Cybersecurity

Penetration Test Report

Rekall Corporation

Penetration Test Report

Jray Pensters, LLC

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	Jray Pensters, LLC
Contact Name	Josh Ryan
Contact Title	Penetration Tester

Document History

Version	Date	Author(s)	Comments
001	10/11/2022	Joshua Ryan	(just one copy - no revisions)

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.

IP Address/URL	Description
192.168.14.0/24 and 192.168.13.0/24 and 34.102.136.180(totalrekall.xyz) and 172.22.117.0/24	Rekall's internal domain, range and public website. As well as their .xyz page, Kali/Windows machines, and Linux servers.

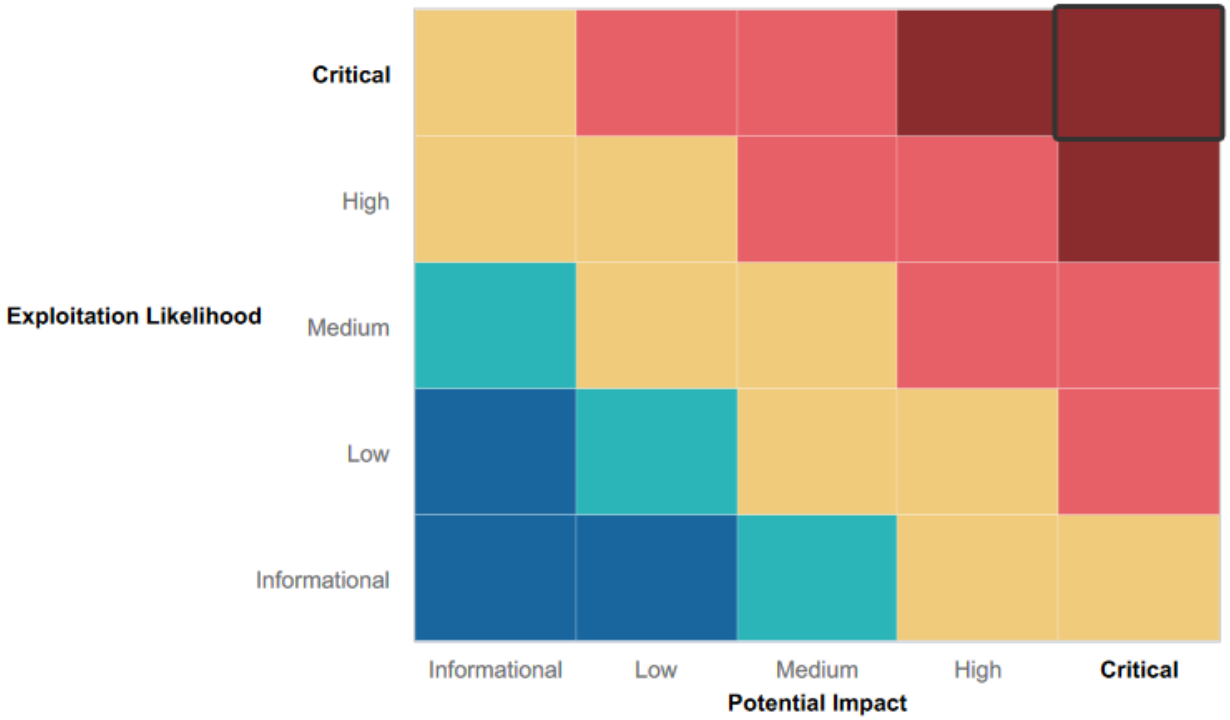
Executive Summary of Findings

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.

- All ports were not open on every IP address. (i.e. 192.168.13.0/24)

```
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-17 21:42 EDT
Nmap scan report for 192.168.13.10
Host is up (0.000010s latency).
Not shown: 998 closed tcp ports (reset)
PORT     STATE SERVICE
8009/tcp open  ajp13
8080/tcp open  http-proxy
MAC Address: 02:42:C0:A8:0D:0A (Unknown)

Nmap scan report for 192.168.13.11
Host is up (0.000010s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE
80/tcp open  http
MAC Address: 02:42:C0:A8:0D:0B (Unknown)

Nmap scan report for 192.168.13.12
Host is up (0.000010s latency).
Not shown: 999 closed tcp ports (reset)
PORT     STATE SERVICE
8080/tcp open  http-proxy
MAC Address: 02:42:C0:A8:0D:0C (Unknown)

Nmap scan report for 192.168.13.13
Host is up (0.000010s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE
80/tcp open  http
MAC Address: 02:42:C0:A8:0D:0D (Unknown)

Nmap scan report for 192.168.13.14
Host is up (0.000010s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
MAC Address: 02:42:C0:A8:0D:0E (Unknown)

Nmap scan report for 192.168.13.1
Host is up (0.0000090s latency).
Not shown: 996 closed tcp ports (reset)
PORT      STATE    SERVICE
5901/tcp  open     vnc-1
6001/tcp  open     X11:1
10000/tcp filtered snet-sensor-mgmt
10001/tcp filtered scp-config
```

- The comments.php page did block the word "script" so SQL injection was harder to bypass.
- When uploading the picture we had to obfuscate it by making it a .jpg.php document type when it was really a .php file created in nano. The image input only took certain file types which was a good security measure.

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.

- The website itself was vulnerable to several XXS and SQL injections
- Sensitive data exposure

- Local file inclusion
- Sensitive data exposure
- Command Injection
- Brute Force Attack
- PHP Injection
- Shellshock
- Directory Traversal
- Open source exposed data
- Scanning Vulnerability
- Pivoting/lateral movement
- Apache struts CVE-2017-5638
- Drupal - CVE-2019-6340
- Privilege-escalation/SSH - CVE-2019-14287
- HTTP Enumeration
- FTP Enumeration
- Metasploit on SLMail service
- Task managers exploit
- Kiwi - LSA dump
- NTLM password cracking

Executive Summary

Jray Pentesters, LLC conducted an extensive security assessment of Rekall's: Web servers, Linux servers and Windows servers to see what exploits and risks they are vulnerable to. We use various penetration testing techniques targeted on this corporation's infrastructure. We started our recon phase by exploiting the web pages vulnerabilities.

The first test of the web app showed, via the website, different cross-site exploits that were used successfully across the different input fields. We also used OSINT tools to enumerate the operating systems, software and processes that the system used. There were other extensive vulnerabilities on the web page including but not limited to: Local file inclusion, SQL injection, command injection and PHP injection.

We moved to attacking the Linux servers on Day 2 and found the IPs and open ports to run on that day. Some exposures found here included several metasploit modules to pivot around the system to gain shell access to run commands and enumerate credentials. We were able to ssh into Alice's user account.

On the third day of testing, we focused on the Windows machines and used tactics such as: Brute force password cracking, HTTP Enumeration, FTP Enumeration, Kiwi/LSA Dump, File Enumeration and NTLM password cracking. All these as a means to escalate our access and schedule tasks to remain in the network with our backdoor. Below you will find more specifics of these particular exploits and the severity rating given to each. Ultimately Jray Pentesters, LLC determined that Rekalls data isn't safe against many known vulnerabilities and should take some immediate steps to fix the findings found during our examination.

Summary Vulnerability Overview

Vulnerability	Severity
DAY 1 (WEB APP)	
XXS - Cross-site scripting	Critical
XSS reflected (advanced)	Critical
XSS Stored	Critical
Sensitive data exposure	Medium
Local file inclusion	Critical
Local file inclusion (advanced)	High
SQL injection	Critical
Sensitive data exposure	Critical
Sensitive data exposure	Critical
Command injection	High
Command injection (advanced)	Critical
Brute force attack	Critical
PHP injection	High
Session management	Medium
Directory traversal	High
DAY 2 (LINUX OS)	
Open source exposed data	High
Scanning Vulnerability	Medium
Open source exposed data	High
Scanning Vulnerability	Medium
Aggressive Scan Vulnerability	High
Scanning Vulnerability (Nessus)	Critical
Apache Tomcat Remote Code Execution Vulnerability (CVE-2017-12617)	High
Shellshock	High
Pivoting/lateral movement	High
Apache struts CVE-2017-5638	Medium
Drupal - CVE-2019-6340	Medium
Privilege-escalation/SSH - CVE-2019-14287	High
DAY 3 - (Windows OS)	
Brute force attack - Password Cracking	High
HTTP Enumeration - port scanning	Medium
FTP Enumeration	Critical
Metasploit on SLMail service	High
Task managers exploit (using shell)	High

Kiwi - LSA dump (brute force)	High
File Enumeration	Medium
User Enumeration - (brute force)	High
Escalating access	Medium
NTLM password cracking - compromise Admin	High

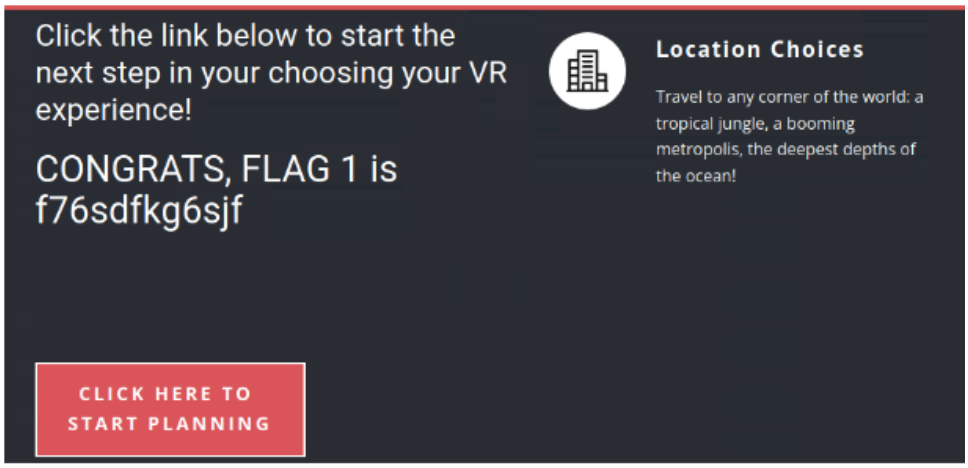
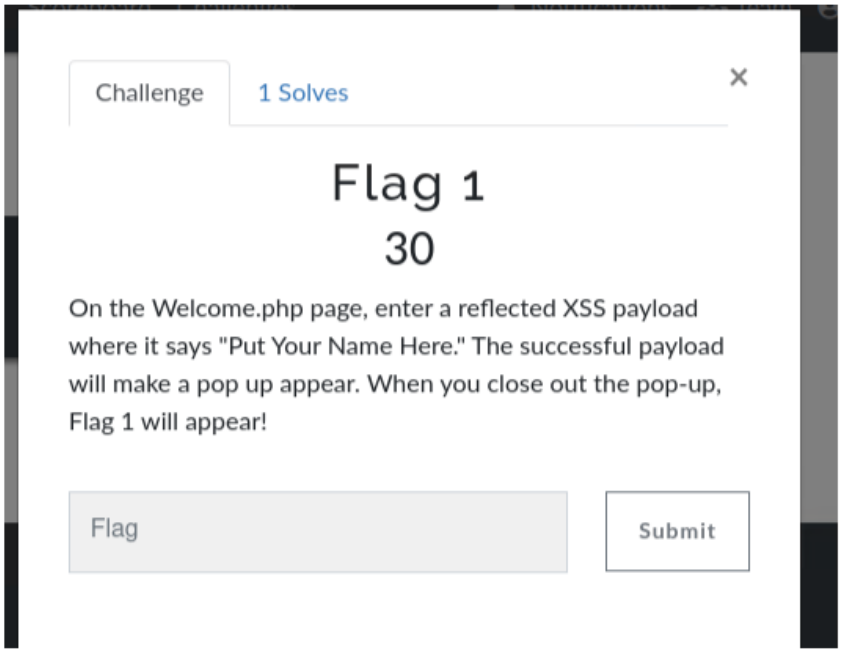
The following summary tables represent an overview of the assessment findings for this penetration test:

Scan Type	Total
Hosts	192.168.14.0/24 and 192.168.13.0/24 and 34.102.136.180 (totalrekall.xyz) and 172.22.117.0/24
Ports	8080, 443, 21, 22, 25, 79, 80, 106, 110, 135, 139, 4444, 34048, 34060, 51164, 58874,

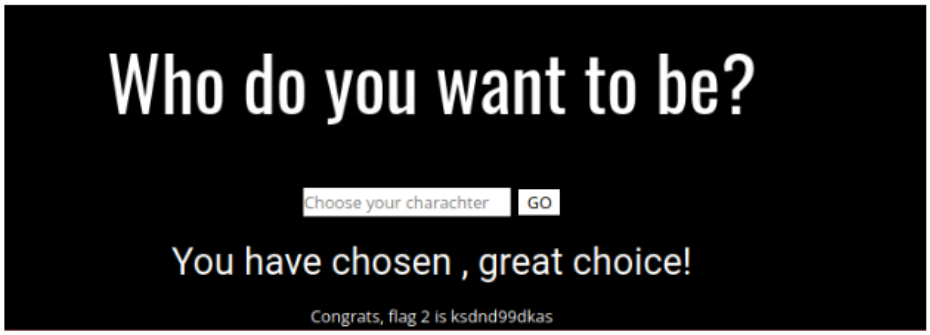
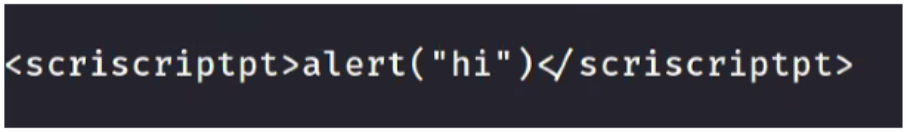
Exploitation Risk	Total
Critical	11
High	17
Medium	9
Low	0

Vulnerability Findings

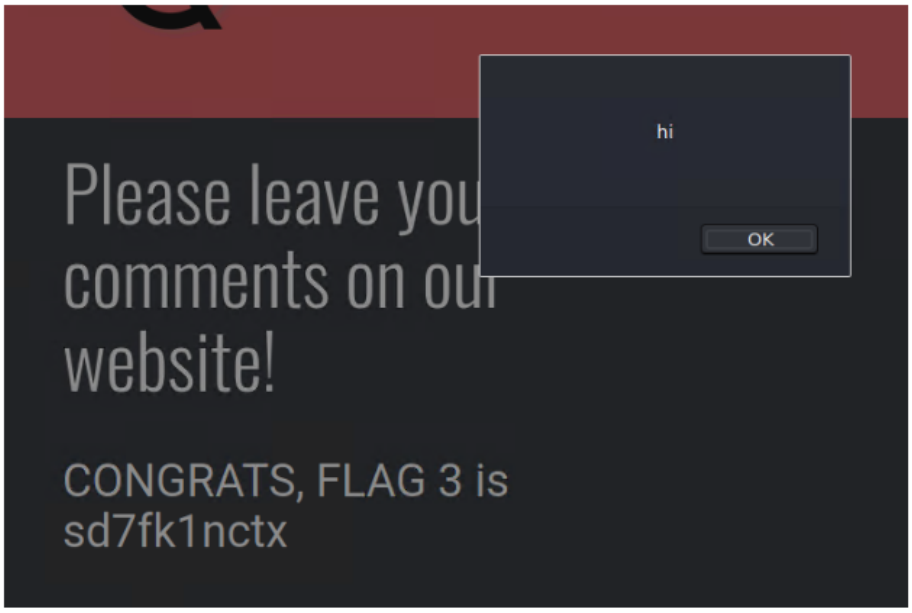
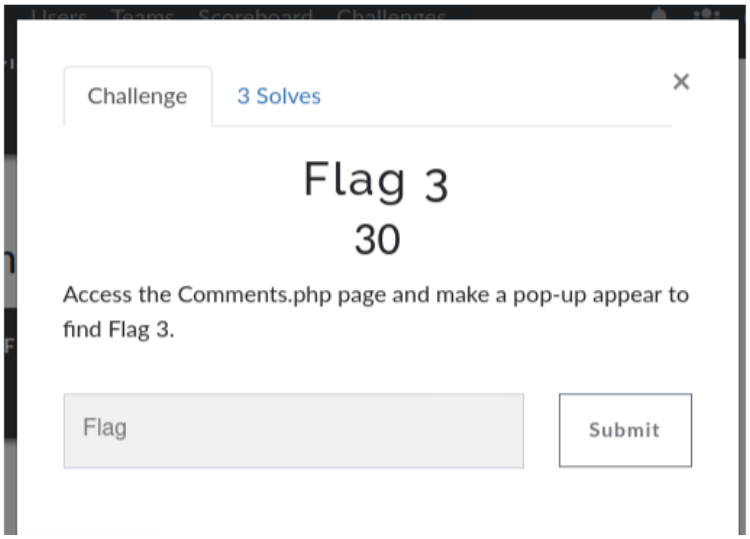
Vulnerability 1	Findings
Title	XXS - Cross-site scripting
Type (Web app / Linux OS / Windows OS)	Web app
Risk Rating	Critical
Description	Used a XXS payload under the “put your name here” field to cause a pop up.

