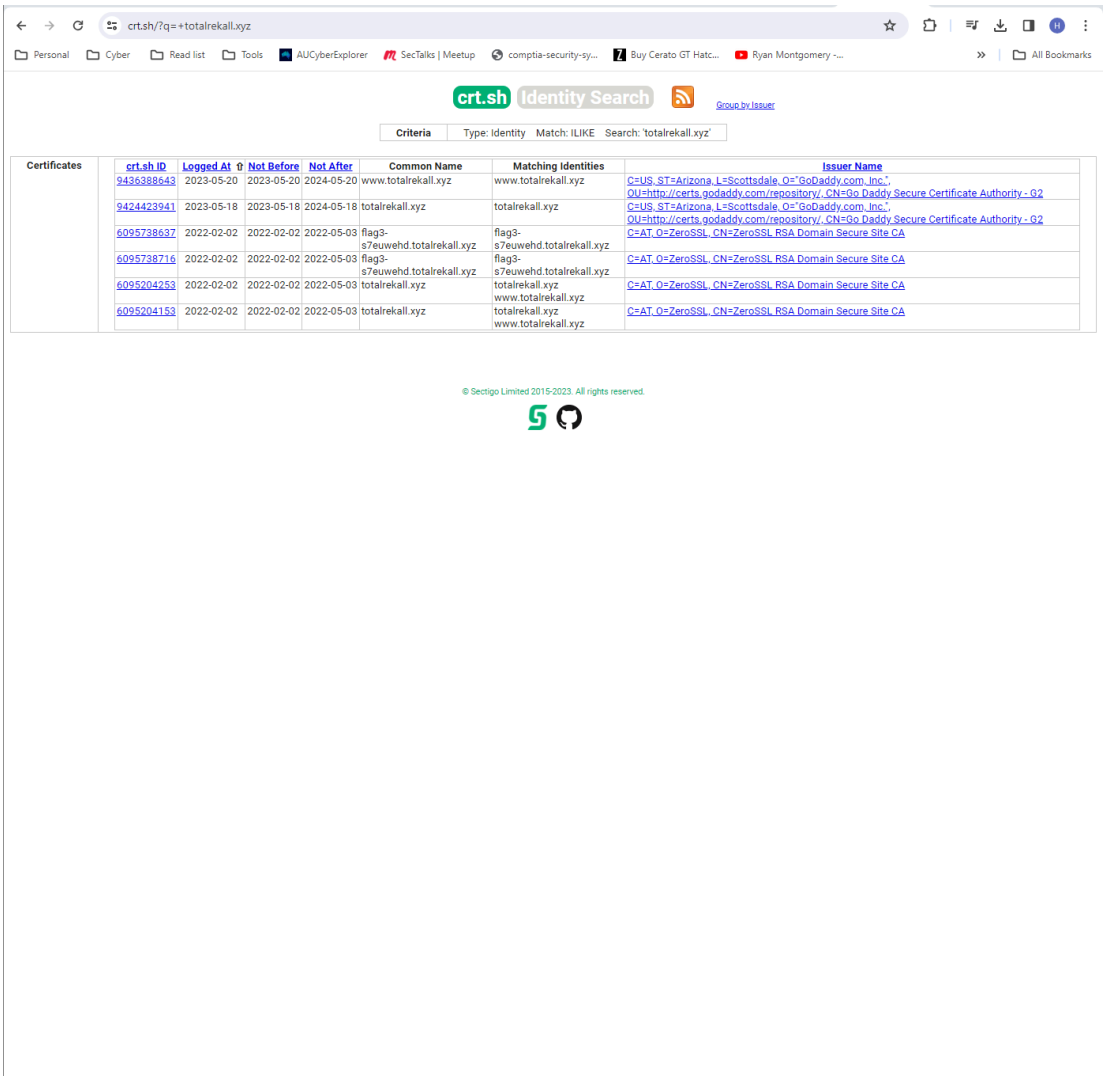
Images	
Affected Hosts	192.168.14.35/admin_legal_data
Remediation	• Limiting Login attempts • Enable MFA • Block malicious IPs • Strong password policy

Vulnerability 12	Findings
Title	Remote Code Execution
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	Was able to exploit a flaw in Apache where port 8080 was open and was able to execute a RCE exploit to get into the system
Images	
Affected Hosts	192.168.13.13
Remediation	• Regular patching and update (Essential 8) • Disable services that are unnecessary. • Least privilege principle

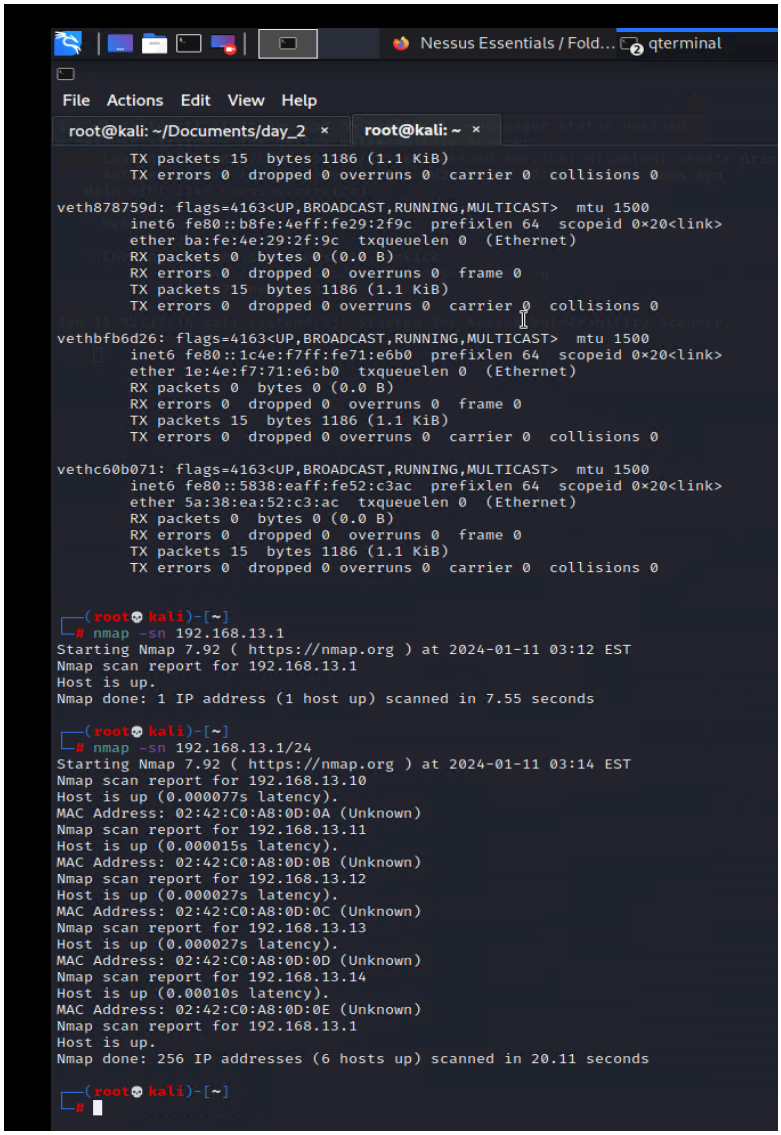
Vulnerability 13	Findings
Title	Open Source Exposed Data - a
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	Conducted reconnaissance via the OSINT tool for domain public records such-as whois records and able to locate sensitive data
Images	
Affected Hosts	https://centralops.net/co/DomainDossier.aspx
Remediation	• Register domains privately to avoid exposing sensitive information to the public domain

