Cybersecurity

Penetration Test Report

Rekall Corporation

Penetration Test Report

Confidentiality Statement

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

Table of Contents

Contact Information

Company Name	Cyber Shield
Contact Name	Yesenia Morales
Contact Title	Pentester

Document History

Version	Date	Author(s)	Comments
001	2/10/2023	Yesenia	Report

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.

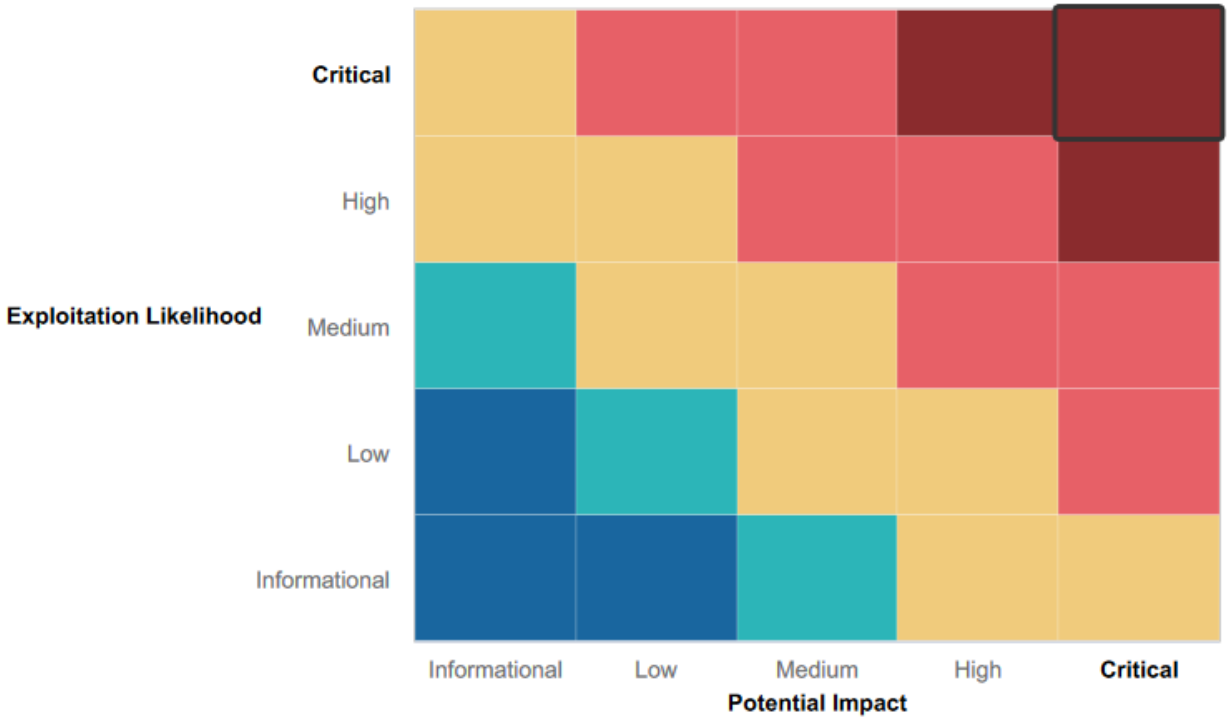
Executive Summary of Findings

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.

- Some input fields on the Rekall website used input validation making it difficult to exploit
- It was difficult to search for exploits that would work even if an exploit was an excellent module ranking

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.

- Rekall web page is vulnerable to SQL injections, XSS, local file inclusion (LFI), PHP injection, and unrestricted view of HTML code that includes sensitive data.
- Apache server is out-of-date with a Struts vulnerability.
- Sensitive data exposed on github site
- SLMail server is vulnerable to exploits which allow access to shell
- FTP port is open

Executive Summary

Throughout one week, our team used a variety of tools to successfully find vulnerabilities within Rekall Corporation's web application, linux and windows machines. These vulnerabilities range from critical to low and there were many critical vulnerabilities found through our penetration testing. This means that these critical vulnerabilities could have a potentially catastrophic impact on the revenue or reputation of Rekall.

We began by testing Rekall's web application for vulnerabilities. We found that it was vulnerable to XSS, SQL injections, local file inclusions and more. We were also able to view sensitive data and traverse through pages that are not supposed to be available for the public.