Images	 
Affected Hosts	192.168.14.35
Remediation	Have a firewall blocking the word script from the enter field.

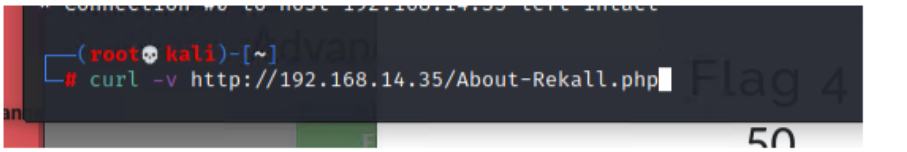
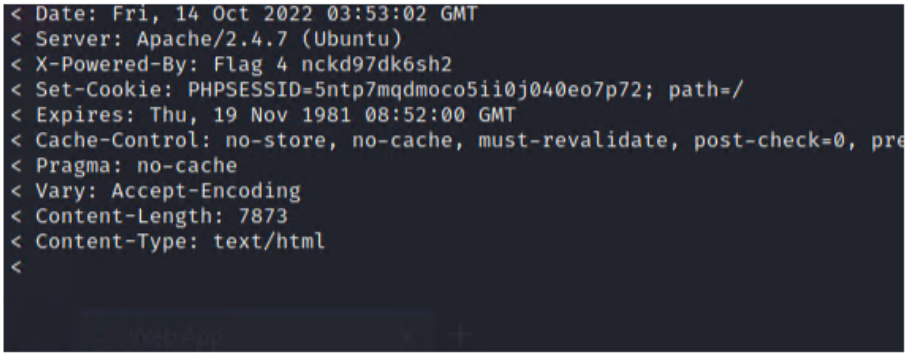
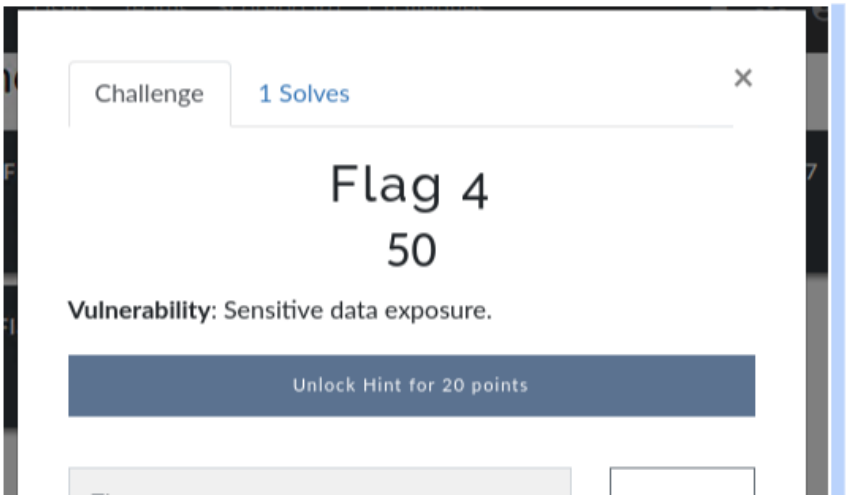
Vulnerability 2	Findings
Title	XSS reflected (advanced)
Type (Web app / Linux OS / Windows OS)	Web app
Risk Rating	**Critical**
Description	The input validation removes the word "script," so the word "script" needs to be split up in the payload—for example: <SCRIPscriptT>alert("hi")</SCRIPscripTt>. Causing another pop up.

Images	 
Affected Hosts	192.168.14.35
Remediation	Possibly have the input field also just block all carrots <> or other special characters.

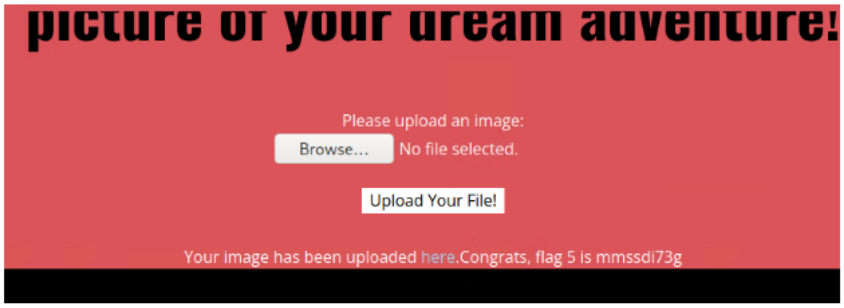
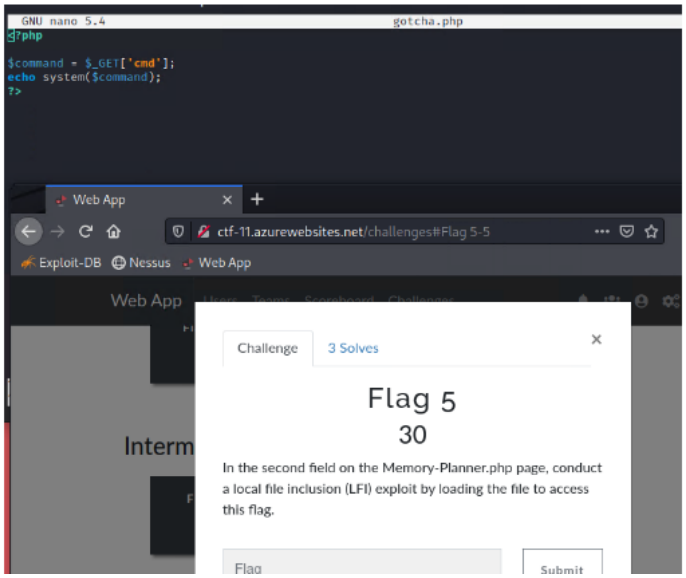
Vulnerability 3	Findings
Title	XSS Stored
Type (Web app / Linux OS / Windows OS)	Web app
Risk Rating	**Critical**
Description	Enter java script into comment.php box

Images	flag 3: DONE  
Affected Hosts	192.168.14.35
Remediation	Block scripts from the input field. Block the carrots as above and key words like script.

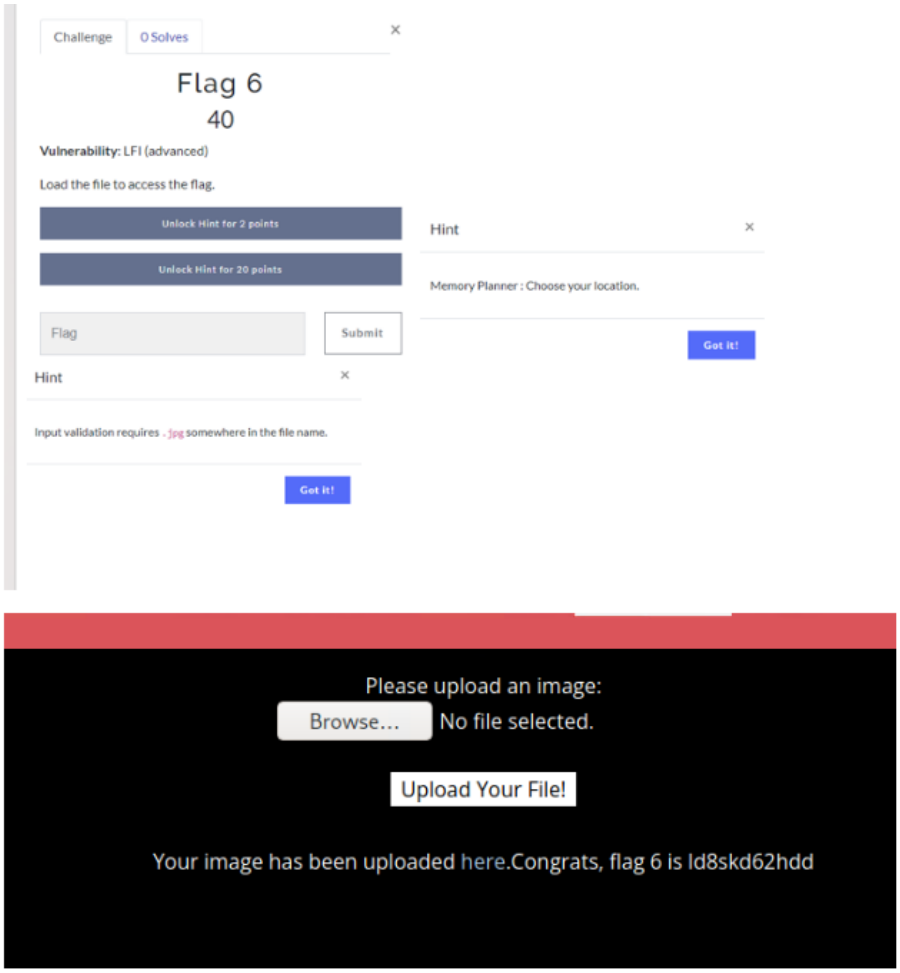
Vulnerability 4	Findings
Title	Sensitive data exposure
Type (Web app / Linux OS / Windows OS)	Web app
Risk Rating	**Medium**

Description	The flag appears in the HTTP response headers. These headers can be seen using BURP or via a cURL request, such as: ○ curl -v http://192.168.14.35/About-Rekall.php
Images	  
Affected Hosts	192.168.14.35
Remediation	This would be difficult to remediate the actual command. Try to secure the webpage better with https and use data segregation better to not leave these documents out in the open.

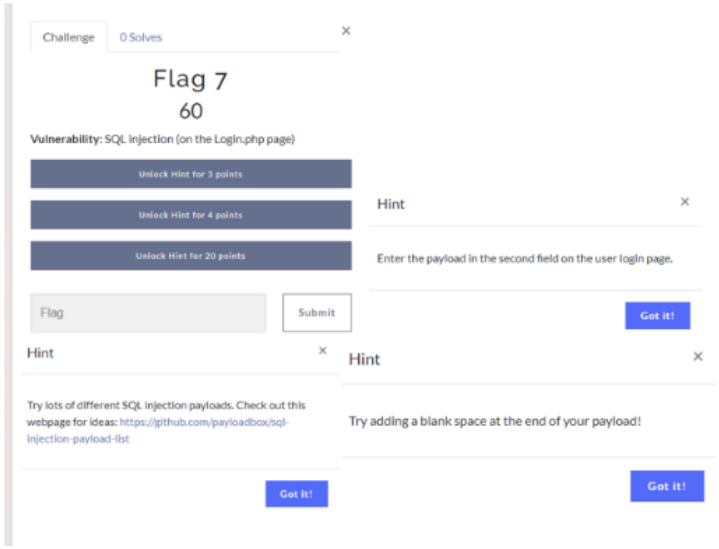
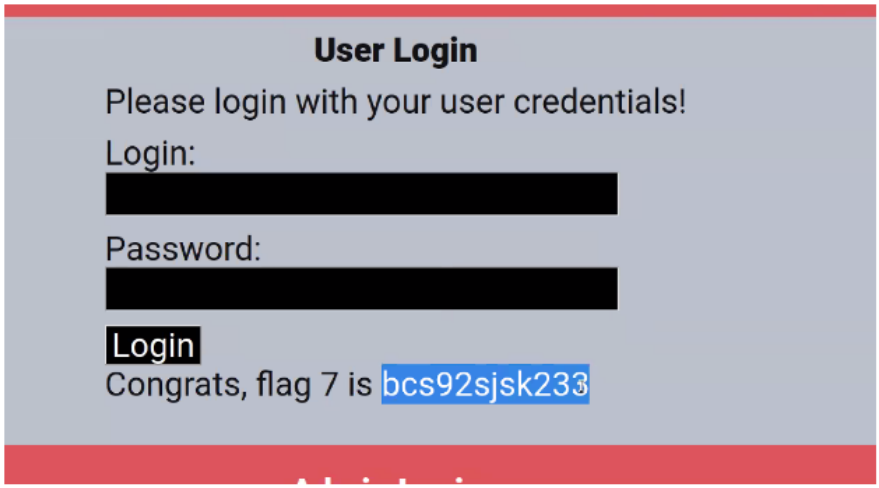
Vulnerability 5	Findings
Title	Local file inclusion
Type (Web app / Linux OS / Windows OS)	Web app

Risk Rating	**Critical**
Description	Uploading an exploit as a php file
Images	 
Affected Hosts	192.168.14.35
Remediation	Better intrusion prevention systems to block certain file types and scan these before they are attached to the server. To safely parse user-supplied file names it's much better to maintain a whitelist of acceptable filenames and use a corresponding identifier (not the actual name) to access the file. Any request containing an invalid identifier can then simply be rejected. This is the approach that OWASP recommends.

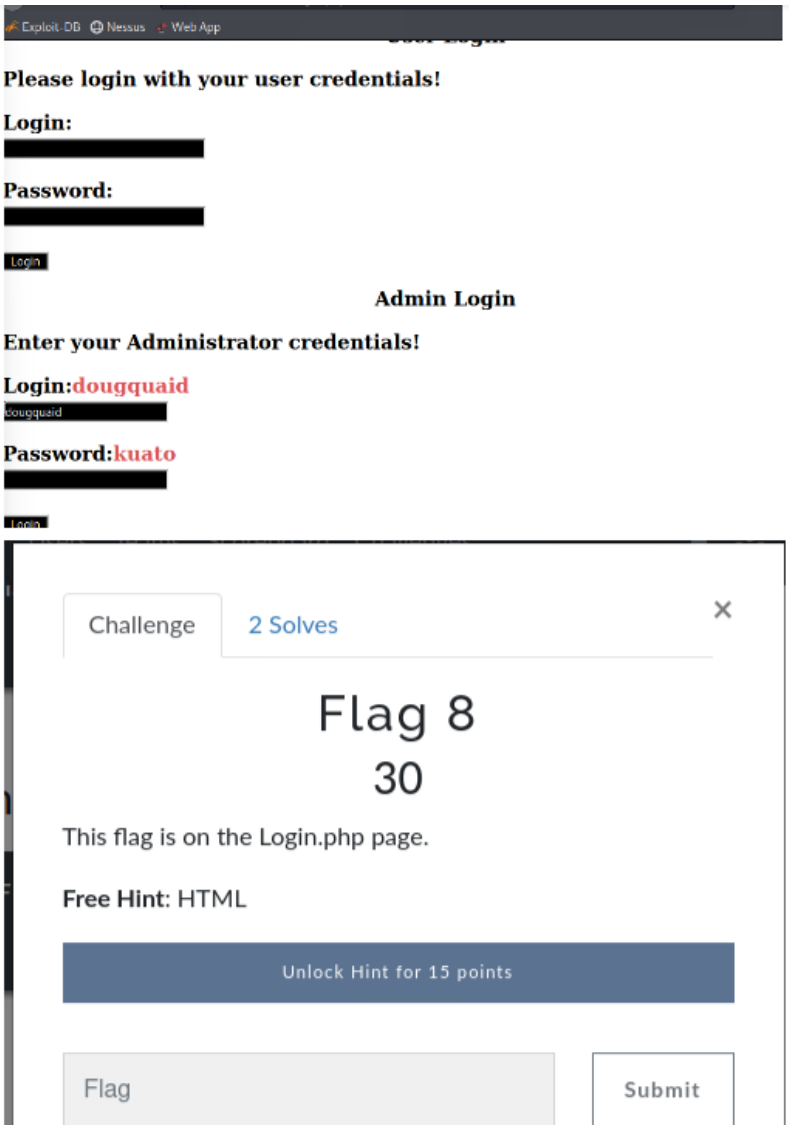
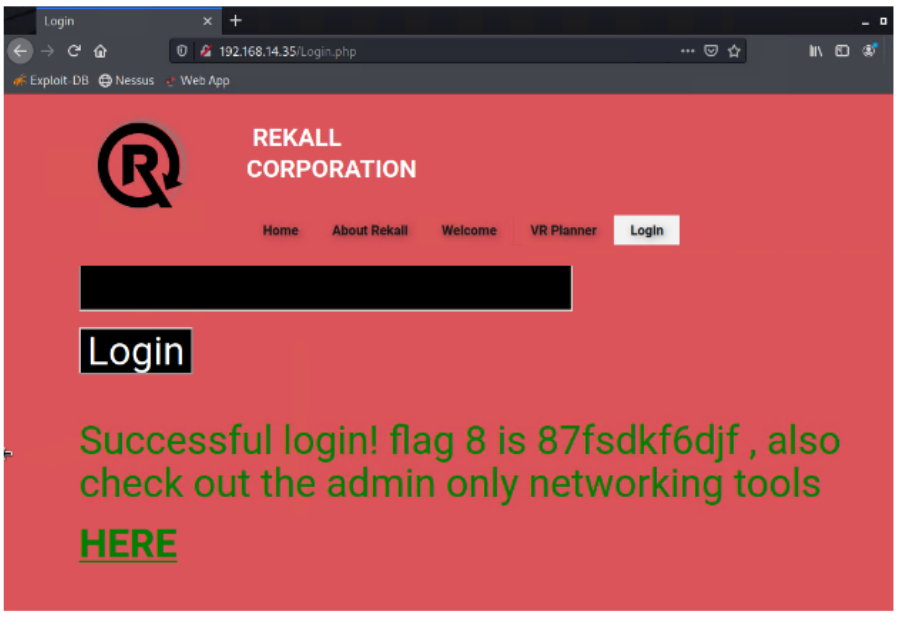
Vulnerability 6	Findings
Title	Local file inclusion (advanced)
Type (Web app / Linux OS / Windows OS)	Web app
Risk Rating	**High**
Description	The input validation checks for the presence of .jpg, so to bypass this upload, name your malicious script with this name: script.jpg.php

Images	
Affected Hosts	192.168.14.35
Remediation	Block the .php from being used anywhere in the input field. Have a better whitelist with possible obfuscation choices. IDS and IPS that detect this.