Vulnerability 14	Findings
Title	Open Source Exposed Data - b
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	Exactly same steps as above and discovered more sensitive information when looking into the data further
Images	
Affected Hosts	https://centralops.net/co/DomainDossier.aspx
Remediation	• Register domains privately to avoid exposing sensitive information to the public domain

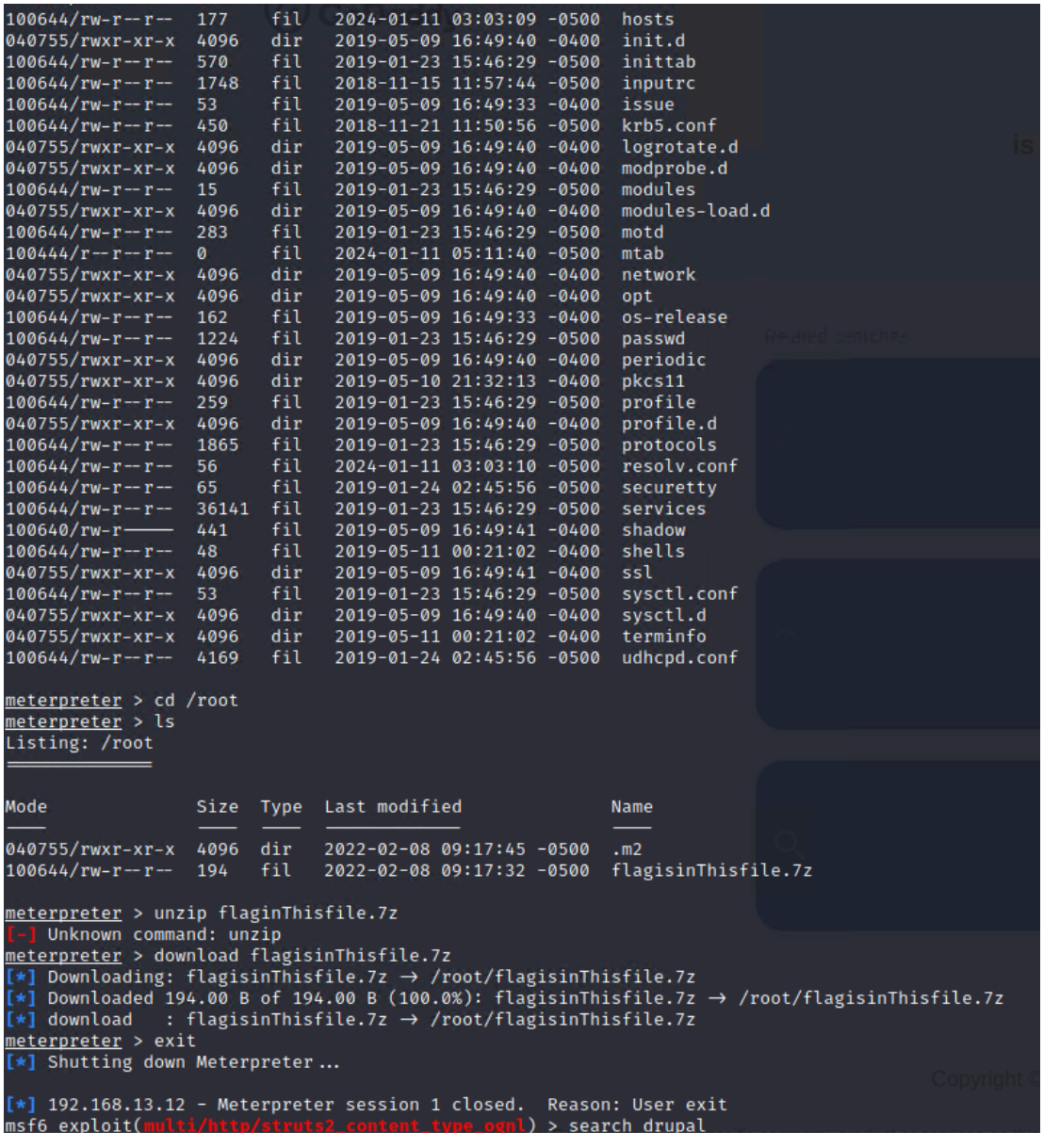
Vulnerability 15	Findings
Title	Open Source Exposed Data x - c

Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	Conducted SSL info via the public domain site crt.sh and was able to locate more sensitive data
Images	
Affected Hosts	crt.sh/?q=+totalrekall.xyz
Remediation	• Restrict DNS records on public domain and maintain a minimum level of exposure