The next step we took was to conduct an open source investigation on total rekall.xyz. We used an online tool called Domain Dossier to view the IP address of totalrekall.xyz, we also used who.is to conduct a domain search on the target, and we used cert.sh to view the SSL certificates. Once that information was found we used a linux tool called nmap to scan Rekall's subnet of IP addresses to discover open ports and detect operating systems on the machines found, which were 192.168.13.10, 192.168.13.11, 192.168.13.12, 192.168.13.13, and 192.168.13.14. Once the targets were known, we used a tool called Nessus to scan for vulnerabilities and exploit them with a tool called Metasploit.

On our final day of testing, we turned to Windows machines on Rekall's network. On Rekall's github repository we discovered credentials for a WIndows login publicly exposed. We also found additional possible vulnerabilities by performing an Nmap on 172.22.117.20 and successfully found an exploit for it. Additional methods and several vulnerabilities were found and are explained in detail below in the Vulnerability Findings section.

We recommend that Rekall Corporation review carefully and immediately fix critical vulnerabilities found. These findings are an immediate threat to business processes and could damage the functionality of business. As mentioned previously, below are details of the vulnerabilities found and suggested remediations to fix them.

Summary Vulnerability Overview

Vulnerability	Severity
XSS Reflected	Critical
XSS Stored	Critical
Local File Inclusion	Critical
SQL Injection	Critical
Command Injection	Critical
HTML Credential Exposure	Critical
PHP Injection	Critical
Directory Traversal	Critical
Apache Struts (CVE-2017-5638)	Critical
Scheduled Tasks	Critical
Shellshock	Critical
Drupal (CVE-2019-6340)	High
Nessus Scan	High
Apache Tomcat RCE (CVE-2017-12617)	High
Exposed Data on Github	High
SLmail	High
FTP Enumeration	High
SSH and Sudo (CVE-2019-14287)	High
Open source exposed data	Medium
Open source exposed data- Certificate Search	Medium
Nmap Scan- Linux	Medium
Server Details Exposed	Medium
Sensitive Data Exposure	Low

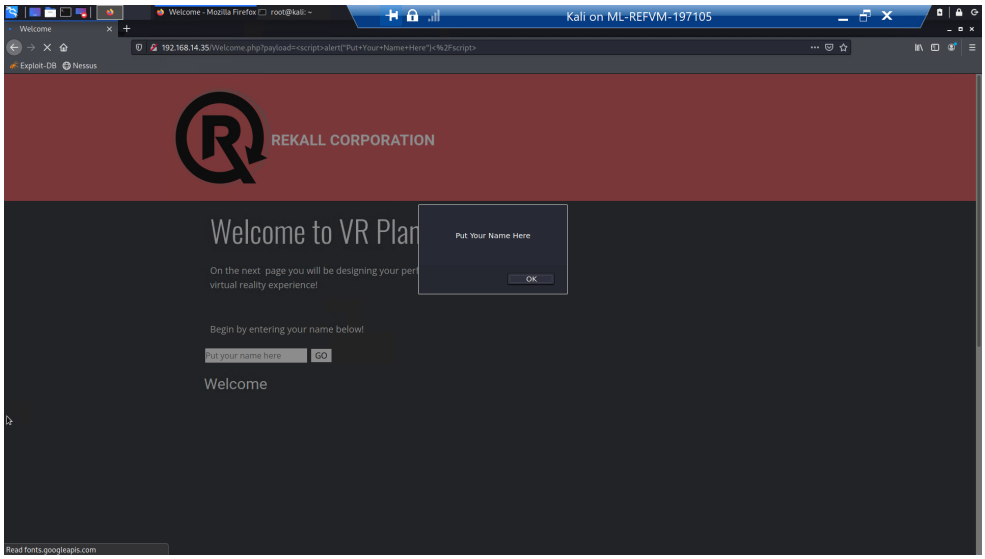
The following summary tables represent an overview of the assessment findings for this penetration test:

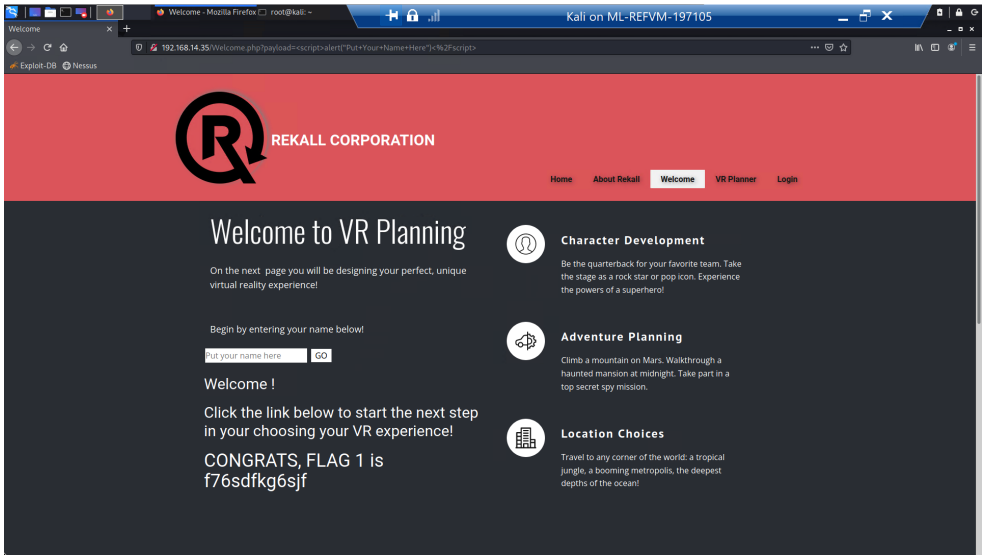
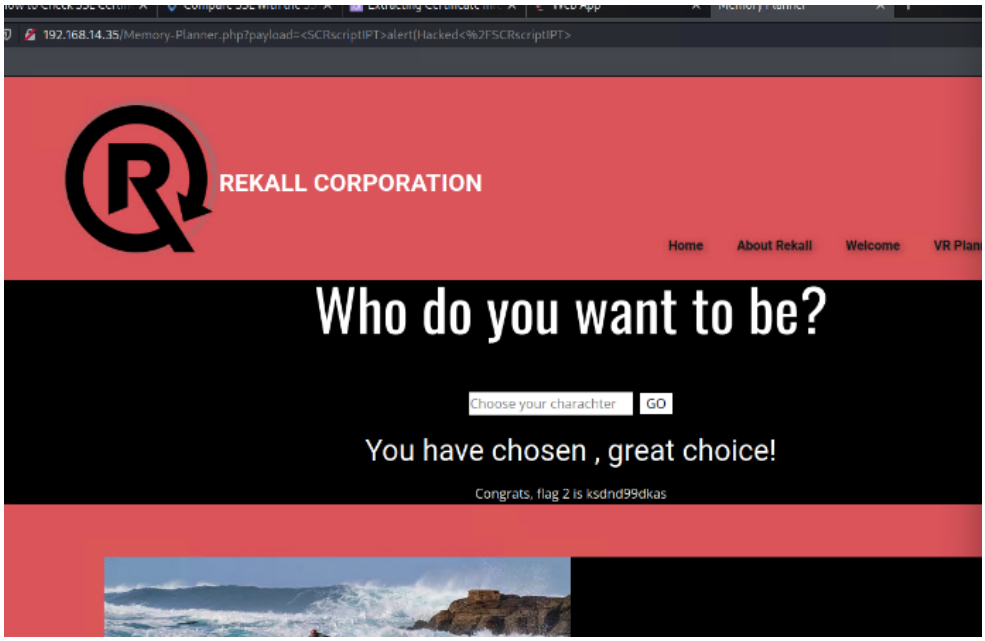
Scan Type	Total
Hosts	192.168.14.35 34.102.136.180 192.168.13.10 192.168.13.11 192.168.13.12 192.168.13.13

	192.168.13.14 172.22.117.10 172.22.117.20
Ports	21(FTP), 22(SSH), 25(SMTP), 80(HTTP), 110(TCP)

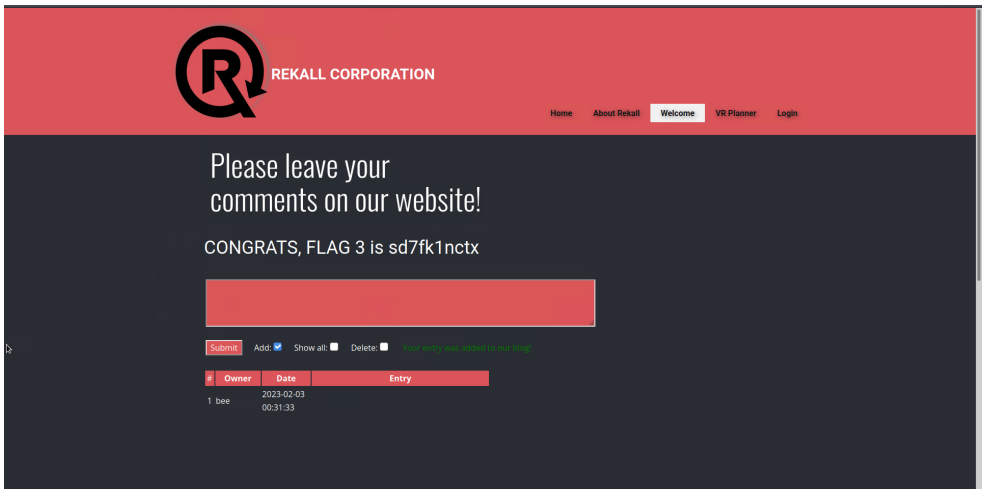
Exploitation Risk	Total
Critical	11
High	7
Medium	4
Low	1