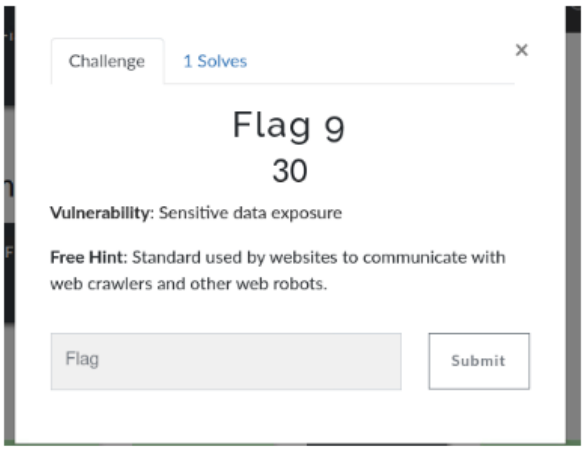
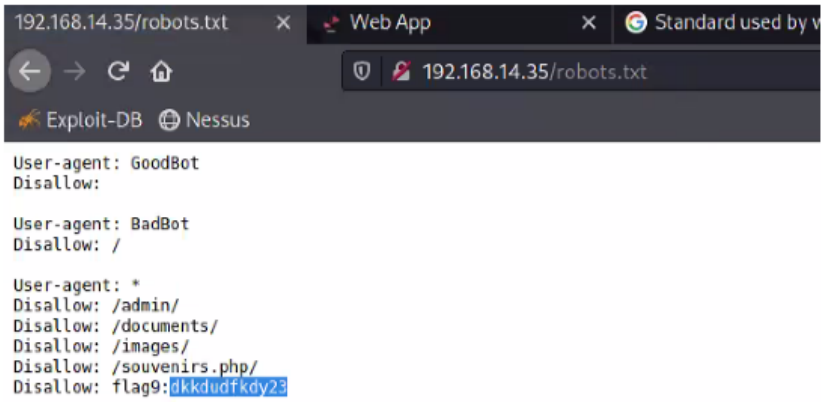
Vulnerability 7	Findings
Title	SQL injection
Type (Web app / Linux OS / Windows OS)	Web app
Risk Rating	**Critical**
Description	In the password field, use the following payload: ok' or 1=1-- (our sql found during our OSINT research)

Images	 
Affected Hosts	192.168.14.35
Remediation	Testing of open source SQL injections and blocking certain known vulnerabilities. Only allow a password to be entered with a valid user log in.

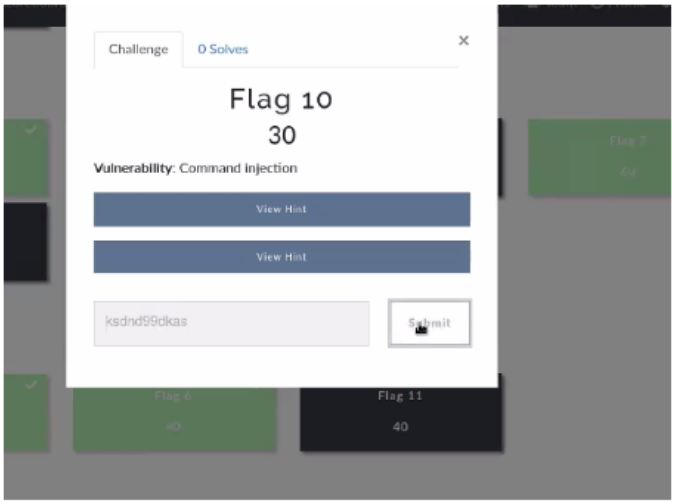
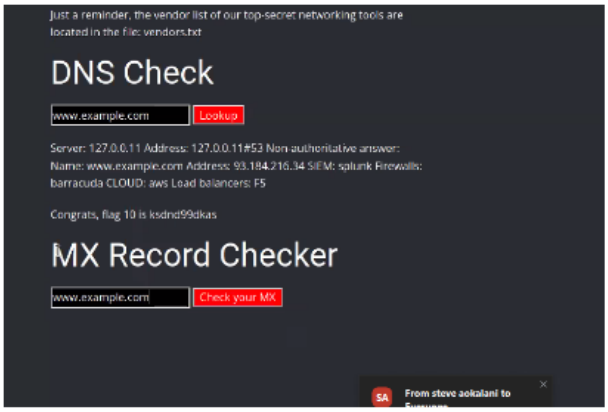
Vulnerability 8	Findings
Title	Sensitive data exposure
Type (Web app / Linux OS / Windows OS)	Web app
Risk Rating	**Critical**
Description	The username and password are in the HTML, or you can view them by highlighting the webpage. Typed the file name plainly into the url. Used the user creds below. Username: dougquaid Password: kuato

Images	 
Affected Hosts	192.168.14.35

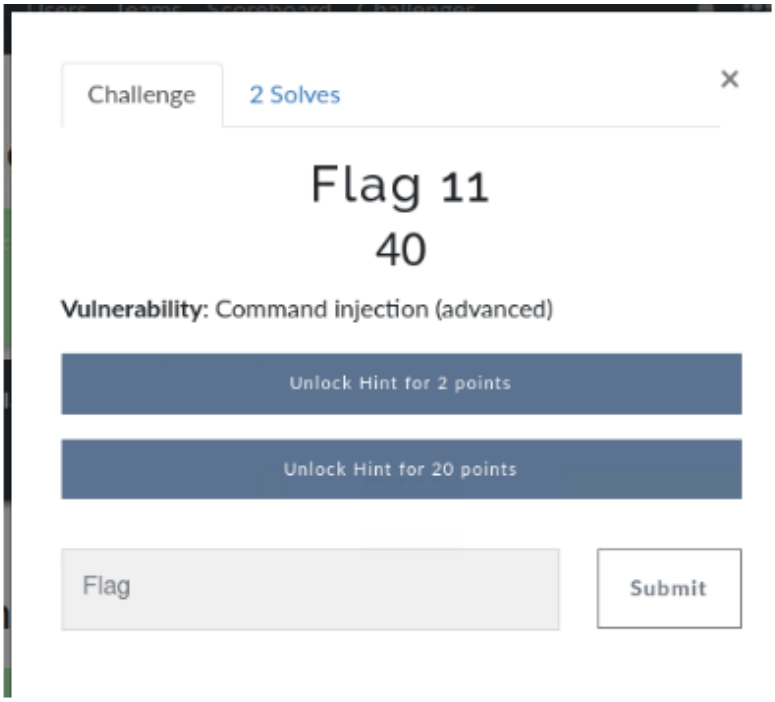
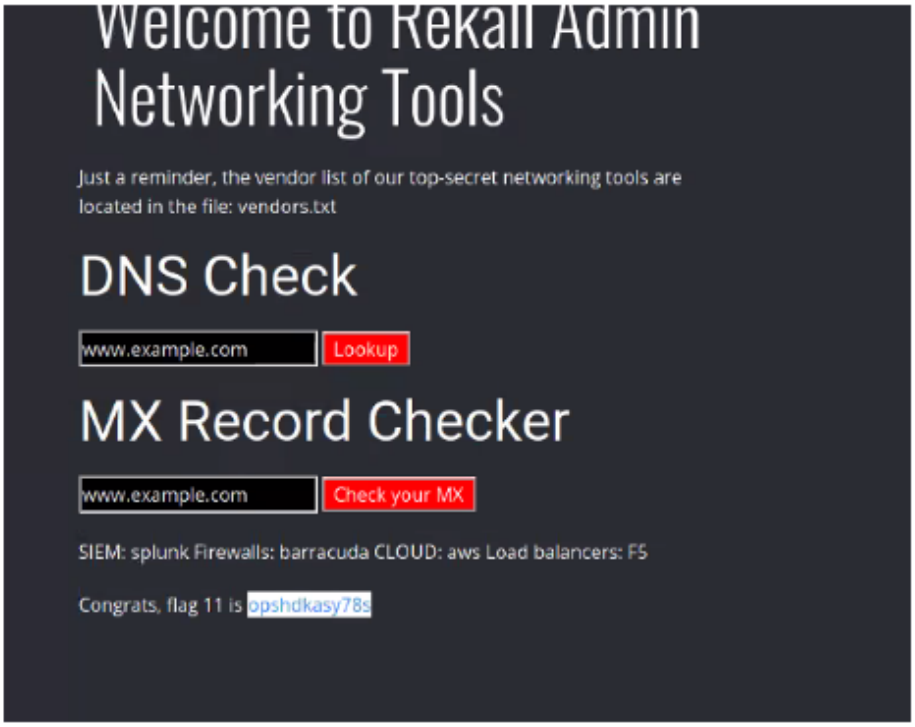
Remediation	Simply don't have any credentials stored anywhere on the web page itself. Also, encrypt this webpage with https so the site is more secure.

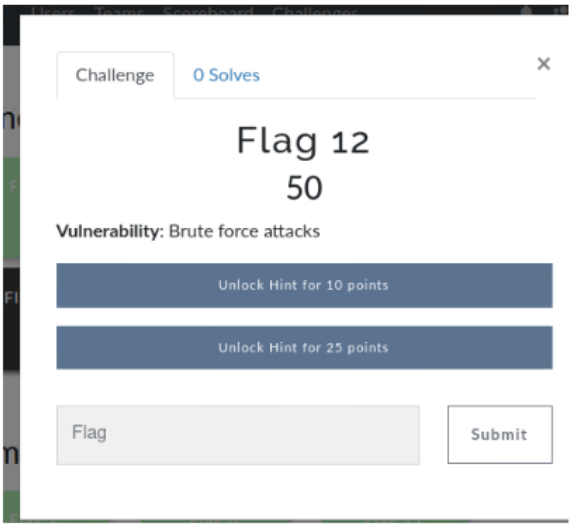
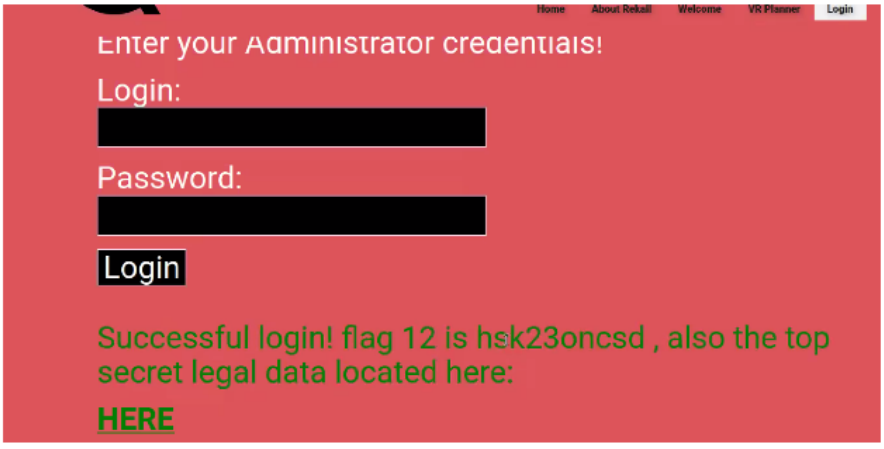
Vulnerability 9	Findings
Title	Sensitive data exposure
Type (Web app / Linux OS / Windows OS)	Web app
Risk Rating	**Critical**
Description	Just access the webpage with a simple url injection of the robot.txt file path.
Images	 
Affected Hosts	192.168.14.35
Remediation	If this site was secured with https we wouldn't be able to just see the text file in plain text. Do not store any valuable information in these web page file paths.

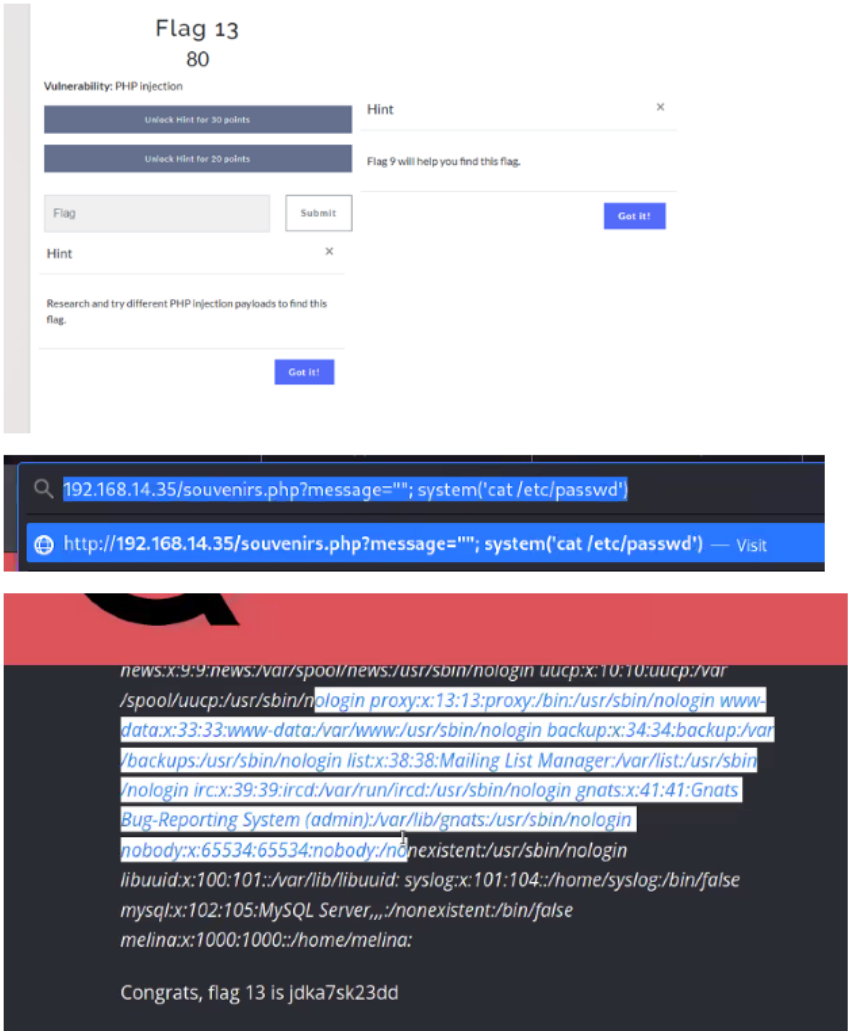
Vulnerability 10	Findings
Title	Command injection

Type (Web app / Linux OS / Windows OS)	Web app
Risk Rating	**High**
Description	Injecting commands into the networking.php path input field. i.e.: www.welcometorecall.com && cat vendors.txt or www.welcometorecall.com ; cat vendors.txt
Images	 
Affected Hosts	192.168.14.35
Remediation	You could block the “&&” from adding command line code into any comment box. Update and patch system regularly. Use Strong Input Validation for Input Passed into Commands and only allow certain characters on the whitelist.