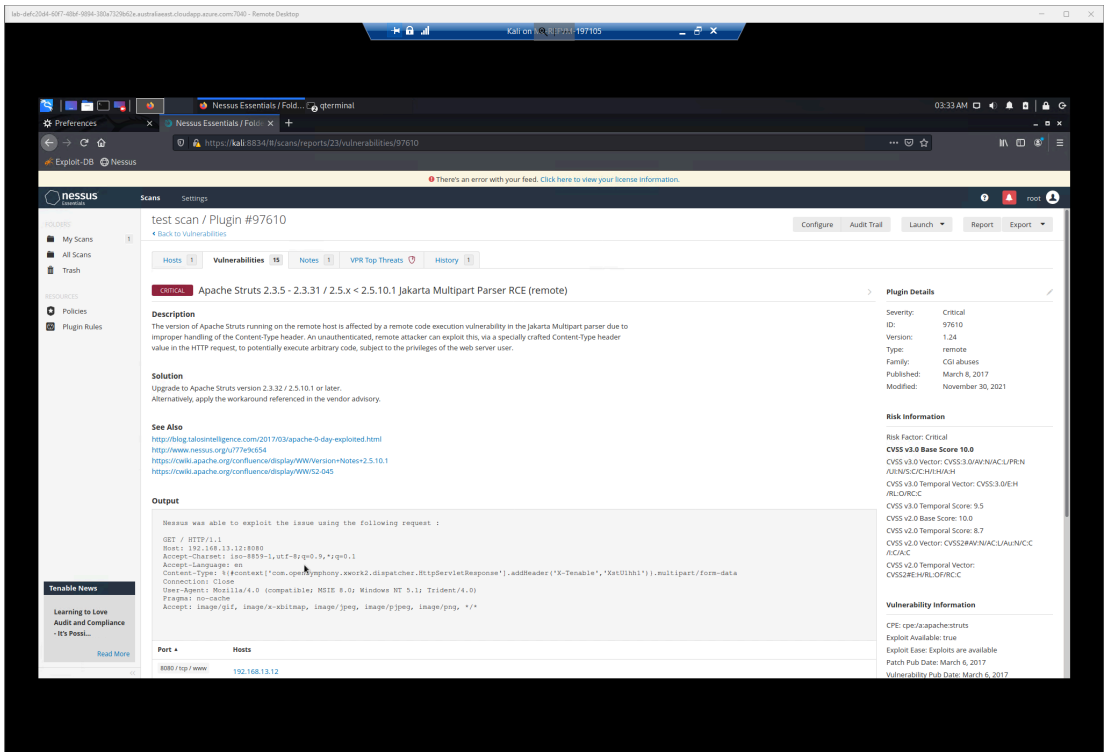
Vulnerability 16	Findings
Title	Nmap Reconnaissance Scan
Type (Web app / Linux OS / WIndows OS)	Linux OS

Risk Rating	High
Description	Conducted a NMAP scan of the IP address & subnet 192.168.13.0/24 and discovered the number of hosts which exposes potentially vulnerable devices on the network
Images	
Affected Hosts	192.168.13.0/24
Remediation	• Introduce firewall and network segmentation to reduce the exposure of internal devices

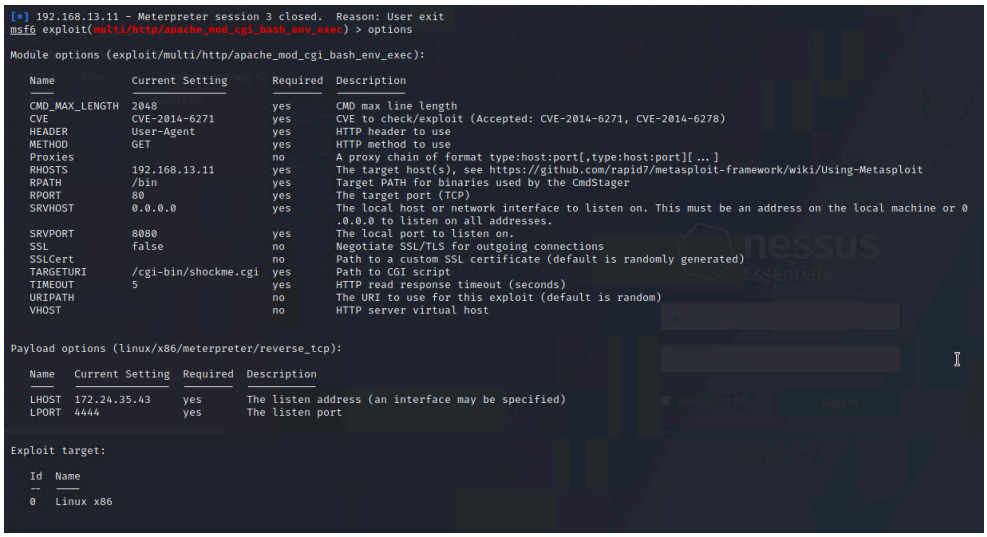
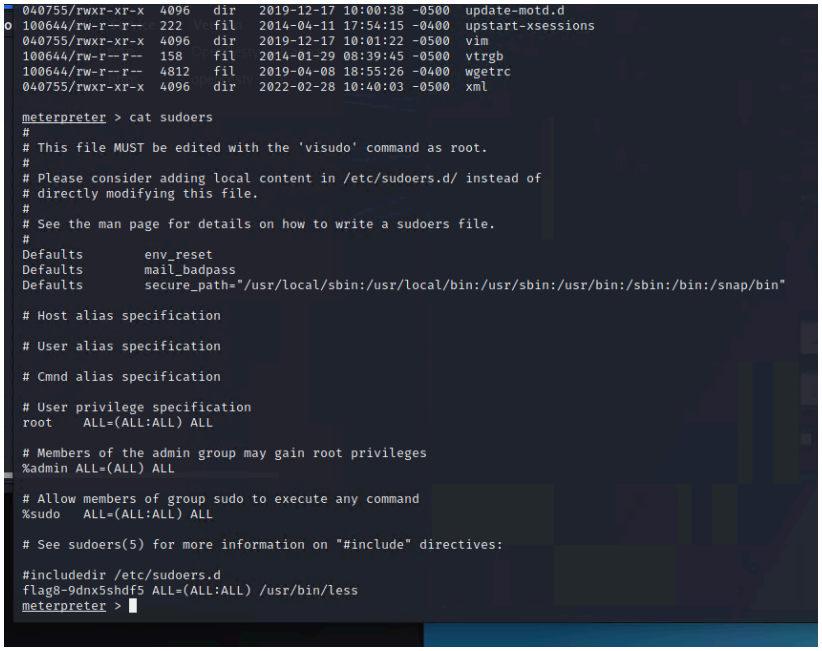
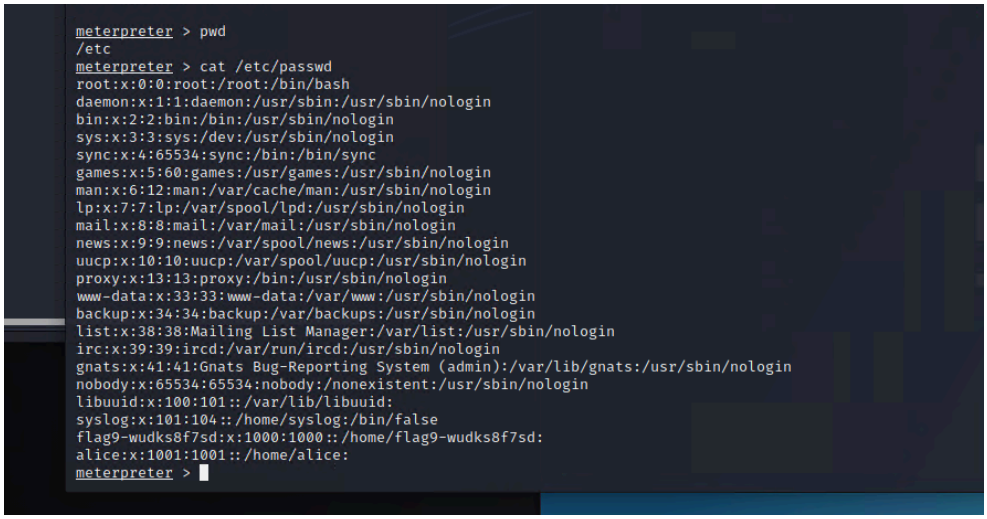
Vulnerability 17	Findings
Title	Struts - CVE-2017-5638
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High