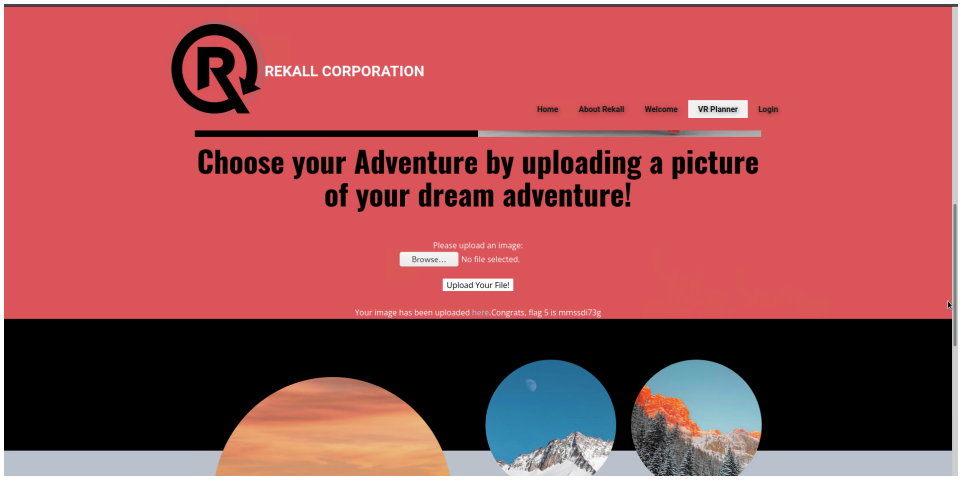
Vulnerability Findings

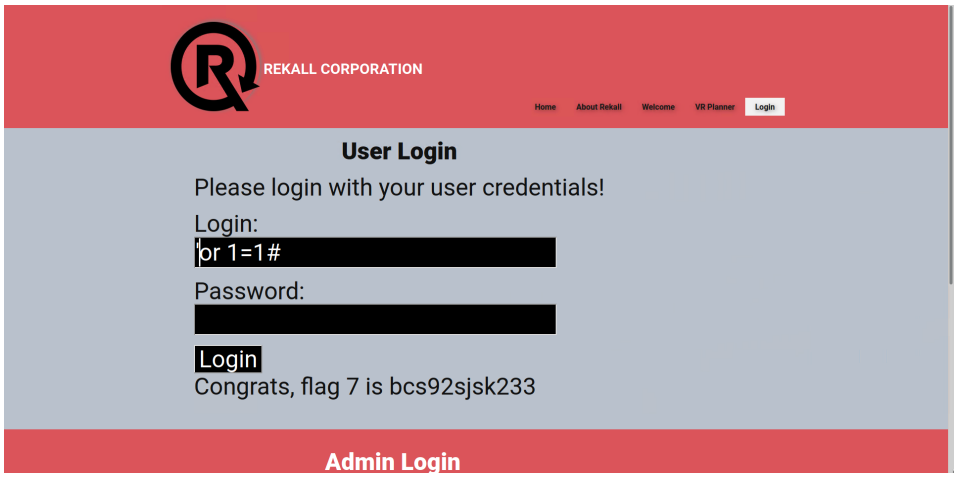
Vulnerability 1	Findings
Title	XSS Reflected- OWASP 7 Identification and Authentication Failures
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	Critical
Description	We were able to successfully insert alerts into the input fields for, “Begin by entering your name below” field on welcome.php and on the Memory-Planner.php page using cross site scripting. On the welcome page we were able to enter <script>alert("Put Your Name Here")</script>. On the Memory Planner Page, there was some user validation so we changed it slightly to <SCRIPscriptT> instead of <script>.
Images	