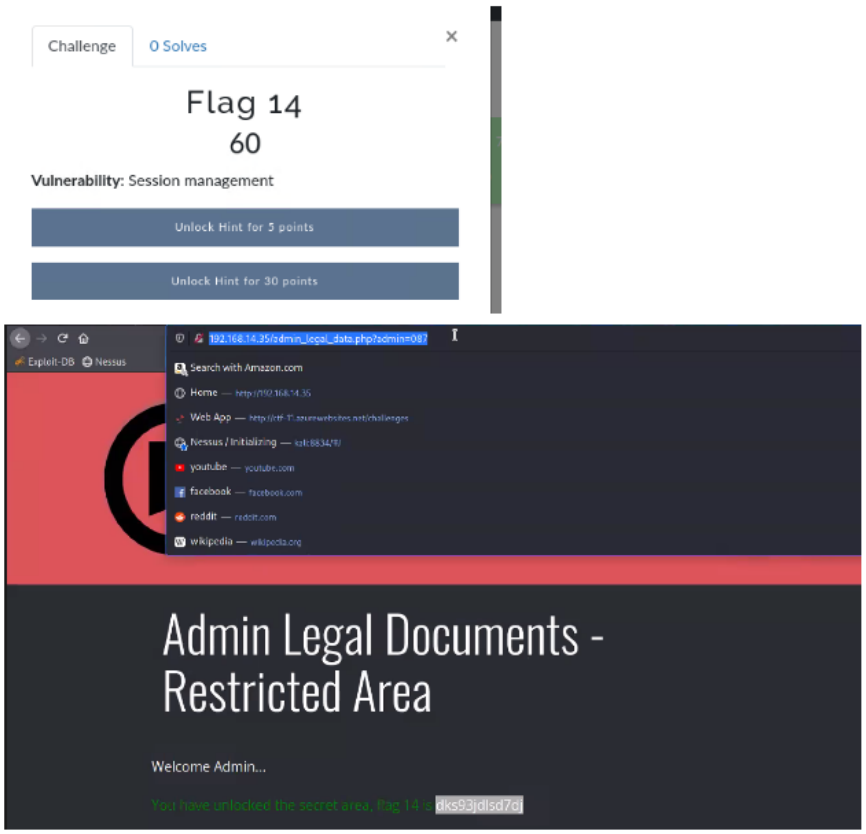
Vulnerability 11	**Findings**
Title	Command injection (advanced)
Type (Web app / Linux OS / Windows OS)	Web app

Risk Rating	**Critical**
Description	Another command injection using: Input validation strips "&" and ";", so the payload will need to be www.welcometorecall.com | cat vendors.txt. This was against the second input field.
Images	 
Affected Hosts	192.168.14.35
Remediation	You could block the “&&” from adding command line code into any comment box. Update and patch system regularly. Use Strong Input Validation for Input Passed into Commands and only allow certain characters on the whitelist.

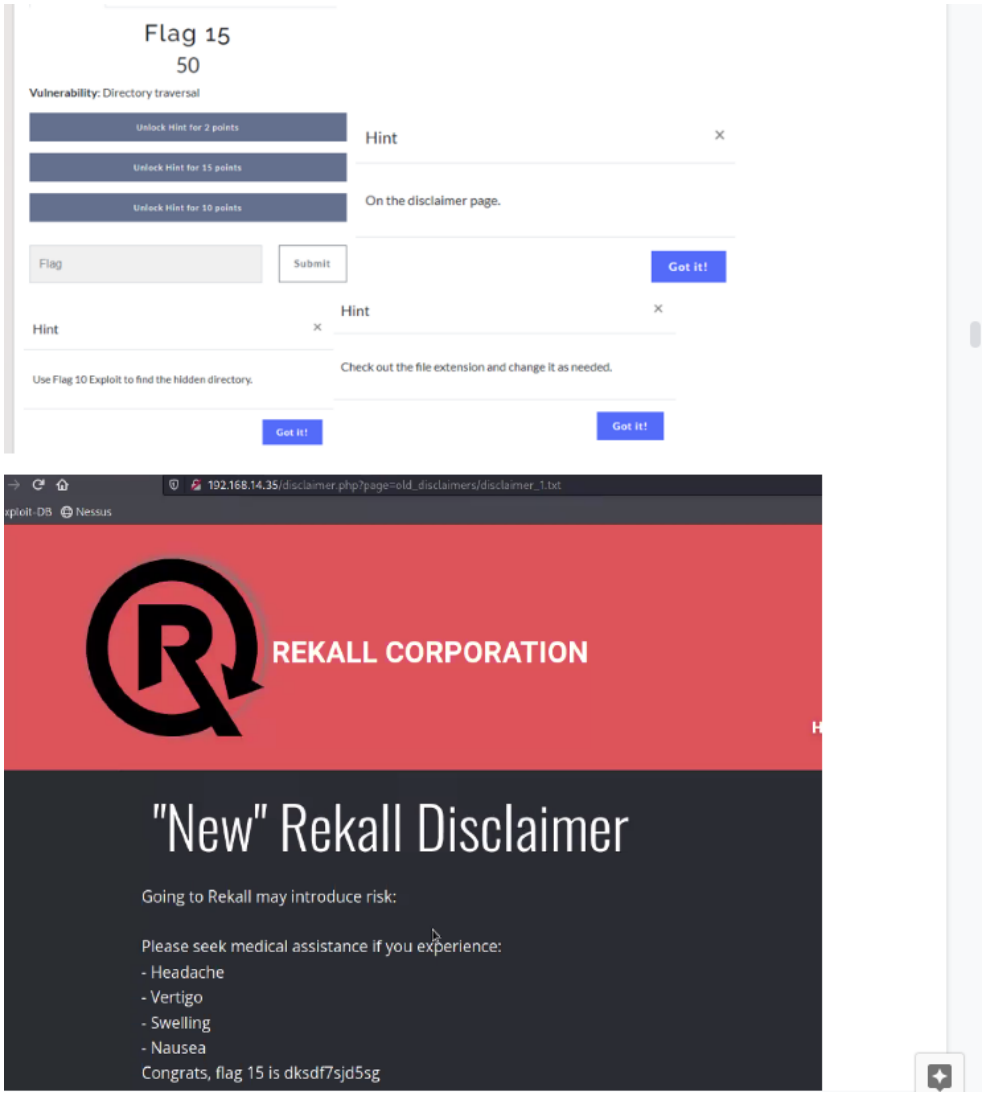
Vulnerability 12	Findings
Title	Brute force attack
Type (Web app / Linux OS / Windows OS)	Web app
Risk Rating	**Critical**
Description	Using the vulnerability in Flag 10 or 11 and viewing the /etc/passwd file, you'll see a user melina. This user has the same password: melina. Use the command inject to move around laterally to find the credentials for melina in the etc/shadow and etc/passwd folders.
Images	 
Affected Hosts	192.168.14.35
Remediation	Use Strong Input Validation for Input Passed into Commands and only allow certain characters on the whitelist. You would need to mitigate by blocking the command to be run through the website at all points. Secure the website with stronger encryption.

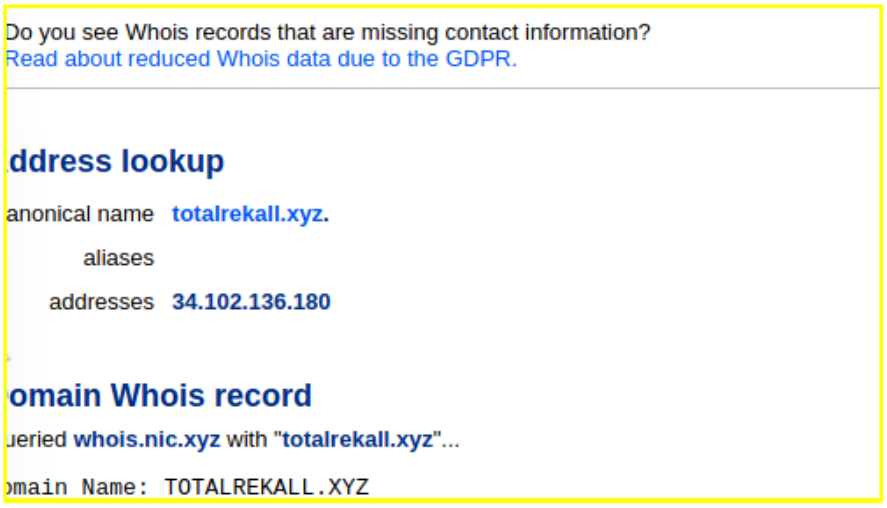
Vulnerability 13	Findings
Title	PHP injection
Type (Web app / Linux OS / Windows OS)	Web app
Risk Rating	**High**
Description	This hidden webpage was identified in the robots.txt file found in Flag 9. The payload to exploit this page is changing the URL to: http://192.168.13.35/souvenirs.php?message=""; system('cat /etc/passwd') OR http://192.168.13.35/souvenirs.php?message=%22%22;%20passthru(%27cat%20/etc/passwd%27)
Images	
Affected Hosts	192.168.14.35
Remediation	In general, it is a good idea to avoid any commands that call the operating environment directly from PHP. Bright Security Dynamic Application Security Testing (DAST) helps automate the detection and remediation of many vulnerabilities including PHP code injection, early in the development process,

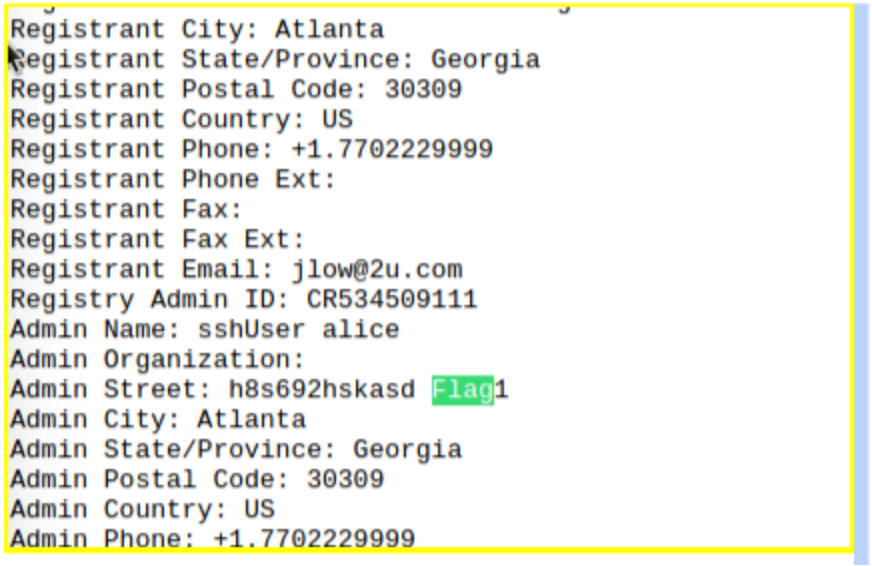
	across web applications and APIs.

Vulnerability 14	Findings
Title	Session management
Type (Web app / Linux OS / Windows OS)	Web app
Risk Rating	**Medium**
Description	The link to this page is provided when Flag 12 is acquired. To view the flag, you will need to test out different session IDs in the URL with Burp. (Intruder would be the most efficient.) 87 is the secret session ID that provides the flag (http://192.168.13.35/admin_legal_data.php?admin=87).
Images	
Affected Hosts	192.168.14.35
Remediation	Use an up-to-date web-server framework to generate and manage the session identifier token, as this will guarantee values that defy prediction. Train developers and IT staff how to hack applications and networks to better predict this behavior.

Vulnerability 15	Findings

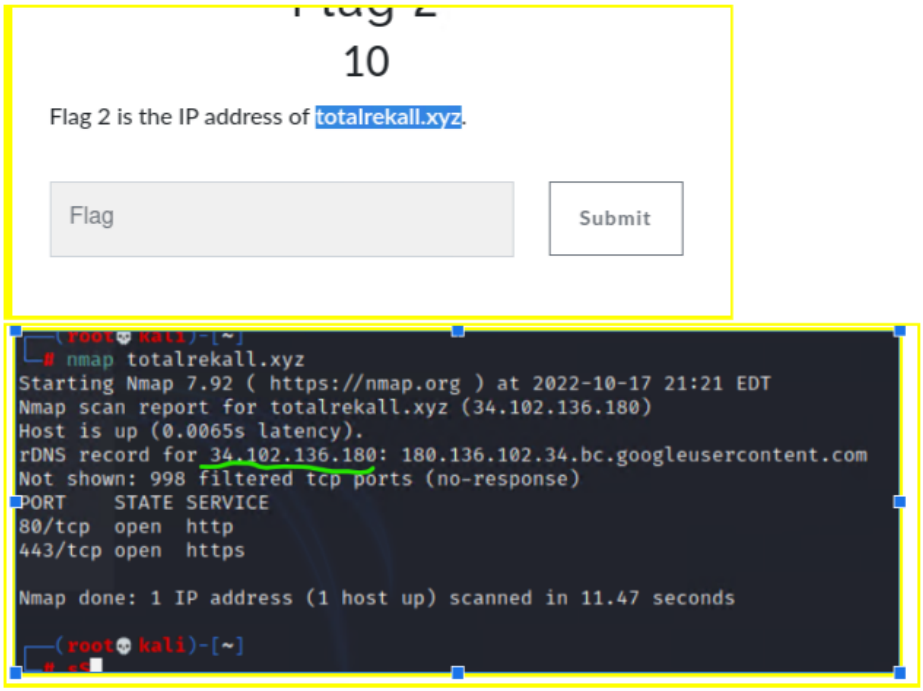
Title	Directory traversal
Type (Web app / Linux OS / Windows OS)	Web app
Risk Rating	**High**
Description	Using the vulnerability from Flag 10 or Flag 11, you can run ls to see the old_disclaimers directory. Using that finding, change the URL to: http://192.168.13.35/disclaimer.php?page=old_disclaimers/disclaimer_1.txt Note that the resource changed from disclaimer_2.txt to disclaimer_1.txt, as this is the older version.
Images	
Affected Hosts	192.168.14.35
Remediation	Monitor all file system interactions the application performs. A file system ACL contains the information about user access privileges to a system object, a single file, or a file directory and informs the computer operating system about those privileges. An access control list is connected to every object by the object's security property.

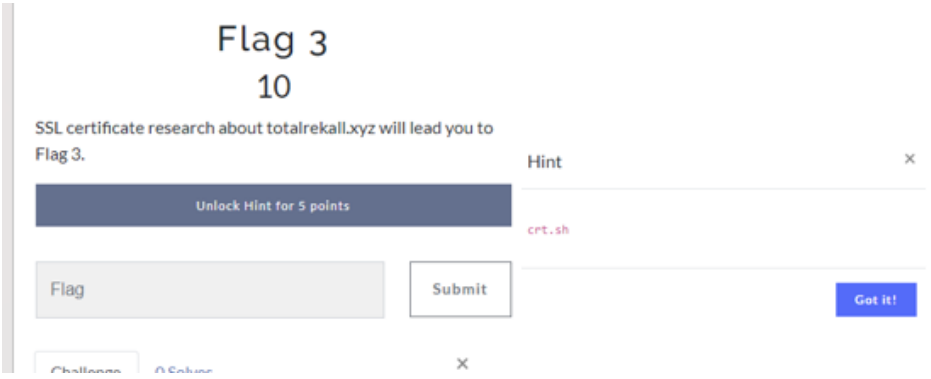
Vulnerability 16	Findings
Title	Open source exposed data
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	High
Description	On the Domain Dossier webpage, view the WHOIS data for totalrekall.xyz. The address will show the flag: Registrant Street: h8s692hskasd Flag1
Images	

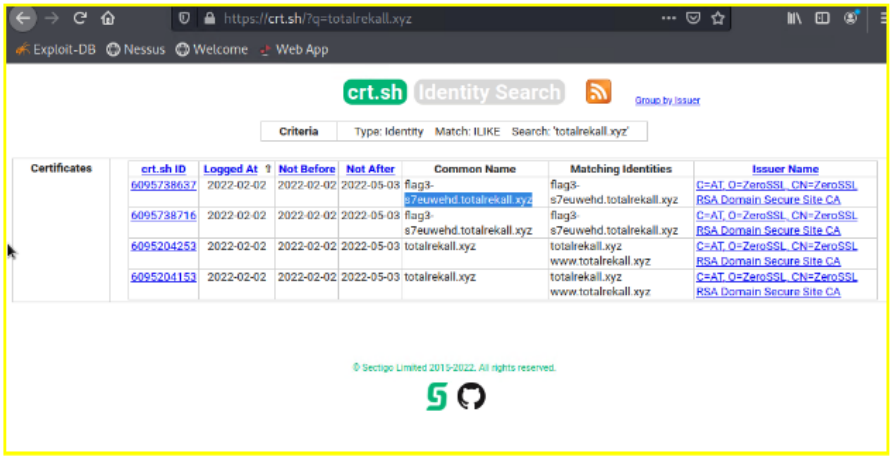


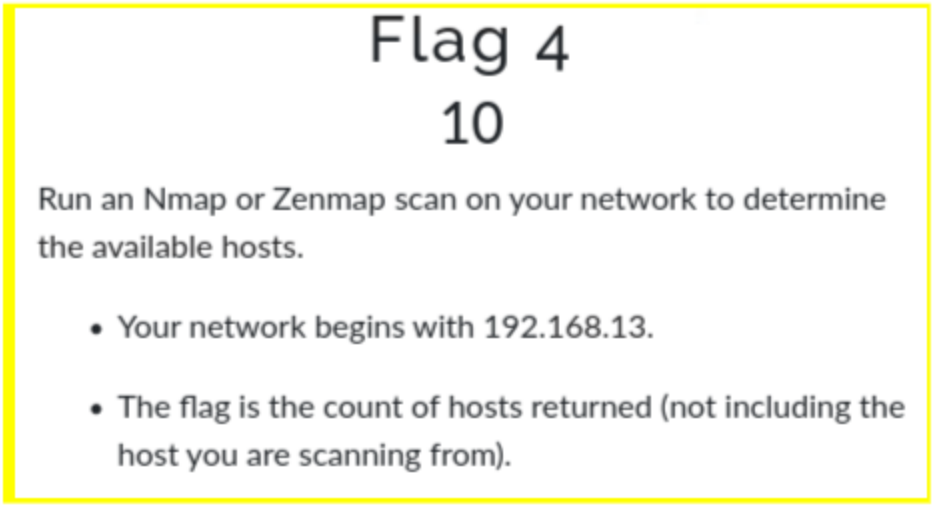
Affected Hosts	https://centralops.net/co/DomainDossier.aspx (34.102.136.180)
Remediation	Remove any vital records from the webpage .xyz that could be collected by these types of scans. Test your website against these types of open source data collection tools.