Description	This host has vulnerable struts with the Nessus scan, searched for struts exploits in Metasploit, found struts2_content_type_ognl exploit set the RHOST, LHOST, Checked ports were correct, ran the exploit and downloaded the flag from the linux server using meterpreter thats was found in the /root location, downloaded and unzipped the file and found the flag
Images	
Affected Hosts	192.168.13.12
Remediation	• Regular patching and update (Essential 8) • Disable services that are unnecessary • Least privilege principle

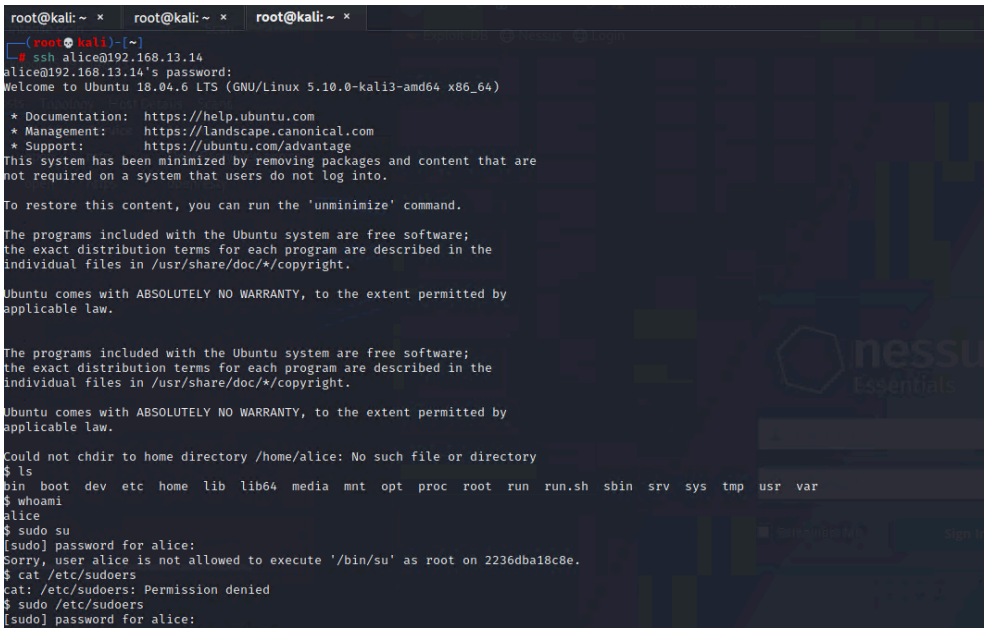
Vulnerability 18	Findings
Title	Nessus Scan

Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	Conducted a advanced scan against the IP 192.168.13.12 which revealed critical vulnerability how
Images	
Affected Hosts	192.168.13.12
Remediation	• Upgrade to Apache Struts version 2.3.32 / 2.5.10.1 or later • Alternatively you can use the workaround in the link below https://www.tenable.com/plugins/nessus/97610

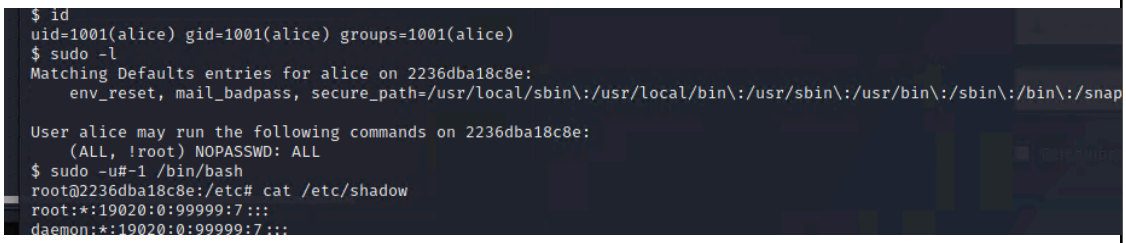
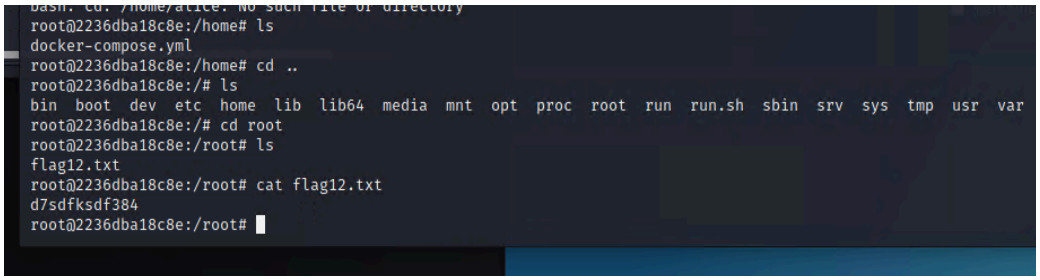
Vulnerability 19	Findings
Title	Shellshock exploit multi/http/apache_mod_cgi_bash_env_exec
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	critical
Description	A flaw in how the Bash shell handles external environment variables. This module targets CGI scripts in the Apache web server by setting the HTTP_USER_AGENT environment variable to a malicious function definition. Allowed us to navigate to /etc/sudoers to view root privilege file and to view the /etc/passwd file Known CVE's: CVE-2014-6271, CVE-2014-6278