	 
Affected Hosts	192.168.14.35
Remediation	• Set up input validation

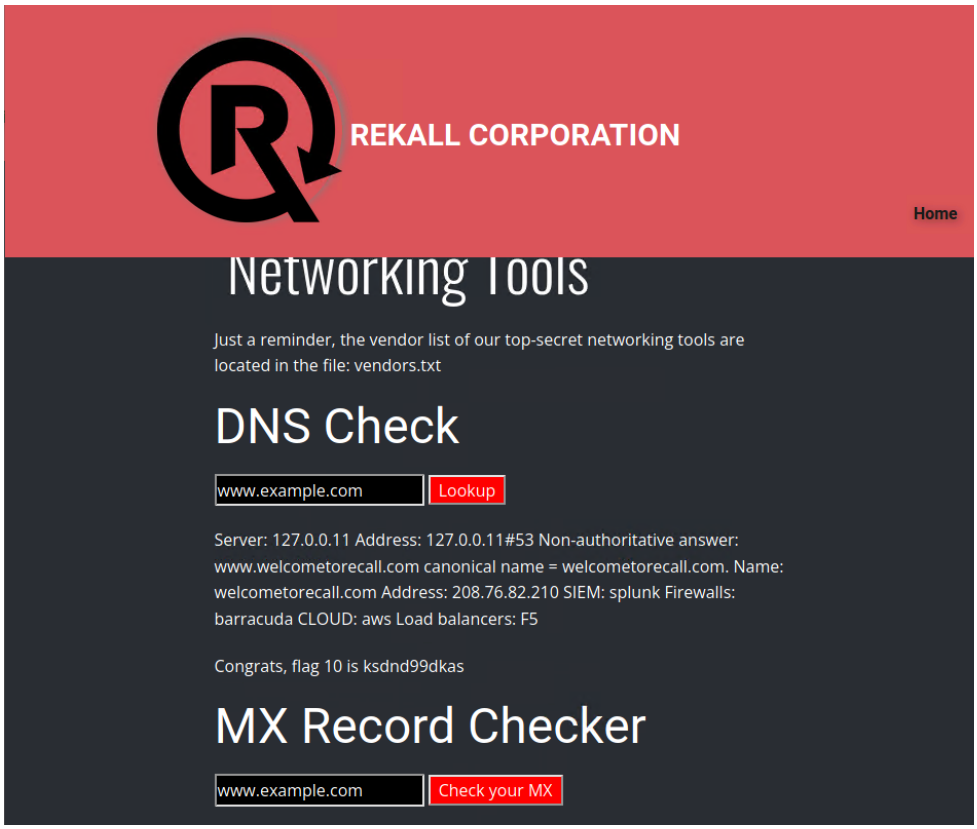
Vulnerability 2	Findings
Title	XSS Stored- OWASP 7 Identification and Authentication Failures
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	**Critical**
Description	We were able to successfully complete an XSS into the input field for comments.php by also entering <script>alert("Put Your Name Here")</script>.

Images	
Affected Hosts	192.168.14.35
Remediation	• Set up input validation

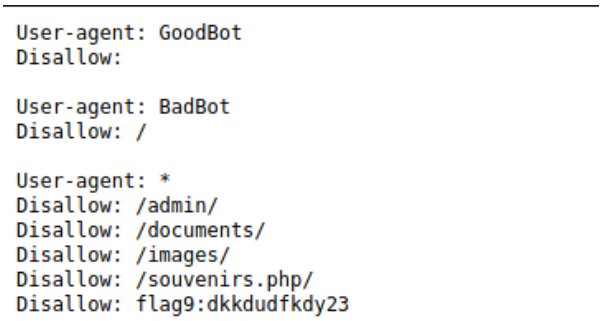
Vulnerability 3	Findings
Title	Local File Inclusion- OWASP 1
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	**Critical**
Description	We were able to conduct a Local File Inclusion and uploaded a malicious php file on the Memory Planner page
Images	
Affected Hosts	192.168.14.35
Remediation	• The application should use allow listing to ensure that only a specific file type like .jpg is uploaded, and not an arbitrary script file type, such as .php