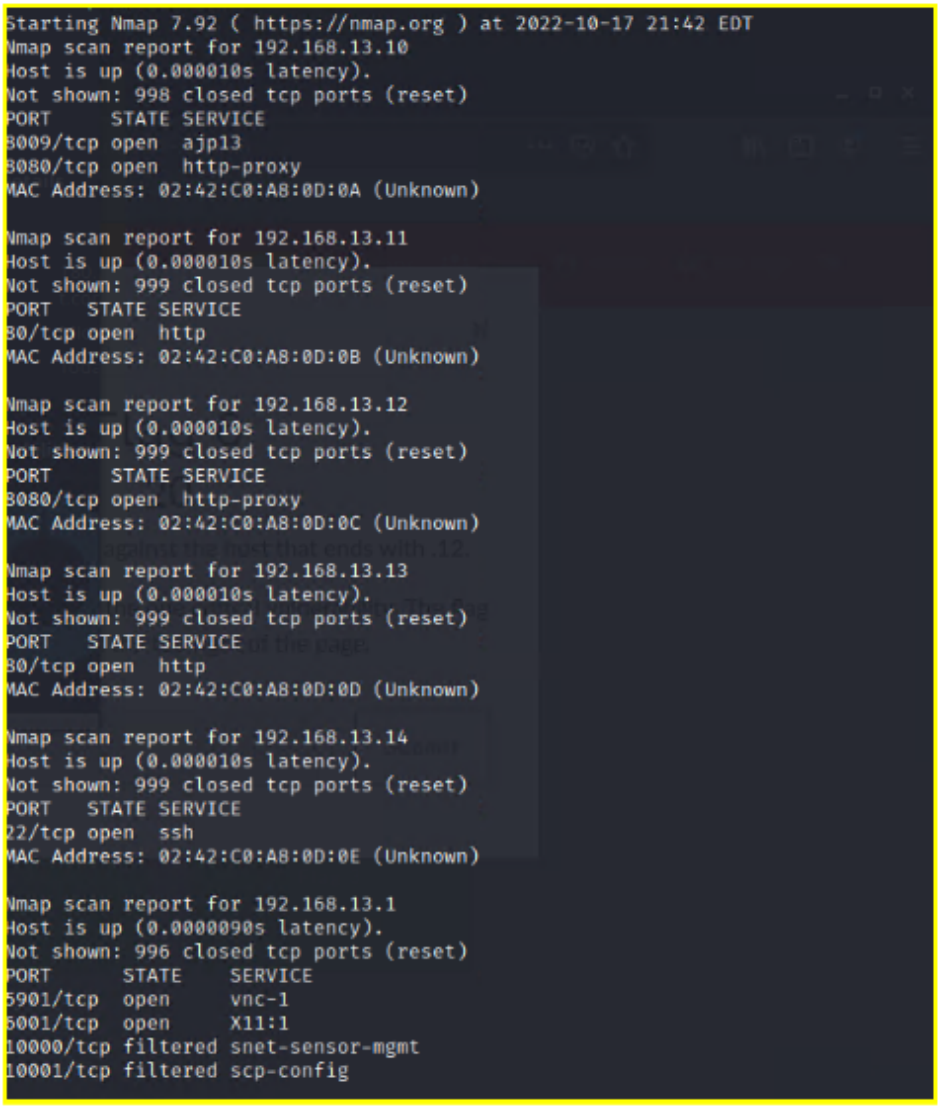
Vulnerability 17	Findings
Title	Scanning Vulnerability
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	**Medium**
Description	Ran a namp against the totalrekall.xyz web page. Found the public IP for this webpage.

Images	
Affected Hosts	34.102.136.180
Remediation	Only allow certain IP addresses to ping your website using scanning tools. Possible defenses include blocking the probes, restricting information returned, slowing down the Nmap scan, and returning misleading information

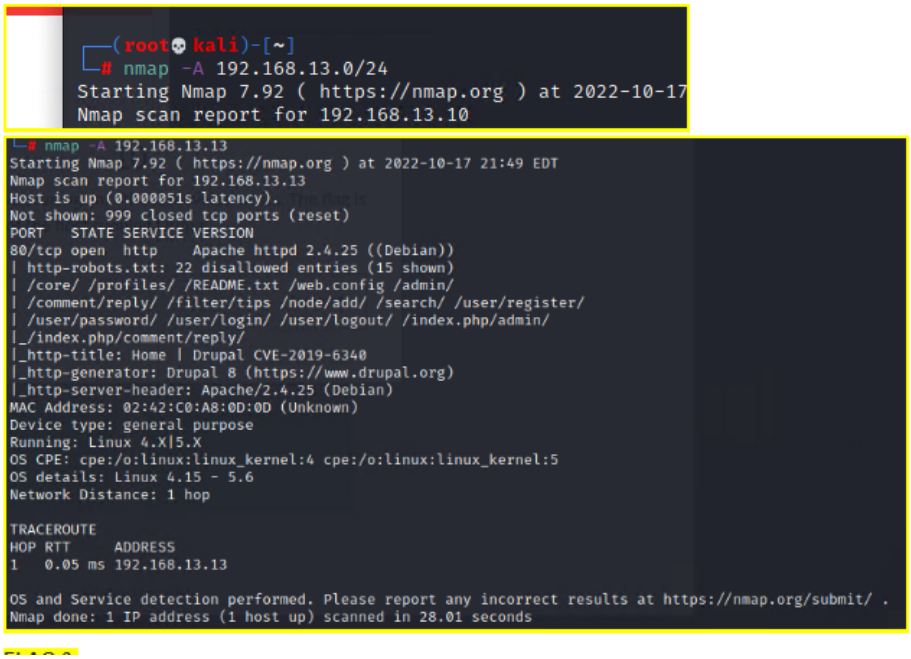
Vulnerability 18	Findings
Title	Open source exposed data
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	High
Description	On crt.sh, search for totalrekall.xyz to view the flag: s7euwehd.totalrekall.xyz.
Images	

	
Affected Hosts	34.102.136.180
Remediation	Enforce automated remediation policies to fix vulnerable open source components, including newly disclosed vulnerabilities. Give developers and security professionals the tools they need to manage open source security from within their native development environments

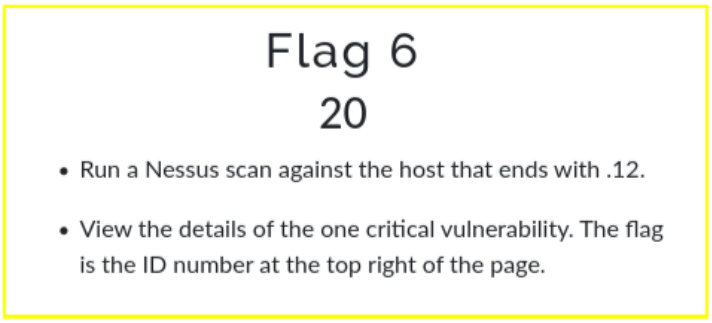
Vulnerability 19	Findings
Title	Scanning Vulnerability
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	**Medium**
Description	Run an Nmap scan for the network (nmap 192.168.13.0/24) to determine that there are 5 hosts excluding the host scanning from.
Images	

	
Affected Hosts	192.168.13.0/24
Remediation	Only allow certain IP addresses to ping your website using scanning tools. Possible defenses include blocking the probes, restricting information returned, slowing down the Nmap scan, and returning misleading information

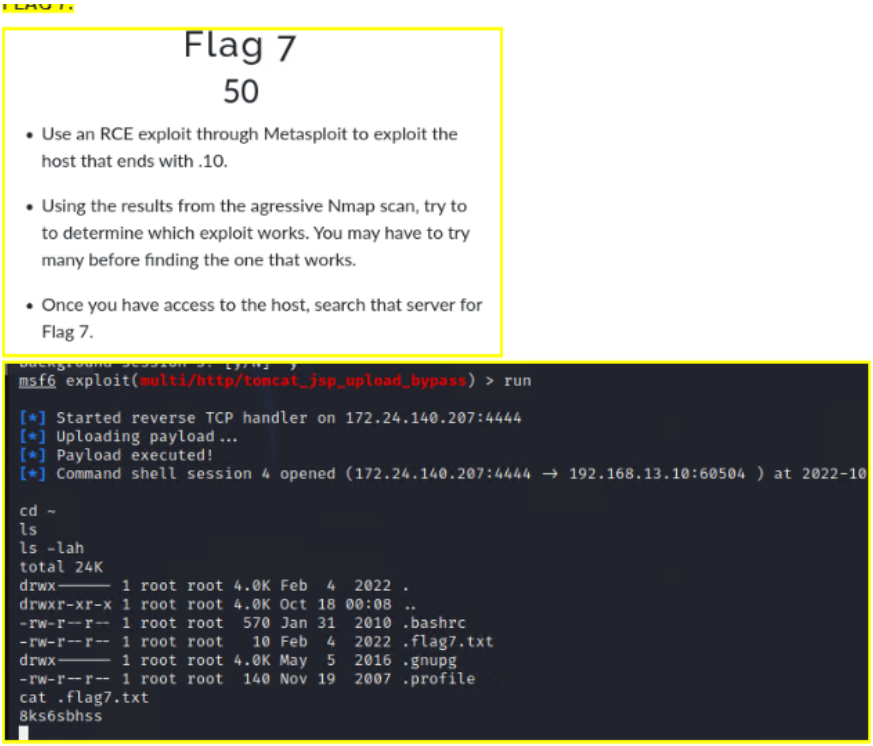
Vulnerability 20	Findings
Title	Aggressive Scan Vulnerability
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	High
Description	Run an aggressive Nmap scan: nmap -A 192.168.13.0/24 Analyze the results to see that the host that runs Drupal is 192.168.13.13.

Images	
Affected Hosts	192.168.13.13
Remediation	Only allow certain IPs to ping this IP and put firewalls to block port 80 or use another port in the form of hiding this port instead of in plain site.

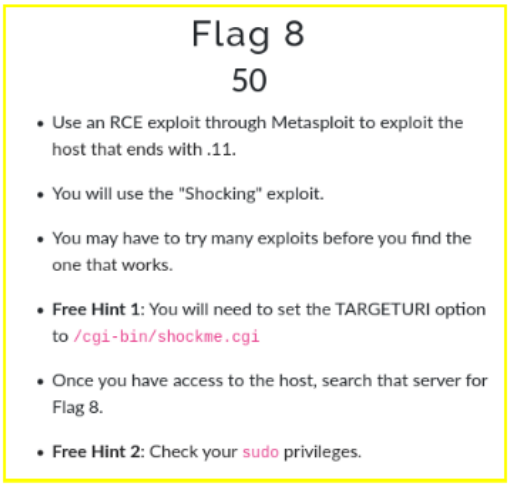
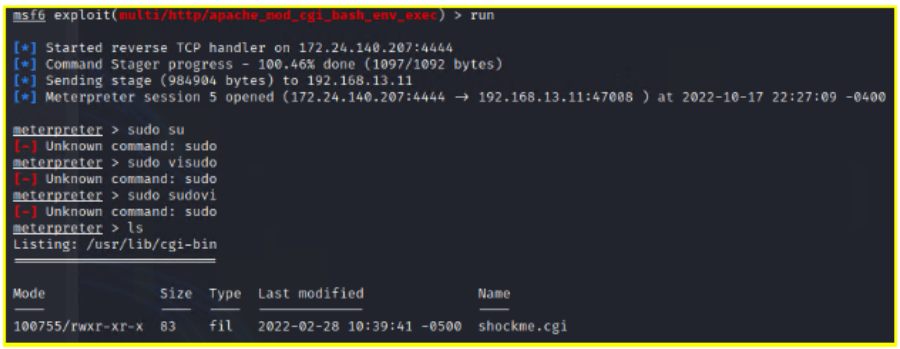
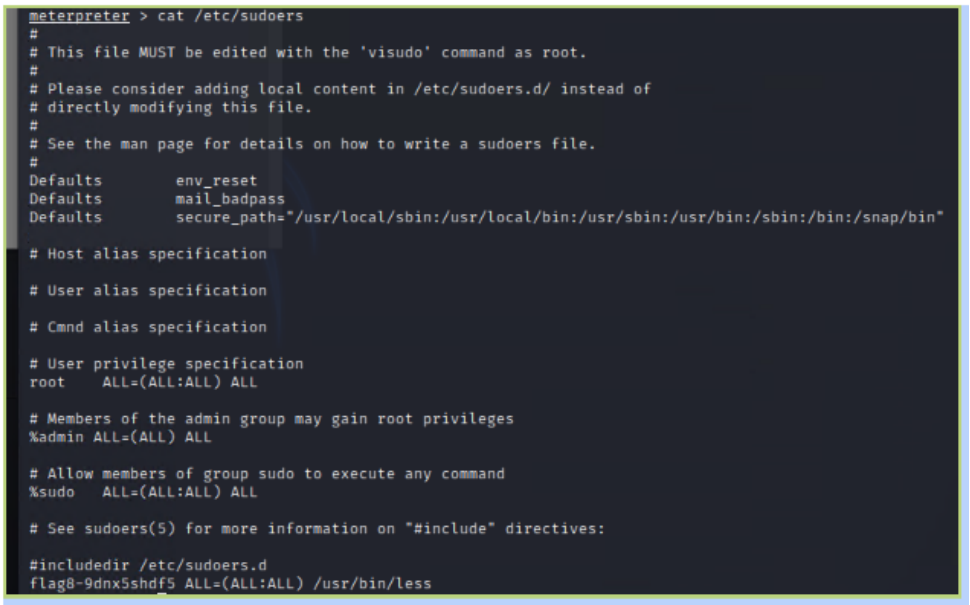
Vulnerability 21	Findings
Title	Scanning Vulnerability (Nessus)
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	**Critical**
Description	Ran a Nessus scan for 192.168.13.12 One critical vulnerability found appearing for Apache Struts. Clicked on this critical vulnerability. The id 97610 displays on the top right of this page.

Images	 97610 
Affected Hosts	192.168.13.12
Remediation	Patch the host IP. You can create an asset filter to view and report on assets where a vulnerability was recently mitigated. You can automate remediation scans using the API.

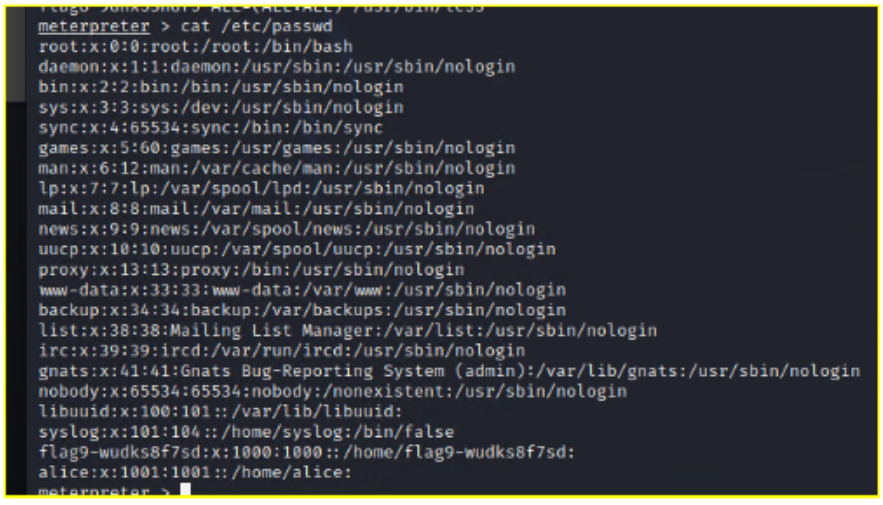
Vulnerability 22	Findings
Title	Apache Tomcat Remote Code Execution Vulnerability (CVE-2017-12617)
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	High
Description	Ran MSFconsole. Search for exploits that have Tomcat and JSP. Used the exploit multi/http/tomcat_jsp_upload_bypass, and set the option for

	the RHOST to 192.168.13.10. After successfully getting a Meterpreter shell, enter "SHELL" to get to the command line. Ran the following to get the flag: / cat /root/.flag7.txt
Images	
Affected Hosts	192.168.13.10
Remediation	Block the TCP ports from this IP or only allow shell access to very controlled IPs. Update your security controls and firewalls with IDS/IPS.