Images	  
Affected Hosts	192.168.13.11
Remediation	• Regularly patch and update • Update bash versions

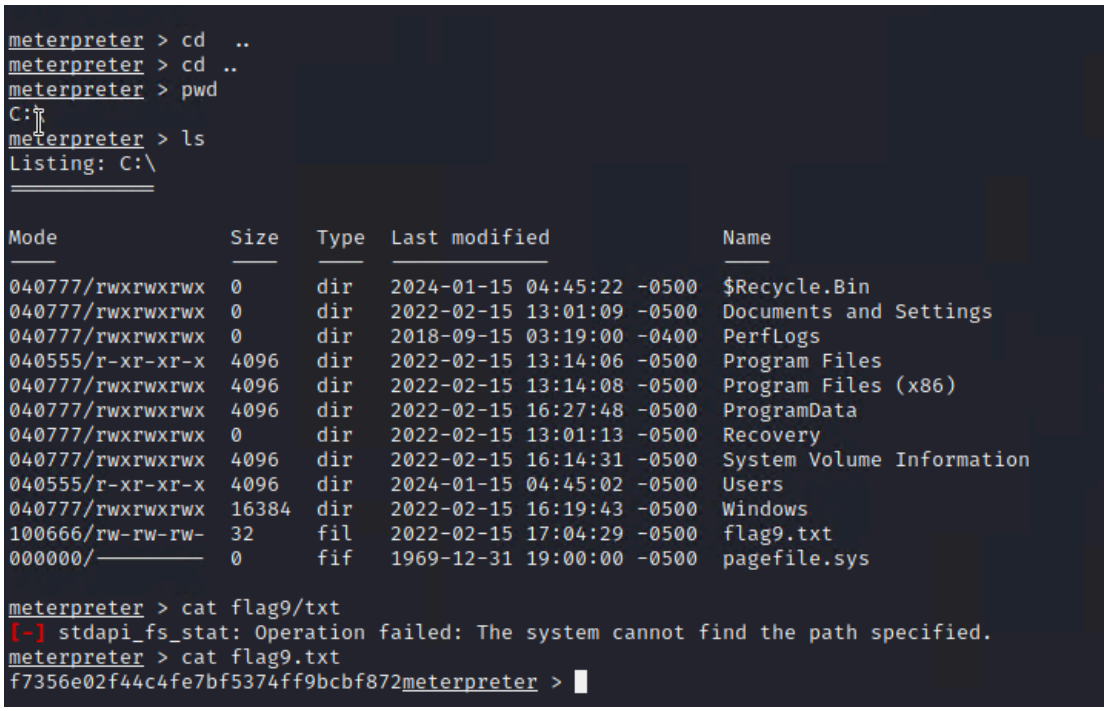
	• Use Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) •

Vulnerability 20	Findings
Title	SSH accessibility with weak password
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	SSH username was found using whois on the domain name. Easily guessed password to SSH into the system as username and password were the same.
Images	
Affected Hosts	192.168.13.14
Remediation	• Change Admin Name from sshUser alice to something less apparent. More complex passwords. Close Port 22

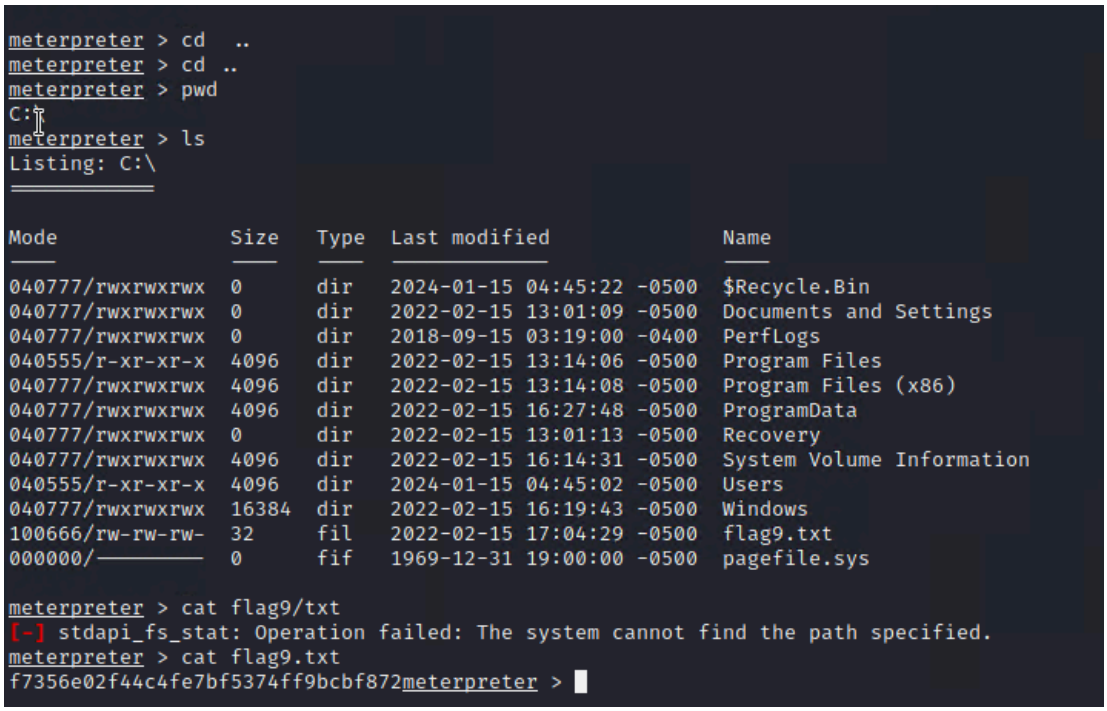
Vulnerability 21	Findings
Title	Privilege Escalation - No password sudo access.
Type (Web app / Linux OS / WIndows OS)	Linux OSi
Risk Rating	critical
Description	Security policy bypass issue that provides a user or a program the ability to execute commands as root on a Linux system when the "sudoers configuration" explicitly disallows the root access. Exploiting the vulnerability requires the user to have sudo

	privileges that allow them to run commands with an arbitrary user ID, except root.Known CVE: CVE-2019-14287
Images	 
Affected Hosts	192.168.13.14
Remediation	• Change sudo access for alice. • To find out if you are vulnerable you will have the sudo version prior to 1.8.28 and run the following command in your terminal to seek for a match: cat /etc/sudoers | grep “(\s*ALL\s*,\s*\!root\s*)” cat /etc/sudoers | grep “(\s*ALL\s*,\s*\!#0\s*)”