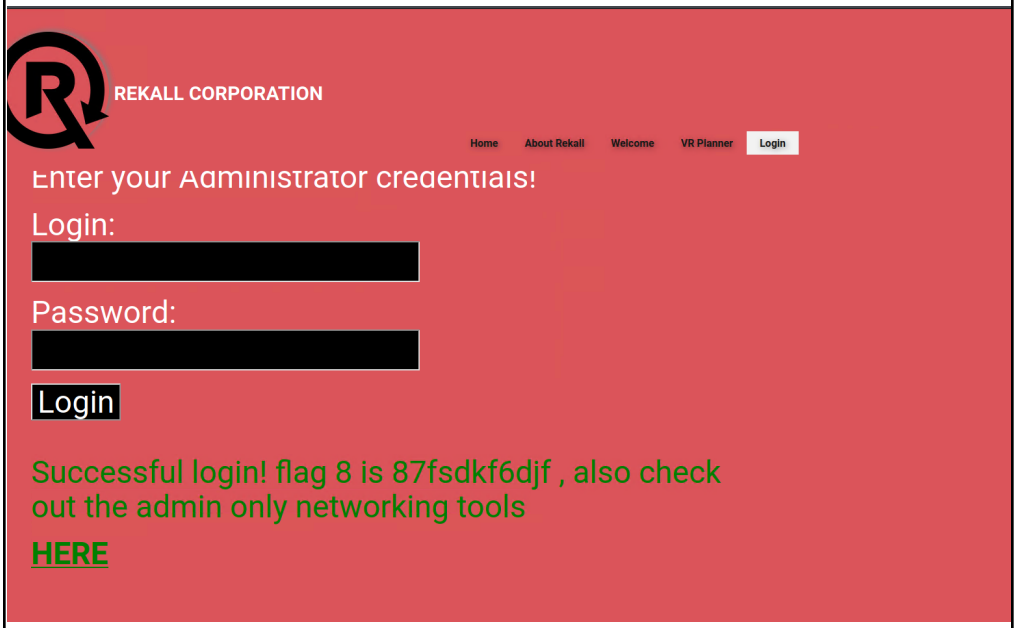
Vulnerability 4	Findings
Title	SQL Injection- OWASP 3 Injection
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	**Critical**
Description	While we accessed the Login page, "'or 1=1#" was entered on the login field which was successful. An SQL injection can allow an attacker to modify the statement's meaning or to execute arbitrary SQL commands.
Images	
Affected Hosts	192.168.14.35
Remediation	• Apply input validation code logic to the client- and server-side code.

Vulnerability 5	Findings
Title	Command Injection- OWASP 3 Injection
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	**Critical**
Description	On the Networking page we were able to successfully conduct a command injection by typing in "www.welcometorecall.com && "

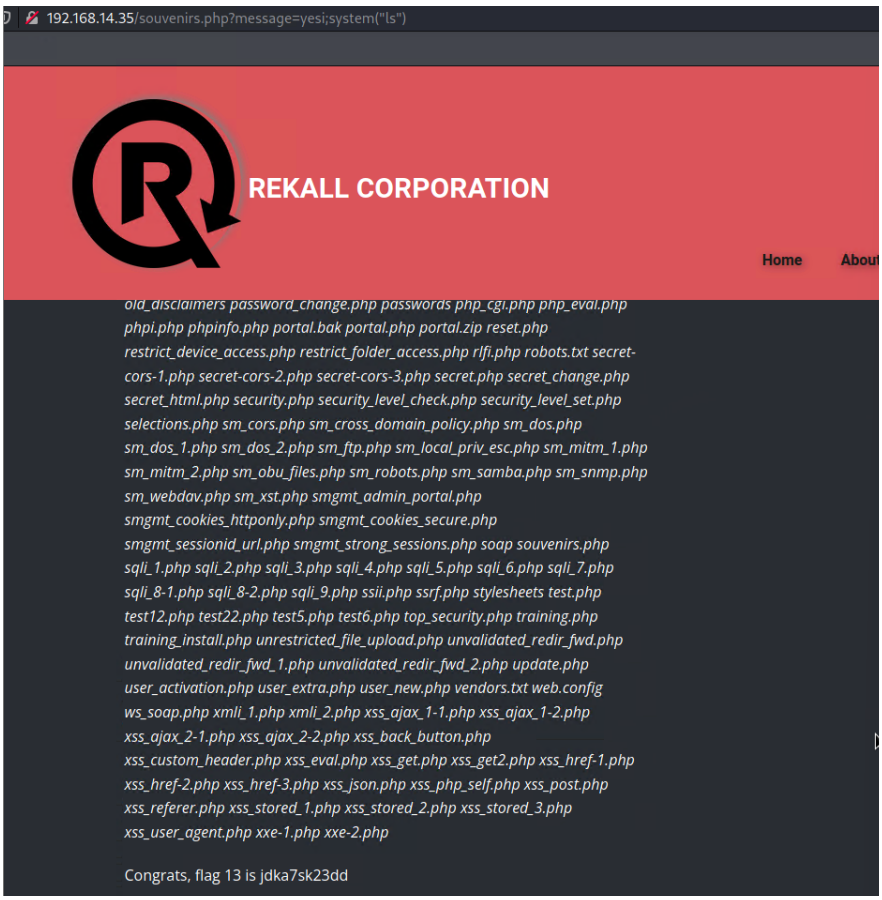
Images	
Affected Hosts	192.168.14.35
Remediation	• set up input validation