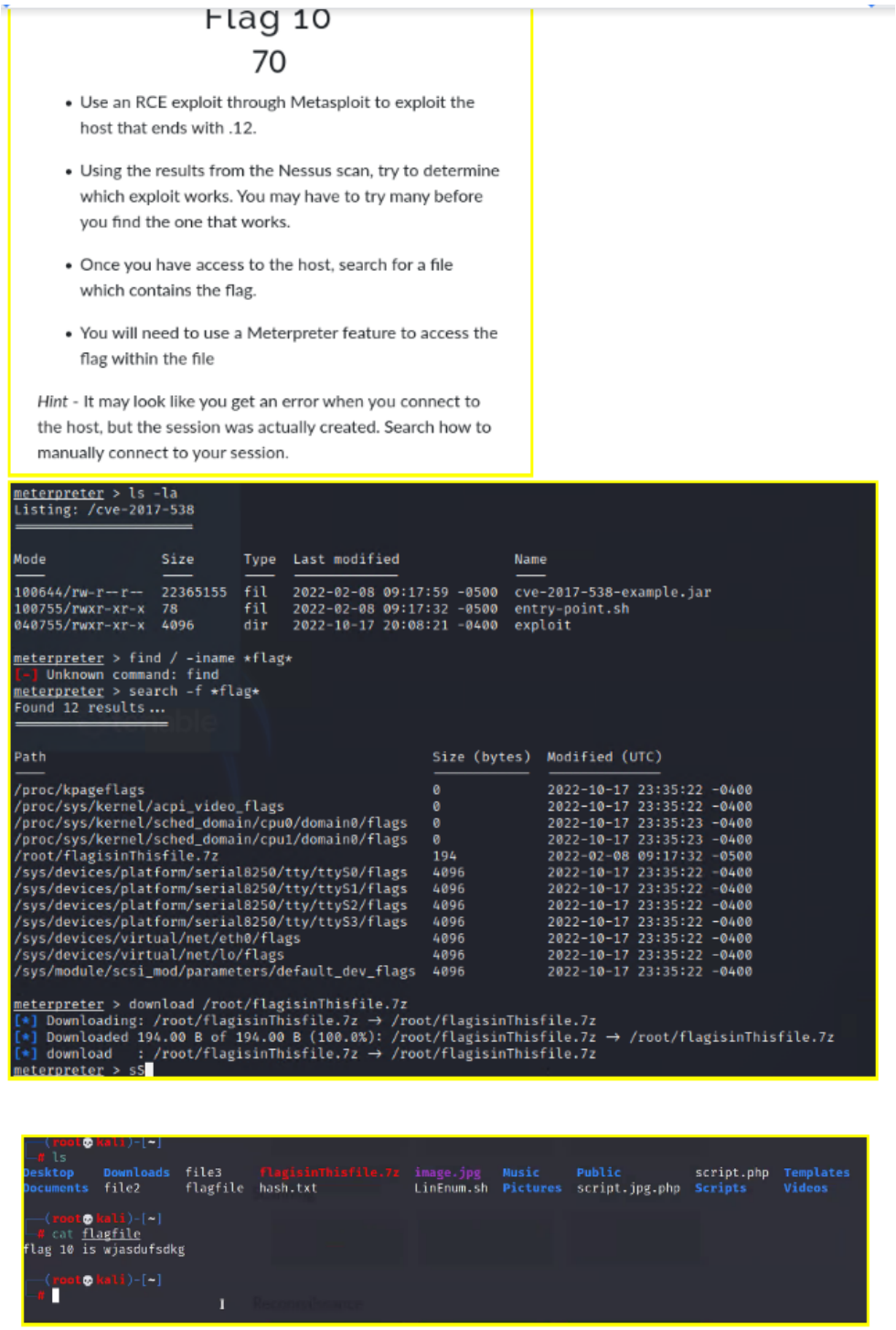
Vulnerability 23	Findings
Title	Shellshock
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	High
Description	Ran MSFconsole, and search exploits that have Shellshock. Ran MSF (exploit/multi/http/apache_mod_cgi_bash_env_exec) and set the following options: target URI(The vulnerable webpage): /cgi-bin/shockme.cgi RHOST: 192.168.13.11 To get the flag, ran the following from a shell on the exploited machine: cat /etc/sudoers

Images	  
Affected Hosts	192.168.13.11
Remediation	Block the TCP ports from this IP or only allow shell access to very controlled IPs. Update your security controls and firewalls with IDS/IPS.

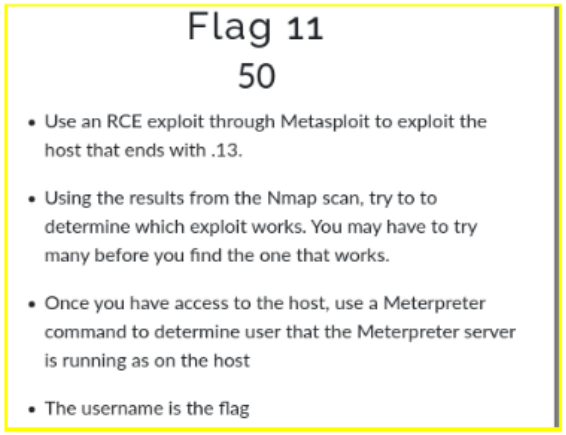
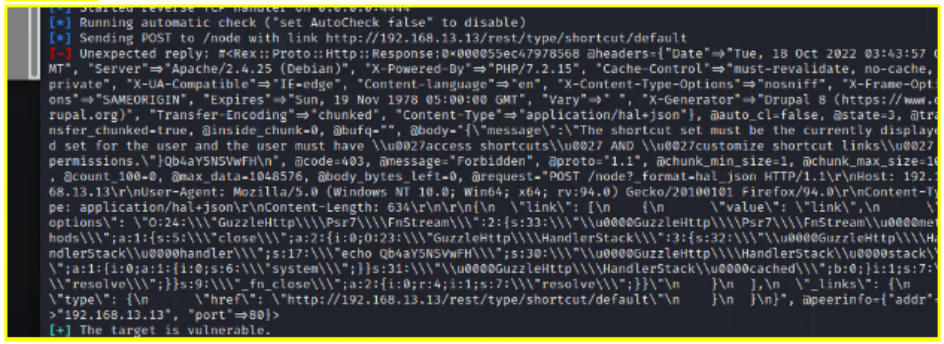
Vulnerability 24	Findings

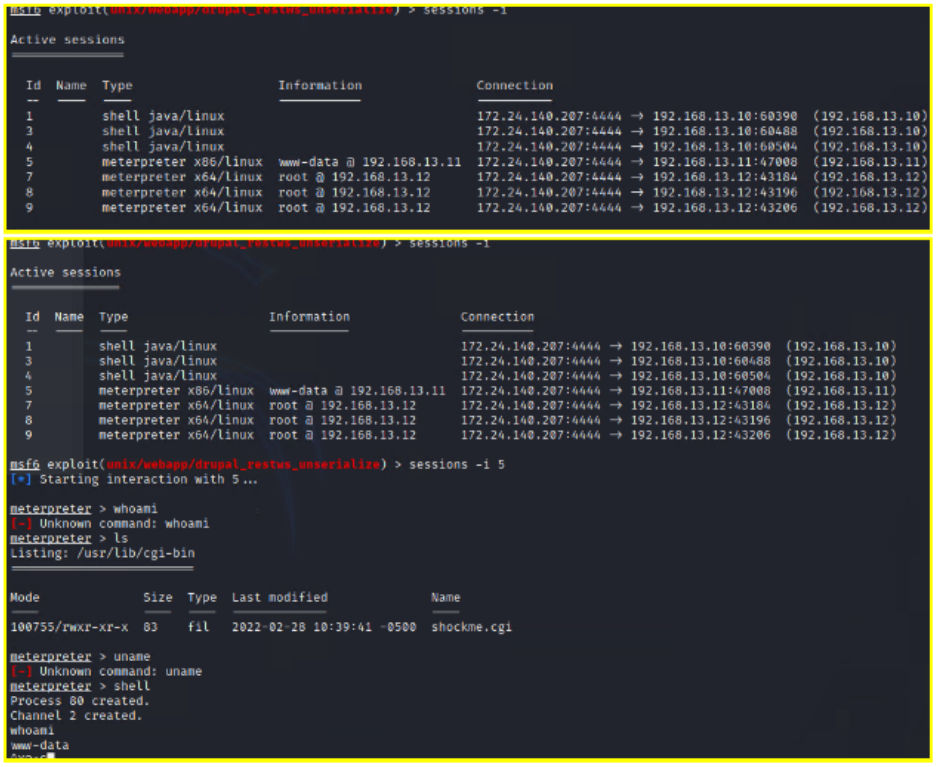
Title	Pivoting/lateral movement
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	**High**
Description	From the same machine as above. Ran cat /etc/passwd. Found our target user to attack: Alice.
Images	 
Affected Hosts	192.168.13.11
Remediation	Block the TCP port from this common reverse shell from being started. Update your security controls and firewalls with IDS/IPS.

Vulnerability 25	Findings
Title	Apache struts CVE-2017-5638
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	**Medium**
Description	Determined via the Nessus scan that this host is vulnerable to Struts. After connecting to MSFconsole, search for Struts exploits. Used the following exploit to get a Meterpreter shell: multi/http/struts2_content_type_ognl. Set the RHOSTS to 192.168.13.12 Manually connect to the session to get the meterpreter shell with: sessions -i <session number> Use Meterpreter to download the following file to your Kali machine: /root/flagisinThisfile.7z From your Kali machine, unzip the file with the following command: 7z x flagisinThisfile.7z Use cat with the flag file to view the flag.

Images	
Affected Hosts	192.168.13.12
Remediation	Patch systems to ensure they are running the latest patches and update regularly.

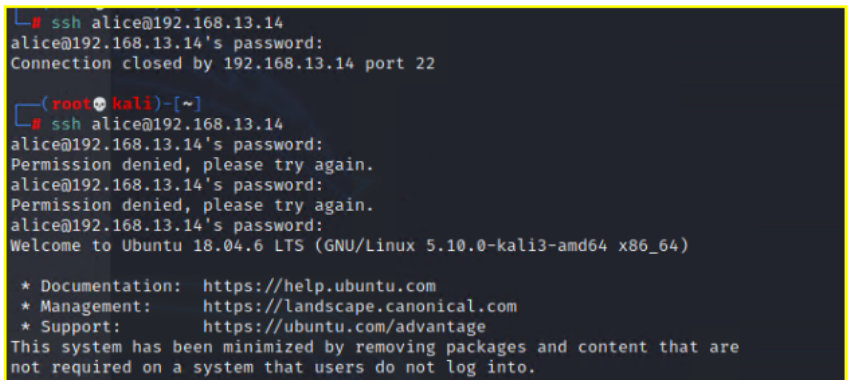
Vulnerability 26	Findings
Title	Drupal - CVE-2019-6340

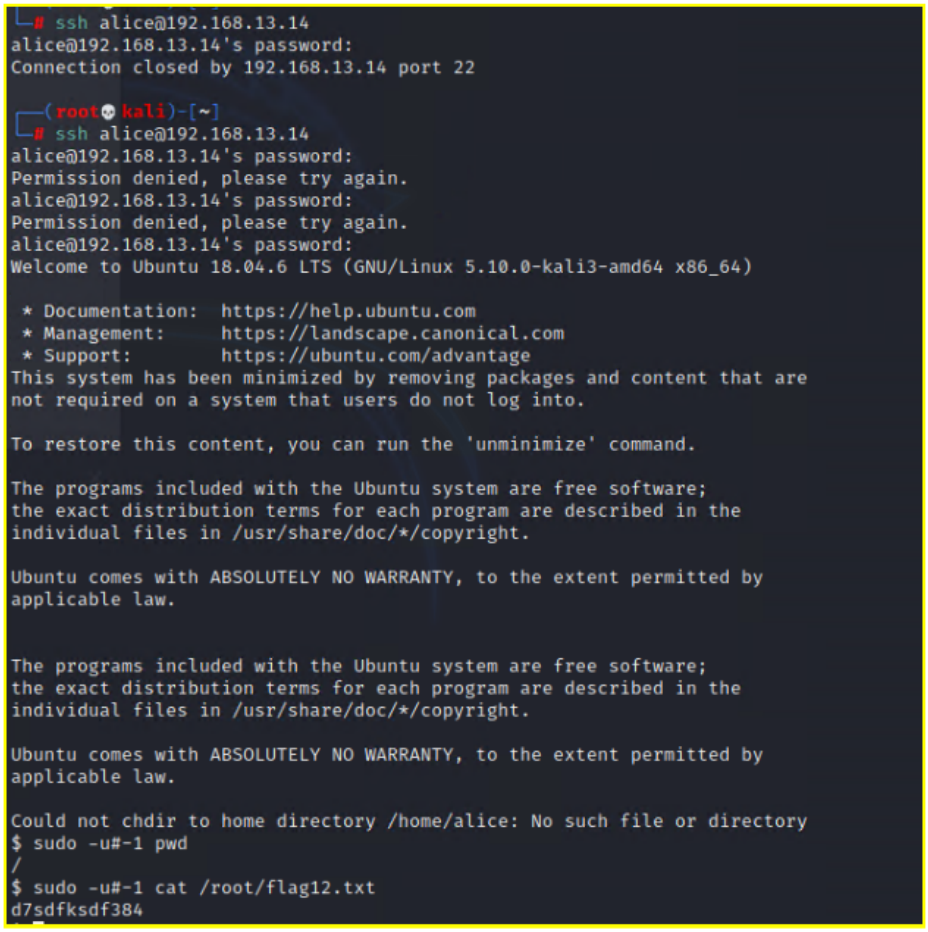
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	**Medium**
Description	After connecting to MSFconsole, search for Drupal exploits. Use the following exploit to get a Meterpreter shell MSF: unix/webapp/drupal_restws_unserialize Set RHOSTS to 192.168.13.13 After getting the Meterpreter shell, run getuid to get the username
Images	 www-data  *continued below*

	
Affected Hosts	192.168.13.13
Remediation	Block the TCP port from this common reverse shell from being started. Update your security controls and firewalls with IDS/IPS.