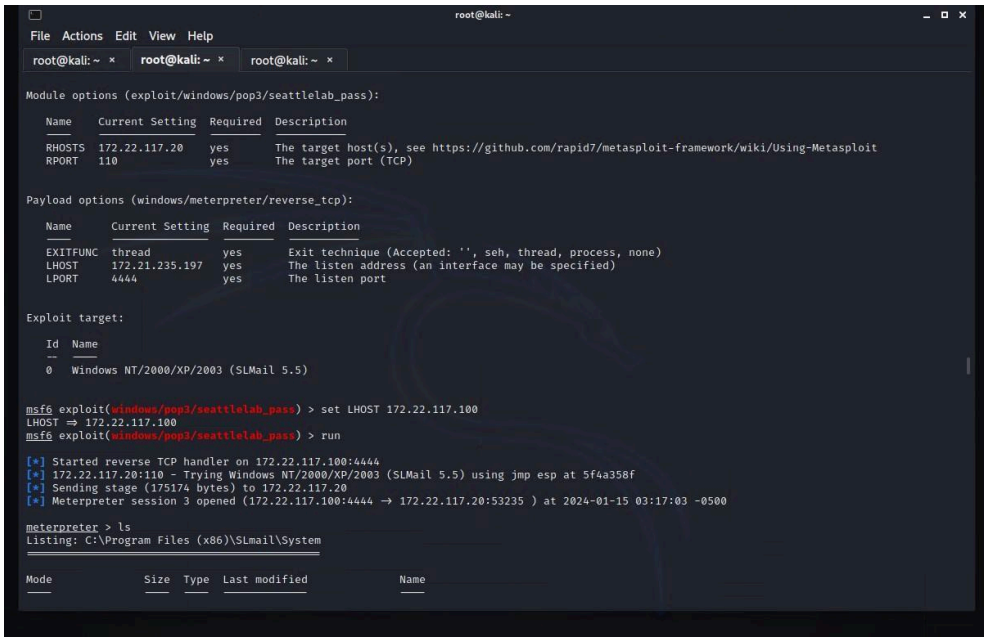
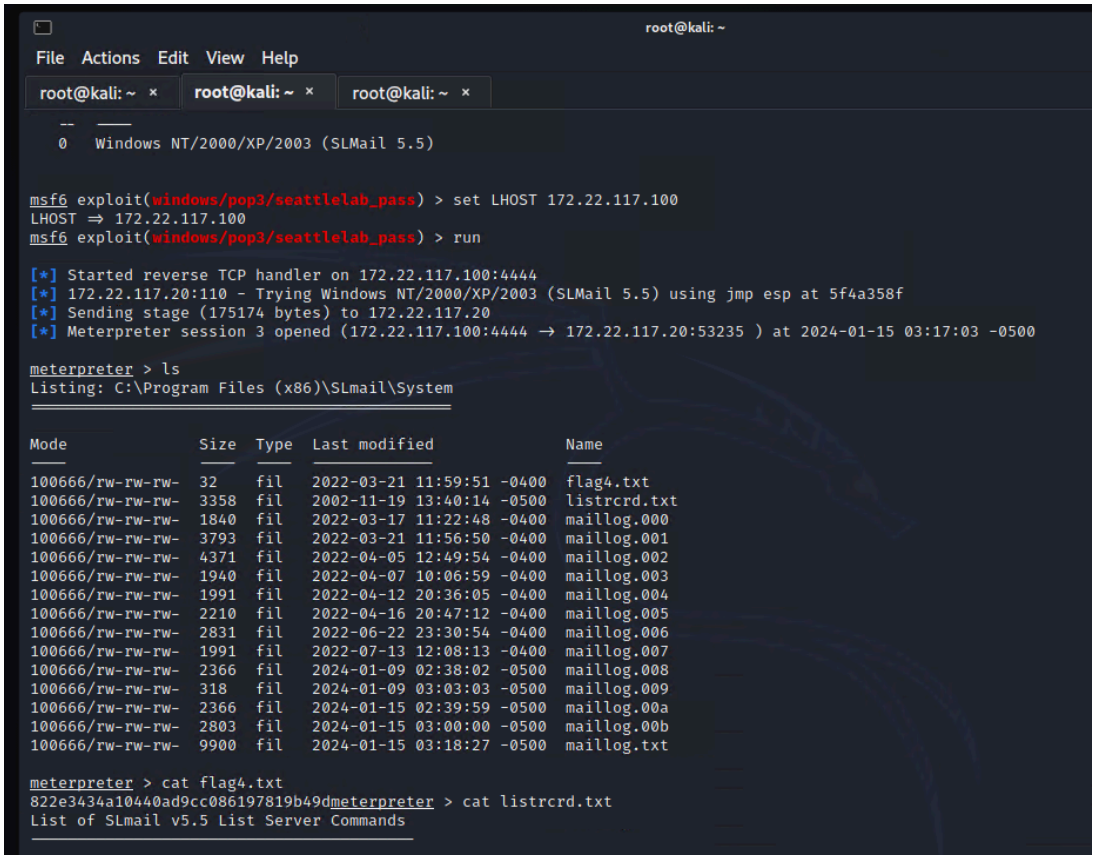
Vulnerability 22	Findings
Title	Cached credentials
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	High
Description	We were able to establish a psexec exploit session through gaining the cached credentials of a administrator user, allowing us the gain root access and look for other users using “net users”