Vulnerability 6	Findings
Title	Sensitive Data Exposure
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	**Low**
Description	We were able to navigate and view robots.txt by going to 192.168.14.35/robots.txt and consequently find other pages within 192.168.14.35.

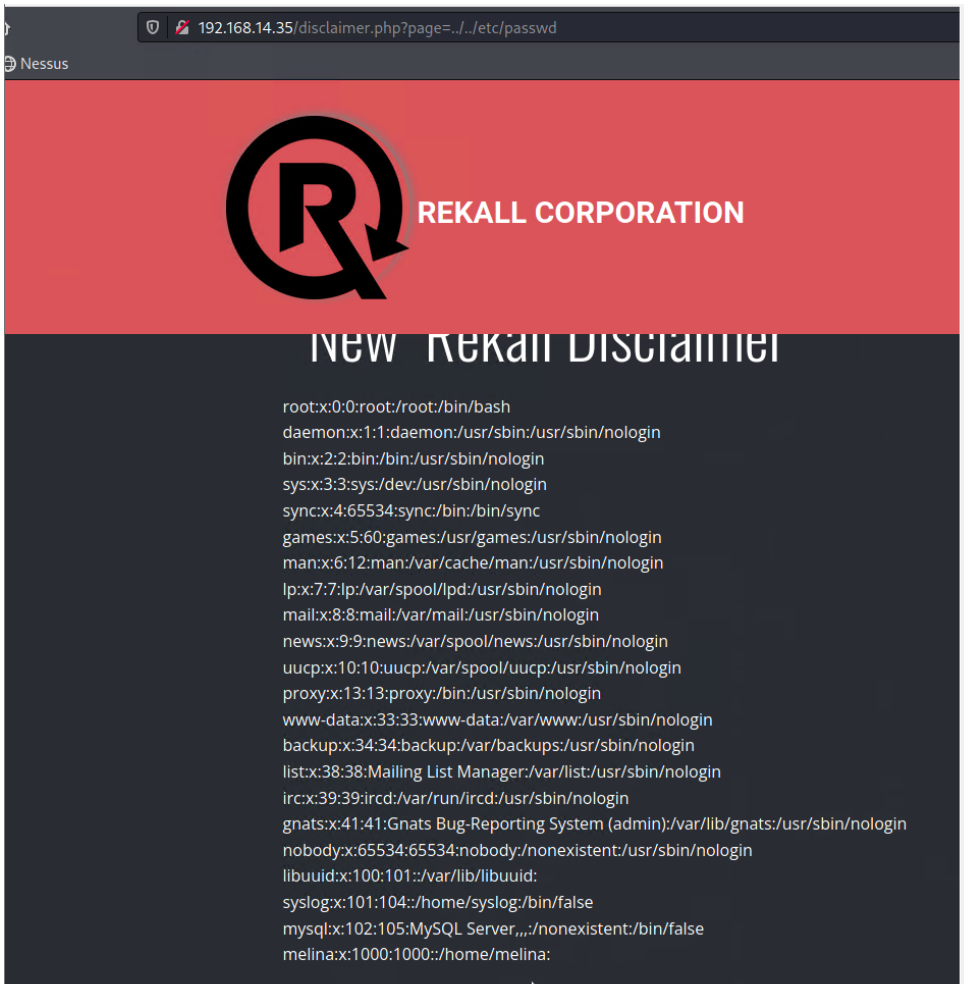
Images	
Affected Hosts	192.168.14.35
Remediation	• Disallow Directories, Not Specific Pages so attackers don't access the specific pages you don't want them to