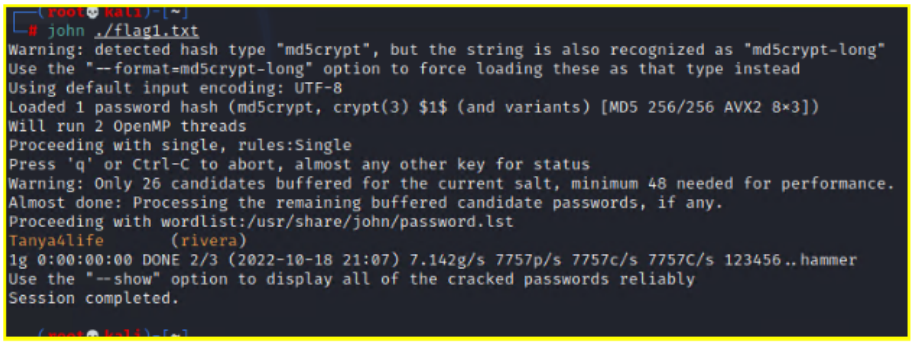
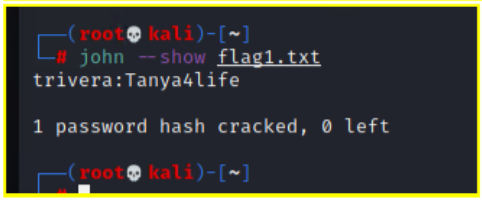
Vulnerability 27	Findings
Title	Privilege-escalation/SSH - **CVE-2019-14287**
Type (Web app / Linux OS / Windows OS)	Linux OS
Risk Rating	**Medium**
Description	When viewing the WHOIS data from Flag 1, notice that the name is: sshuser Alice SSH into the server: ssh alice@192.168.13.14 Guess that the password is: alice To conduct the privilege escalation exploit and obtain the flag, run the following: sudo -u#-1 cat /root/flag12.txt

Images

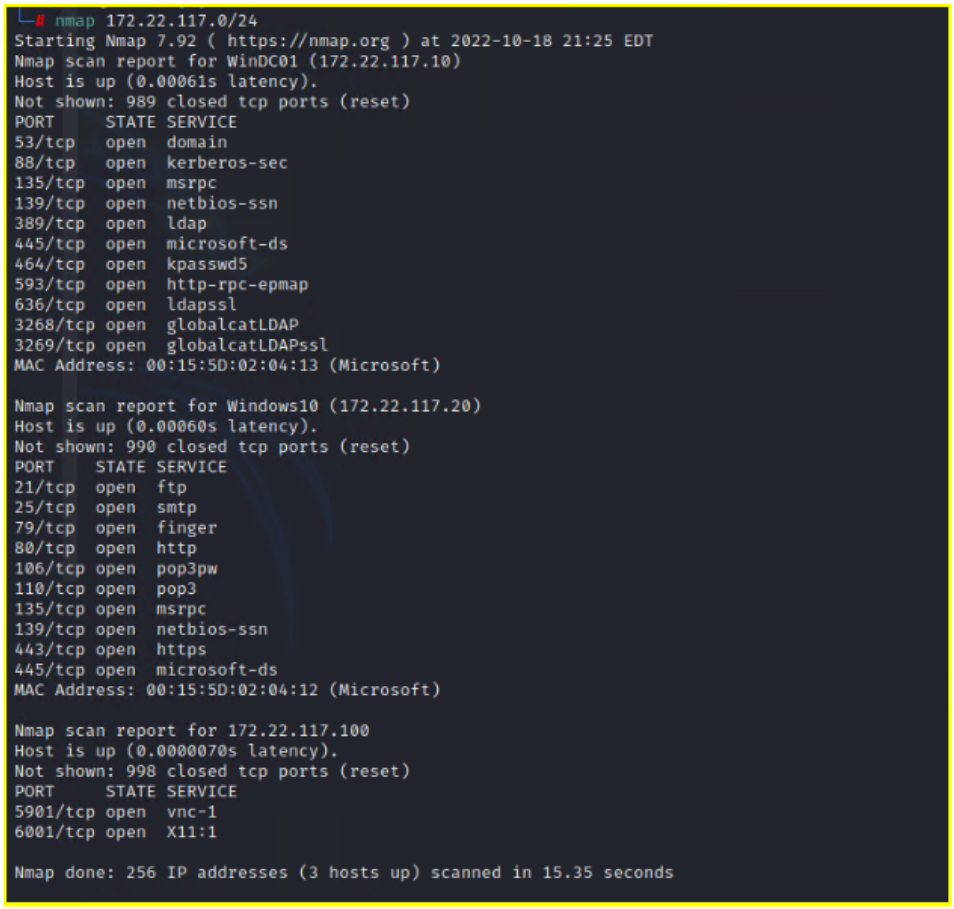
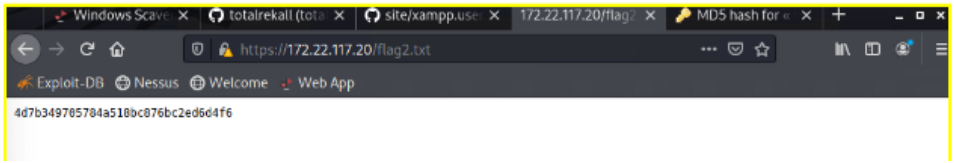


	
Affected Hosts	192.168.13.14
Remediation	Patch systems to ensure they are running the latest patches and update regularly. Enforce strong passwords.

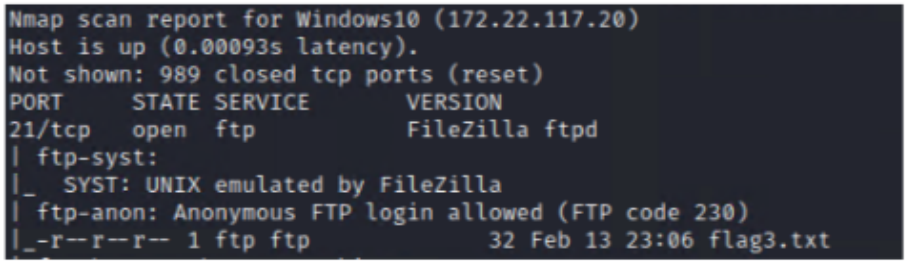
Vulnerability 28	Findings
Title	Brute force attack - Password Cracking
Type (Web app / Linux OS / Windows OS)	Windows OS
Risk Rating	**High**
Description	Searching GitHub should lead to finding the totalrekall GitHub page. Searching the site repository will lead to the xampp.users page, which contains the credentials trivera:$apr1$A0vSKwao$GV3sgGAj53j.c3GkS4oUC0. Used john the ripper to then crack the hash.

Images	  
Affected Hosts	N/A
Remediation	Do not save these credentials anywhere in an open platform and this should be a new policy to implement. Always enforce strong passwords.

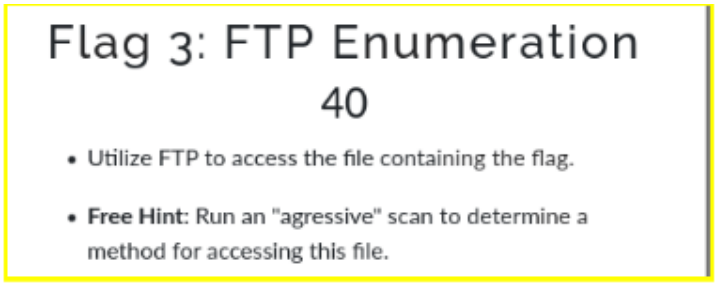
Vulnerability 29	Findings
Title	HTTP Enumeration - port scanning
Type (Web app / Linux OS / Windows OS)	Windows OS
Risk Rating	**Medium**
Description	From the Kali machine, a port scan of the subnet that the Kali machine is on (172.22.117.0/24) will reveal two machines: Win10 @ 172.22.117.20 Server2019 @ 172.22.117.10 The port scan will reveal several ports open on Win10, one of which is HTTP.

	The credentials cracked from the discovered GitHub page, trivera / Tanya4life, will grant access. Inside is flag2.txt is the hidden info.
Images	 (flagbelow) 
Affected Hosts	172.22.117.0/24 (172.22.117.20)
Remediation	Use https as secure means of encryption across all variations of the web address. Block these ports and use port scanning on the security side as well to monitor this behavior.

Vulnerability 30	Findings
Title	FTP Enumeration
Type (Web app / Linux OS / Windows OS)	Windows OS

Risk Rating	Critical
Description	Returning to the port scan results will show "FTP" open on port 21. If the Nmap scan was done using the -A flag or using the NSE script for FTP anonymous access, the scan will reveal that FTP anonymous access is possible.
Images	

```
Nmap scan report for Windows10 (172.22.117.20)
Host is up (0.00093s latency).
Not shown: 989 closed tcp ports (reset)
PORT     STATE SERVICE       VERSION
21/tcp   open  ftp           FileZilla ftpd
| ftp-syst:
|_  SYST: UNIX emulated by FileZilla
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_-r--r--r-- 1 ftp ftp            32 Feb 13 23:06 flag3.txt
```




```
┌──(root💀kali)-[~]
└─# ftp
ftp> open 172.22.117.20
Connected to 172.22.117.20.
220-FileZilla Server version 0.9.41 beta
220-written by Tim Kosse (Tim.Kosse@gmx.de)
220 Please visit http://sourceforge.net/projects/filezilla/
Name (172.22.117.20:root):
```

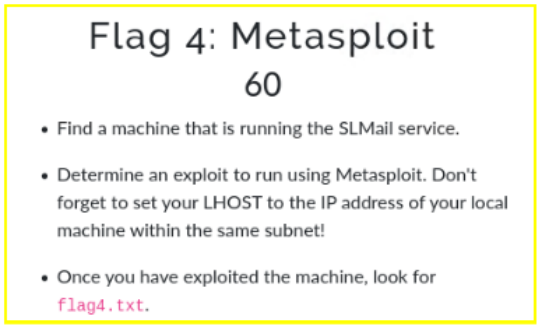
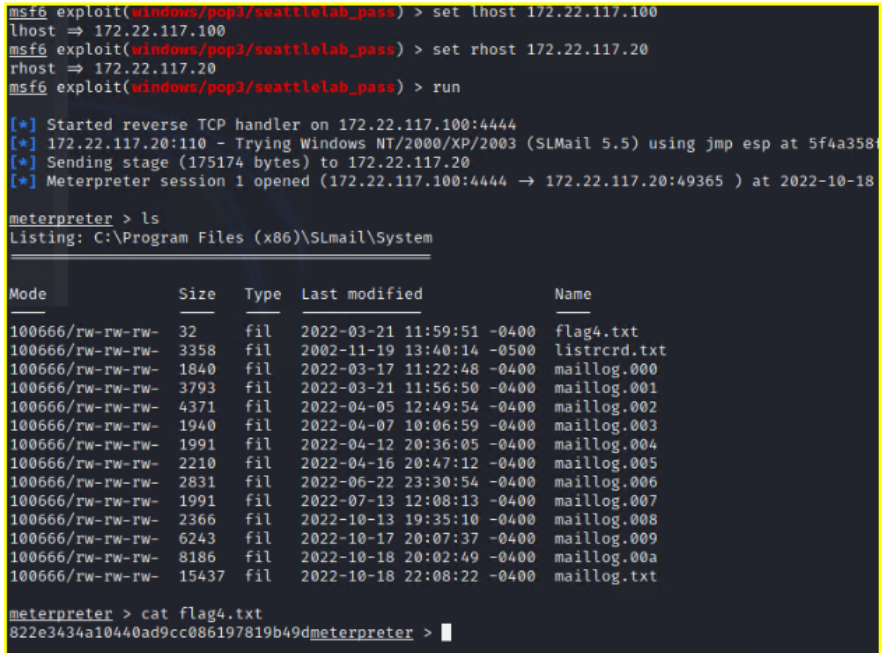
```
┌──(root💀kali)-[~]
└─# ftp
ftp> open 172.22.117.20
Connected to 172.22.117.20.
220-FileZilla Server version 0.9.41 beta
220-written by Tim Kosse (Tim.Kosse@gmx.de)
220 Please visit http://sourceforge.net/projects/filezilla/
Name (172.22.117.20:root): anonymous
331 Password required for anonymous
Password:
230 Logged on
Remote system type is UNIX.
ftp> ls
200 Port command successful
150 Opening data channel for directory list.
-r--r--r-- 1 ftp ftp              32 Feb 15  2022 flag3.txt
226 Transfer OK
ftp> cat flag3.txt
?Invalid command
ftp> head flag3.txt
?Invalid command
ftp> download flag3.txt
?Invalid command
ftp> open flag3.txt
Already connected to 172.22.117.20, use close first.
ftp> get flag3.txt
local: flag3.txt remote: flag3.txt
200 Port command successful
150 Opening data channel for file transfer.
226 Transfer OK
32 bytes received in 0.00 secs (390.6250 kB/s)
ftp>
```

moved to other terminate after get (download) the file

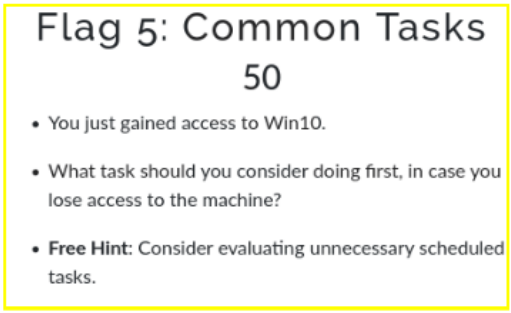
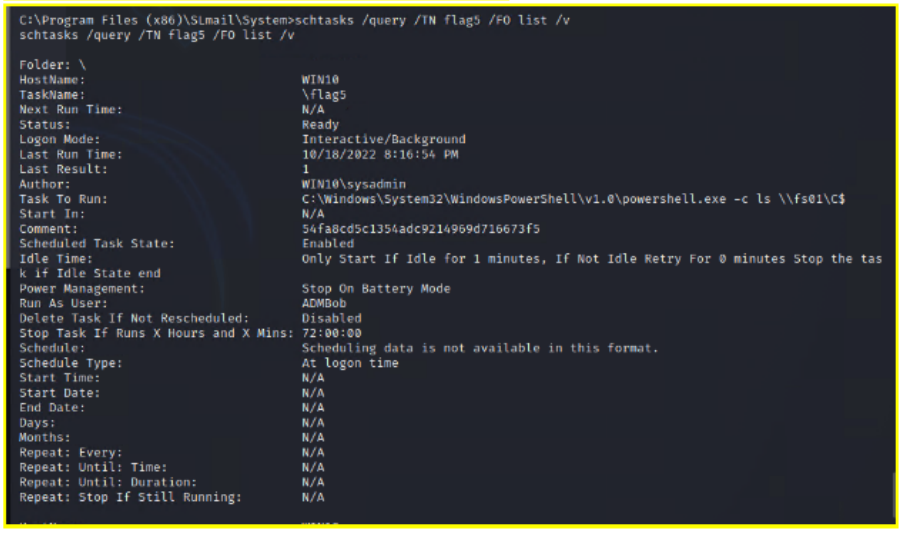
```
┌──(root💀kali)-[~]
└─# cat flag3.txt
89cb548970d44f348bb63622353ae278
```

Affected Hosts	172.22.117.20
Remediation	It is best to close ports that are not used often. Certain firewall rules and IDS/IPS should be implemented. Only allow access when needed.

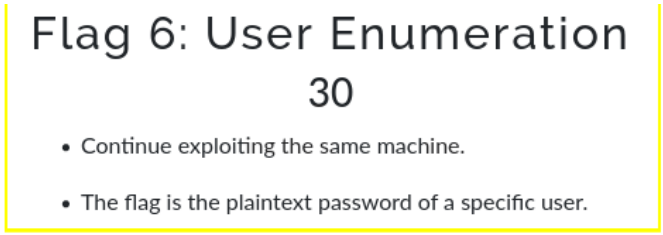
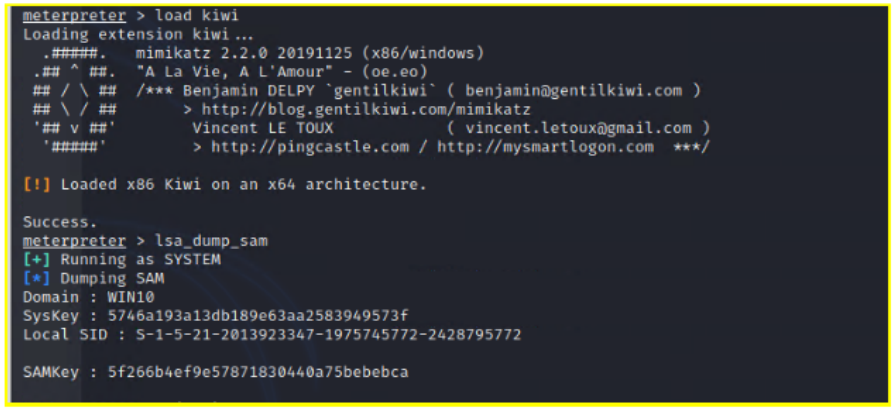
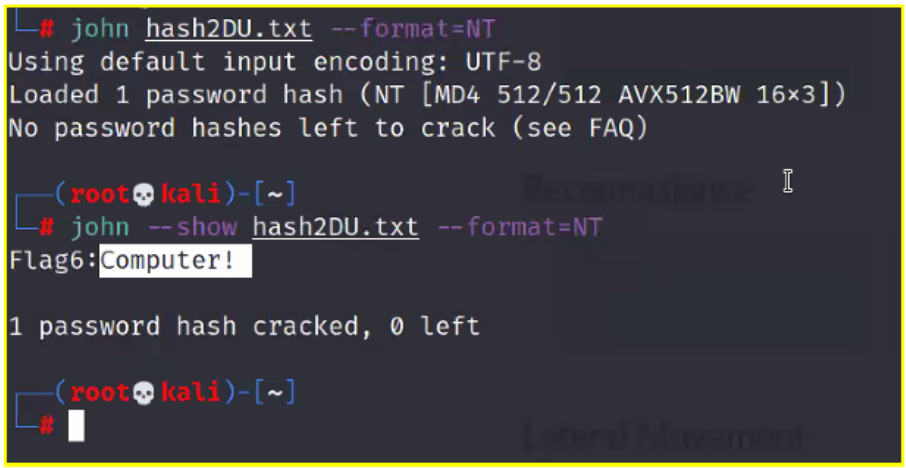
Vulnerability 31	Findings
Title	Metasploit on SLMail service

Type (Web app / Linux OS / Windows OS)	Windows OS
Risk Rating	High
Description	Load Metasploit via MSFconsole, loading the SLMail module and setting the RHOSTS to 172.22.117.20, and then running the exploit will grant a Meterpreter shell. identified the most viable exploit and successfully got a reverse shell.
Images	 has to set to inide the network not my LHOST IP 
Affected Hosts	172.22.117.20
Remediation	Update and patch all systems. Set up the firewall rules using IDS/IPS for these open ports and closely monitor them.