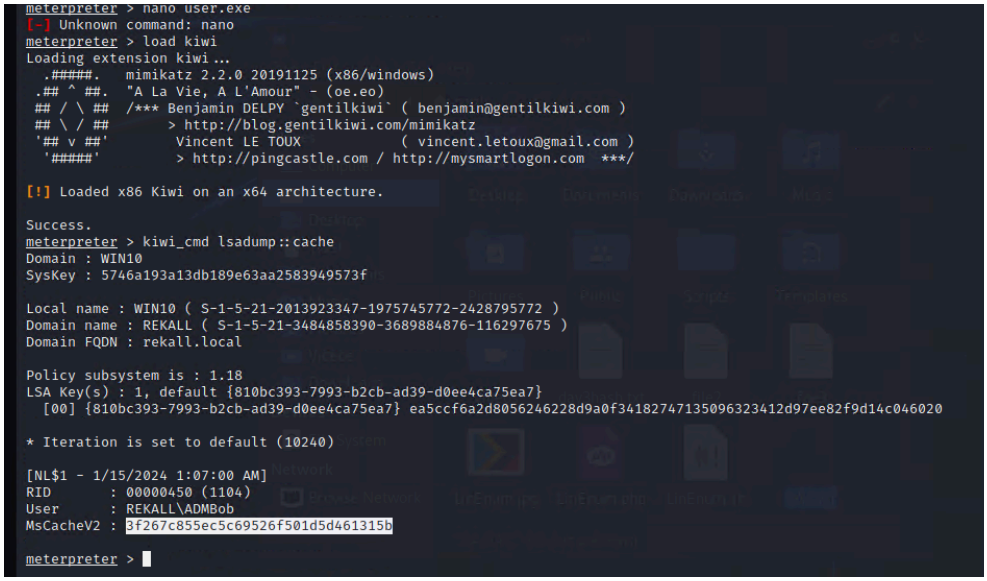
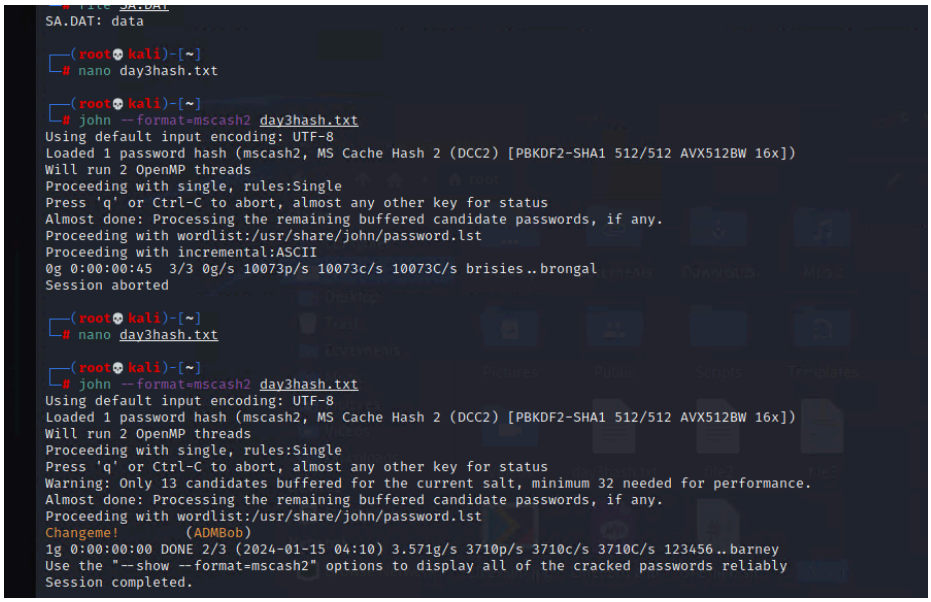
Images	
Affected Hosts	172.22.117.10
Remediation	• Do not store credentials in the cache • Implement MFA

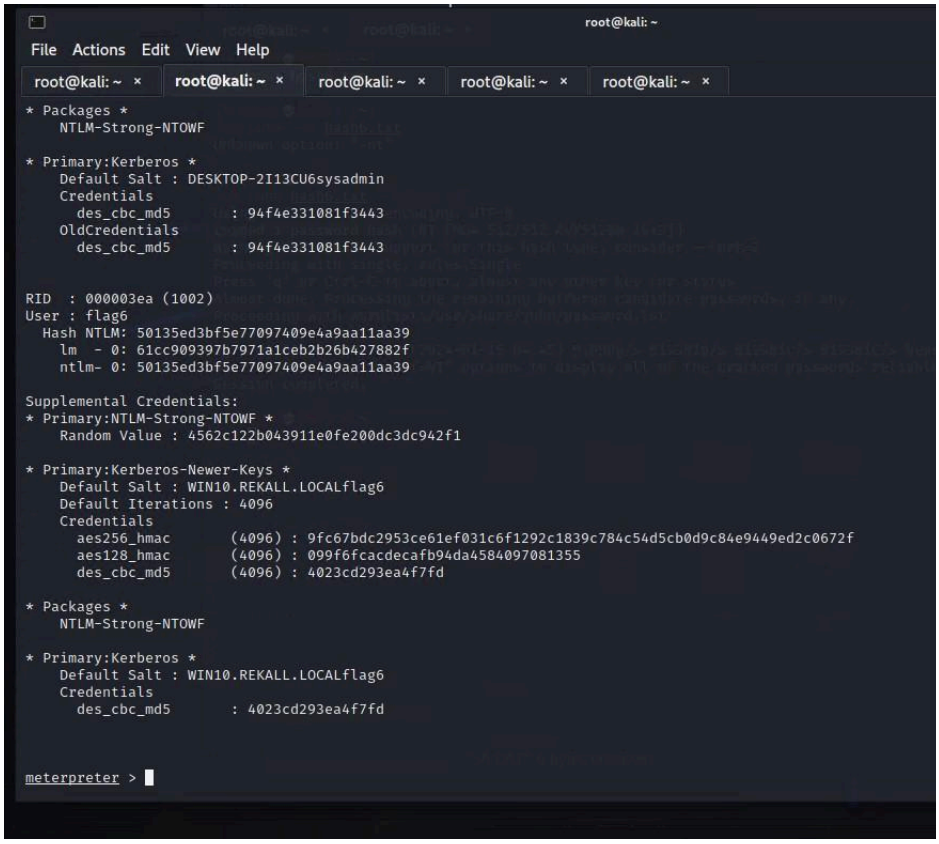
Vulnerability 23	Findings
Title	Protecting sensitive files
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	High
Description	Was able to access the root directory via the previously established meterpreter session