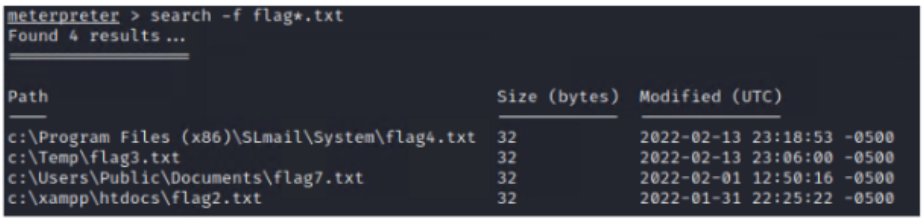
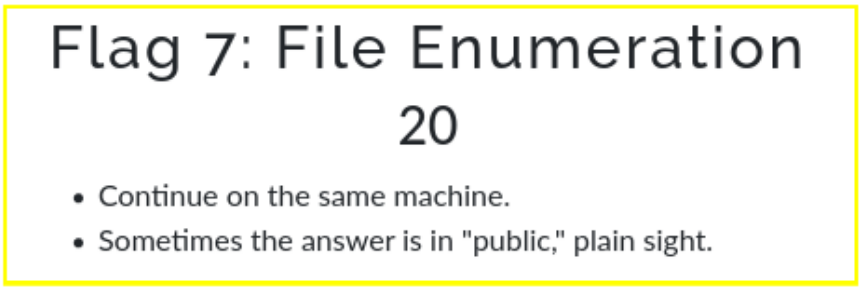
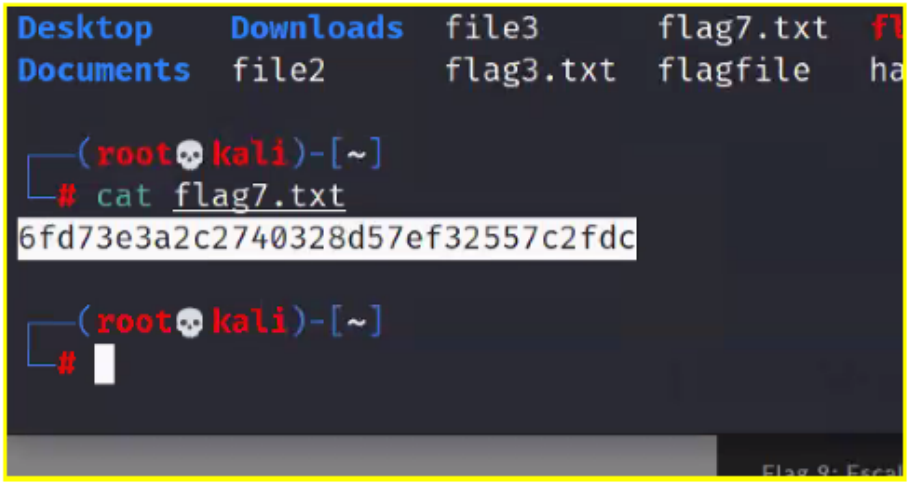
Vulnerability 32	Findings
Title	Task managers exploit (using shell)
Type (Web app / Linux OS / Windows OS)	Windows OS

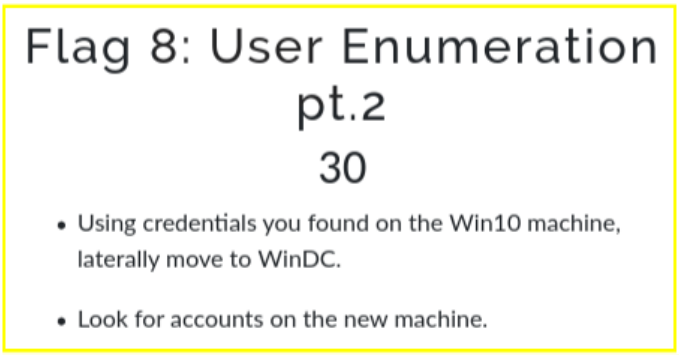
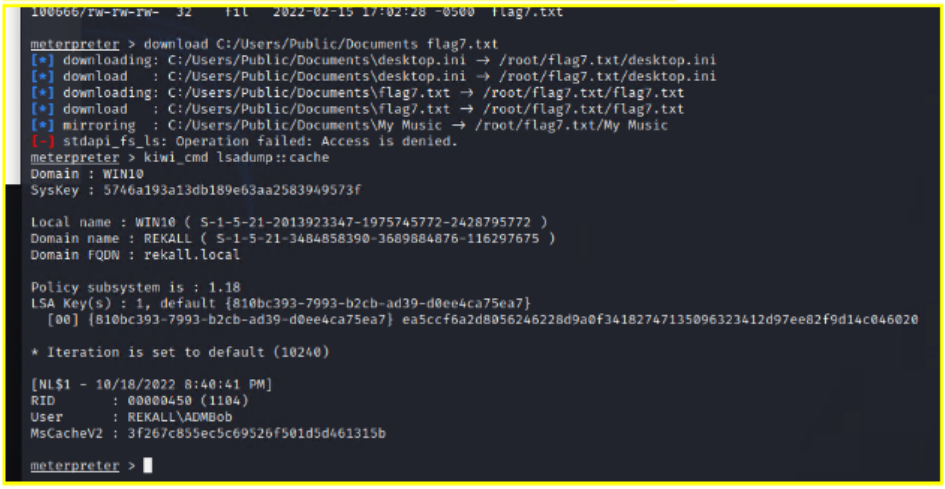
Risk Rating	High
Description	Scheduled tasks by dropping into a command shell within Meterpreter and using the schtasks command schtasks /query.
Images	 
Affected Hosts	172.22.117.20
Remediation	Set up task scheduler to only allow tasks to be scheduled by the GUI and not on the command line. Patch and update systems regularly and monitor the logs for this type of activity.

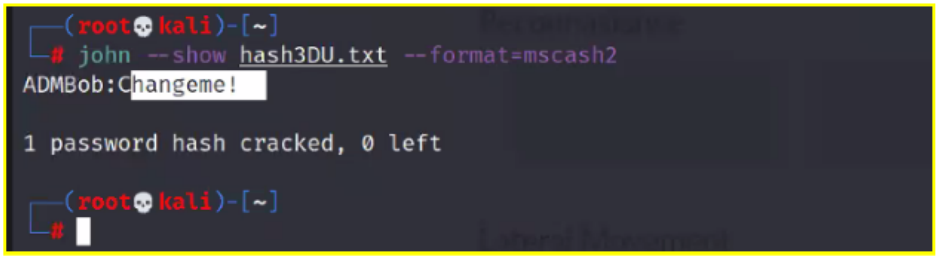
Vulnerability 33	Findings
Title	Kiwi - LSA dump (brute force/SLMail)
Type (Web app / Linux OS / Windows OS)	Windows OS
Risk Rating	High
Description	Used kiwi command lsa_dump_sam of the SAM file and then used john the ripper to crack the hash.

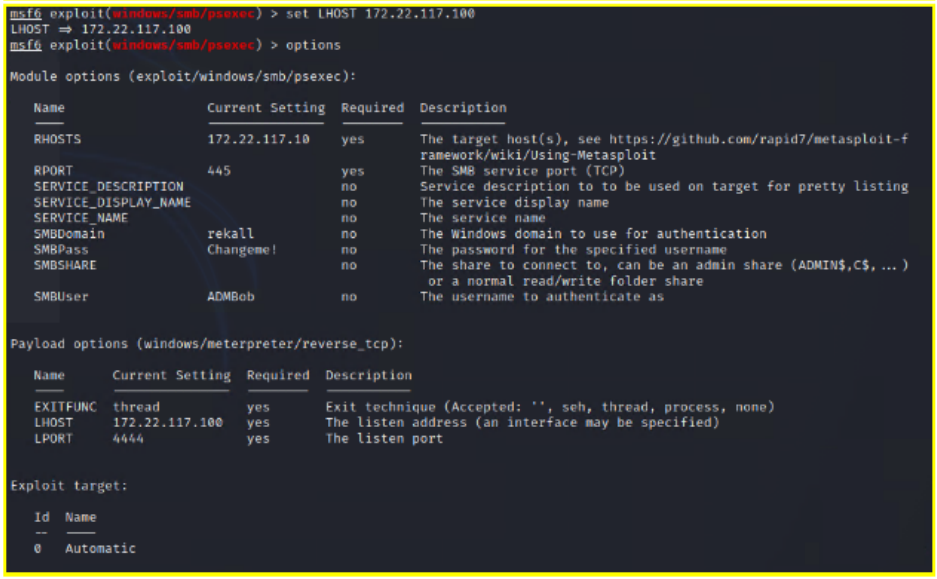
Images	   
Affected Hosts	172.22.117.20
Remediation	Update and patch all systems against this type of exploit. Set up the firewall rules using IDS/IPS for these open ports and closely monitor them. Use much longer and harder passwords as a standard across the network to make them

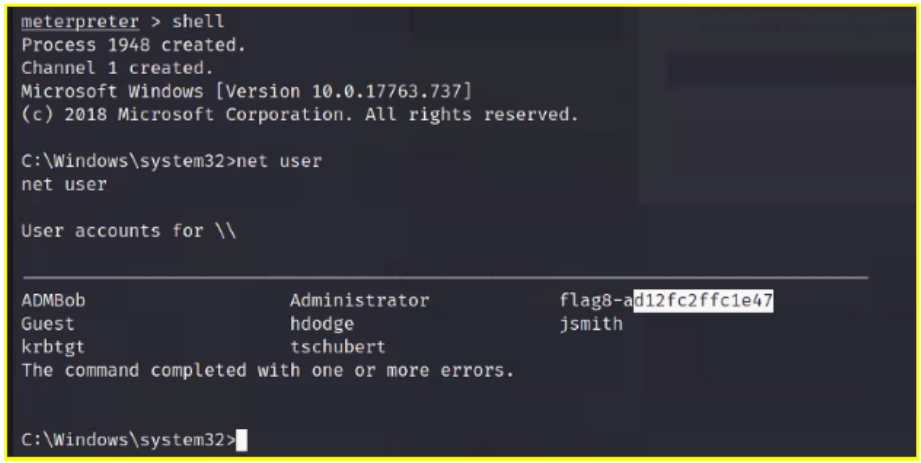
	harder to crack.

Vulnerability 34	Findings
Title	File Enumeration
Type (Web app / Linux OS / Windows OS)	Windows OS
Risk Rating	**Medium**
Description	The meterpreter shell then allowed us to go undetected and search the file system for any files that had our key words for this flag.
Images	  
Affected Hosts	172.22.117.20
Remediation	Set alerts for your logging procedure/systems to pick up on this type of activity on the network. Patch and update regularly.

Vulnerability 35	Findings
Title	User Enumeration - (brute force)
Type (Web app / Linux OS / Windows OS)	Windows OS
Risk Rating	**High**
Description	Using kiwi to dump the cached credentials on Win10 will reveal that an administrator, ADMBob, has their credentials cached. These new credentials have access to the Server2019 machine. By using the PsExec module in Metasploit with these credentials, a SYSTEM shell was obtained on Server2019.
Images	 

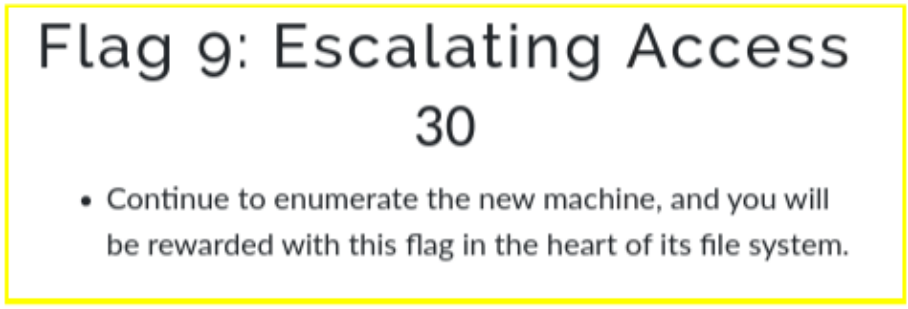
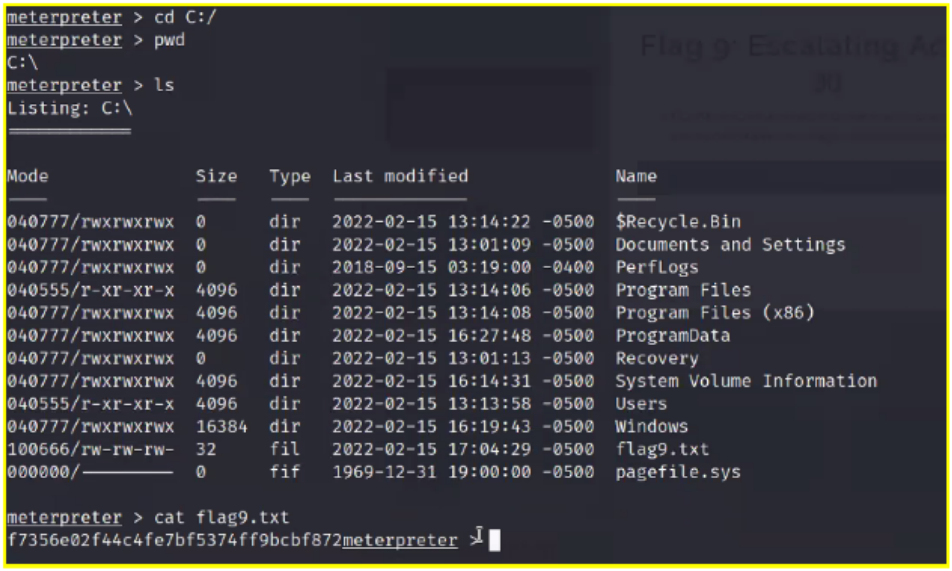




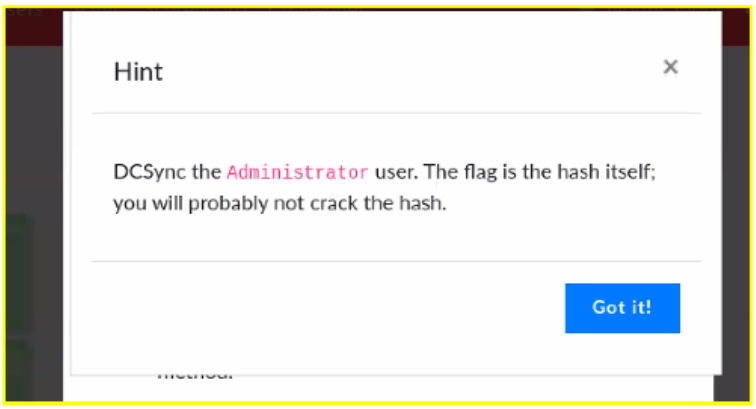
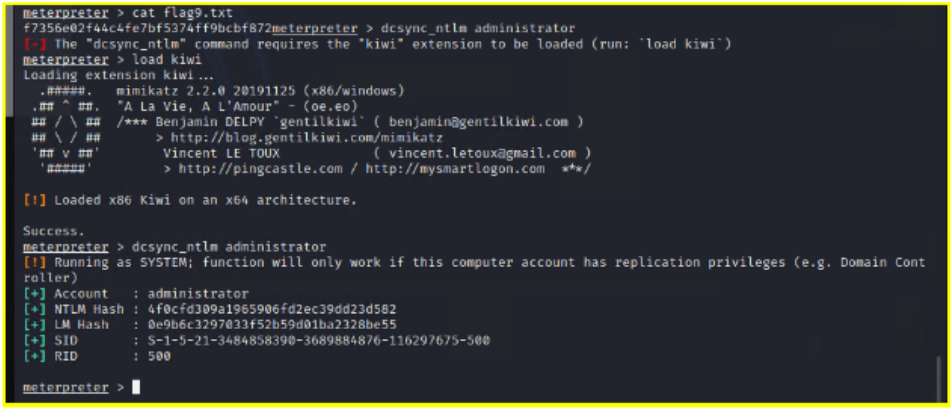


Affected Hosts	172.22.117.10
Remediation	Use Software as a service to validate local and remote sign-ins in conjunction with your firewalls and IPS or IDS for your log monitoring.

Vulnerability 36	**Findings**
Title	Escalating access
Type (Web app / Linux OS / Windows OS)	Windows OS

Risk Rating	**Medium**
Description	By moving to the root, C:\, and listing the files, flag9.txt can be read via cat in Meterpreter.
Images	 
Affected Hosts	172.22.117.20
Remediation	Use detection tools and prevention systems to alert your security team of suspicious activity.

Vulnerability 37	Findings
Title	NTLM password cracking - compromise Admin
Type (Web app / Linux OS / Windows OS)	Windows OS
Risk Rating	**High**
Description	Using kiwi to DCSync the Administrator user on Server2019 will reveal their NTLM password hash.

Images	  
Affected Hosts	172.22.117.20
Remediation	Set alerts for your logging procedure/systems to pick up on this type of activity on the network. Patch and update regularly.