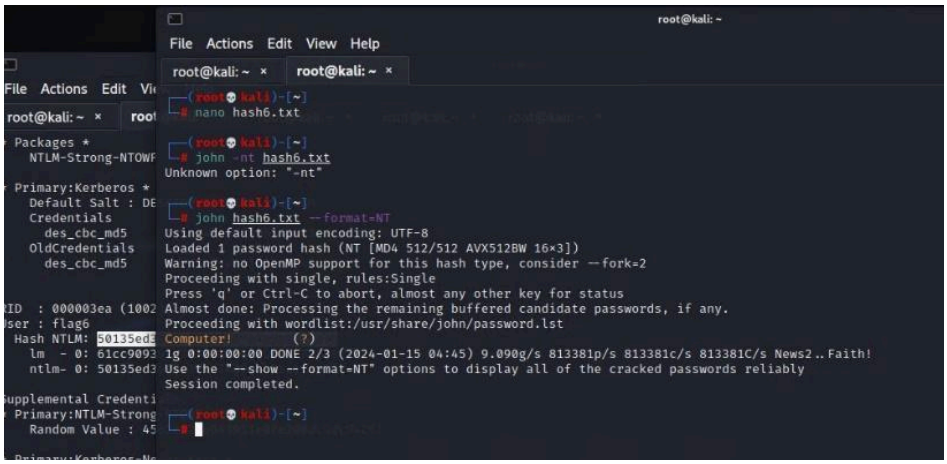
Images	
Affected Hosts	172.22.117.10
Remediation	• Access control regulation • Properly protect sensitive files especially root

Vulnerability 24	Findings
Title	SLMail Vulnerability
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	critical
Description	Using the windows/pop3/seattlelab_pass exploit we were able to open a meterpreter session and view the SLMail system logs. Known CVEs: CVE-2003-0264, CVE-1999-0380, CVE-1999-0231

Images	 
Affected Hosts	172.22.117.20
Remediation	• Restrict password length • Data execution prevention—flags certain areas of memory as non-executable or executable, which stops an attack from running code in a non-executable region. • Use a different Mail service

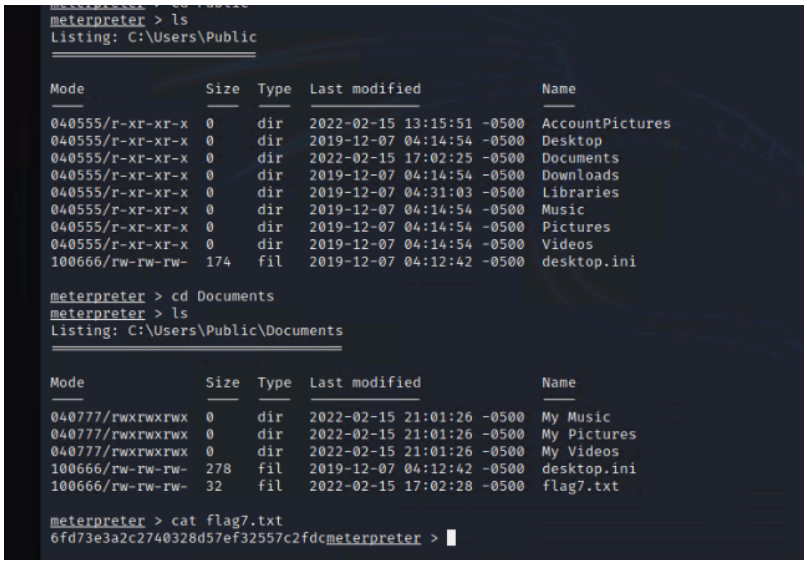
Vulnerability 25	Findings
Title	Credentials Dump
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Critical
Description	Using mimikatz kiwi module lsadump:: cache and sam were able to retrieve password hashes for both ADMBob (administrative account) and user Flag6 (sysadmin account)
Images	 



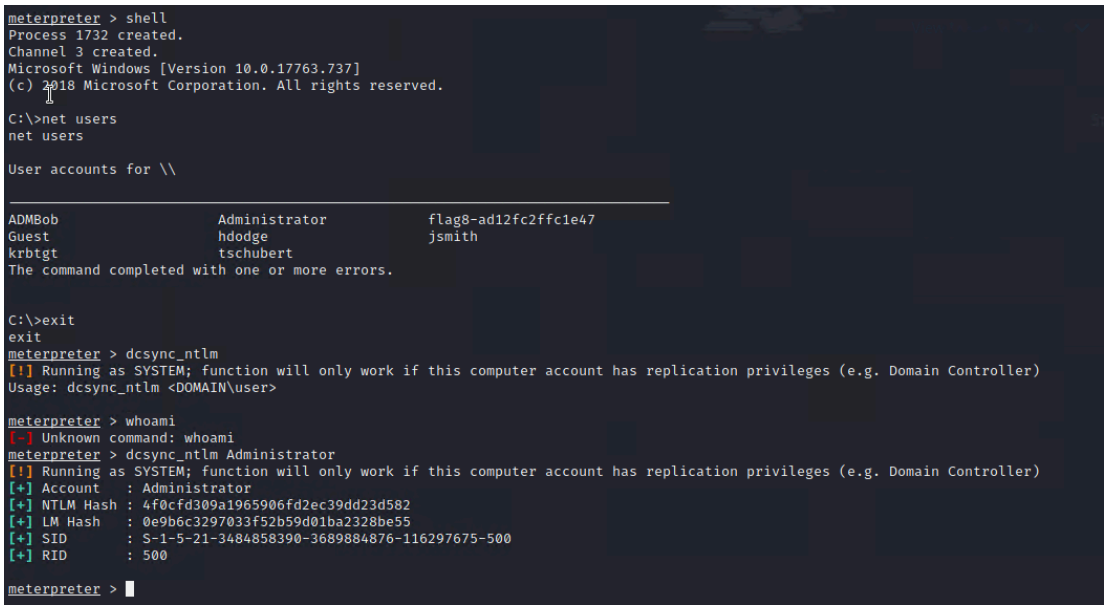


Affected Hosts	172.22.117.20
Remediation	• Regularly update permissions.

Vulnerability 26	Findings
Title	Directory Search

Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Medium
Description	By navigating to the public documents directory we were able to display files
Images	
Affected Hosts	172.22.117.20
Remediation	• Secure sensitive files in non-accessible areas and/or restrict access.

Vulnerability 27	Findings
Title	Domain Controller Impersonation
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	High
Description	Through the previously established meterpreter session i was also able to use a program called ‘Kiwi’ to DCSync the Administrator user in which provided me with the Hashed password for Administrator that could be unhashed through ‘John’

Images	
Affected Hosts	172.22.117.10
Remediation	• Correctly protect Hashed passwords from tools such as ‘kiwi’ and other exploitation tools • Make use of more robust authentication methods • Monitor network traffic to DC controllers • Alert in real time on changes to replication permissions. • Routinely audit the need for replication permissions and aggressively enforce the principle of least privilege. • If a legitimate need for replication permissions exists, adopt compensating controls, such as login restrictions and enhanced auditing, to mitigate the risk of credential theft. • Do not allow users to possess administrative privileges across security boundaries. This greatly reduces the ability of an adversary to escalate their privileges.