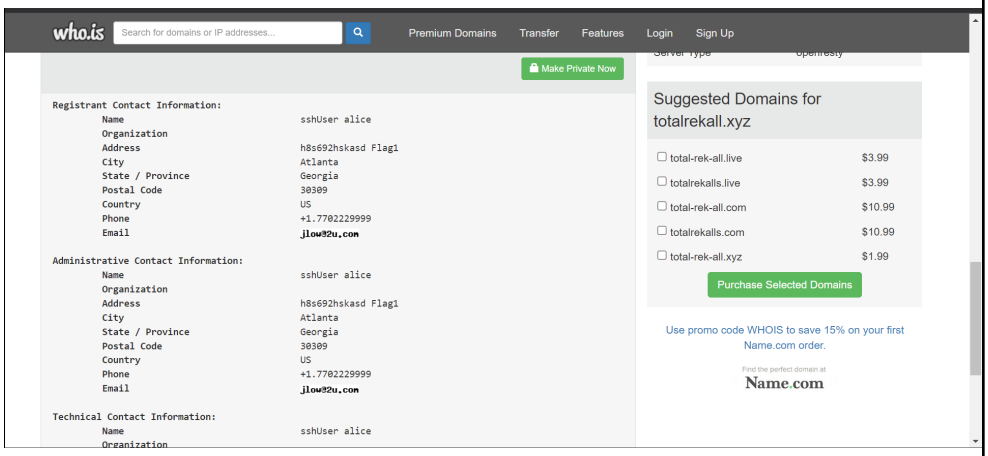
Vulnerability 7	Findings
Title	HTML Credential Exposure- OWASP 1 Broken Access Control
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	**Critical**
Description	On the Login page, we were able to view the page source and explore the HTML. We were able to find the administrator credentials and login successfully using username: doug quip and password kuato.
Images	

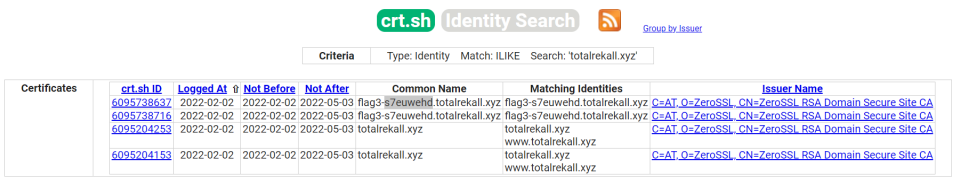
Affected Hosts	192.168.14.35
Remediation	• Delete sensitive information from the HTML page • Implement 2-factor authentication

Vulnerability 8	**Findings**
Title	PHP Injection- OWASP 3 Injection
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	**Critical**
Description	We were able to inject a malicious php onto the Souvenirs Page we discovered earlier. More specifically we loaded a php into the URL by adding system("ls") and this allowed us to see the list of all pages included in the website. Allowing the upload of a .php script file could result in malicious scripts being run against the database to modify or delete data and could possibly cause system outages.
Images	
Affected Hosts	192.168.14.35
Remediation	• The application should avoid incorporating user-controllable data into

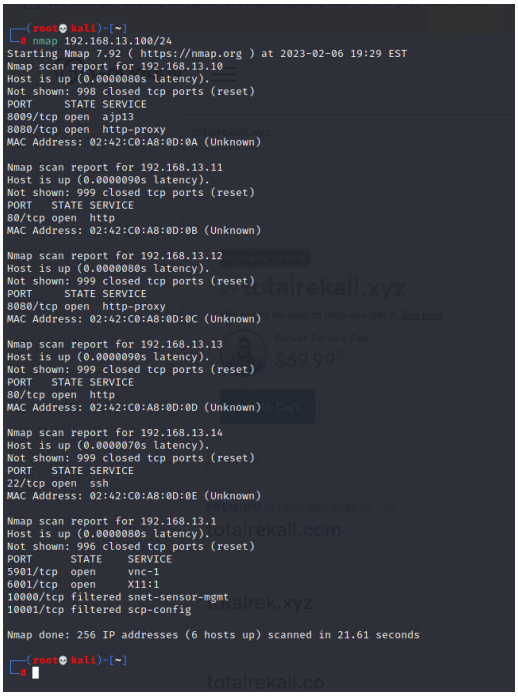
	operating system commands. ● Validation/Sanitization on User Input

Vulnerability 9	Findings
Title	Directory Traversal- OWASP 1 Broken Access Control
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	**Critical**
Description	On the Disclaimer Page we tested the directory traversal vulnerability and found that it would allow us to traverse to etc/passwd and view it.
Images	
Affected Hosts	192.168.14.35
Remediation	● Limiting user input when calling for files from the web application. ● Use input validation to limit the user's ability to modify the file being accessed. ● Web servers should run under a special service user account that only has access to that web folder.

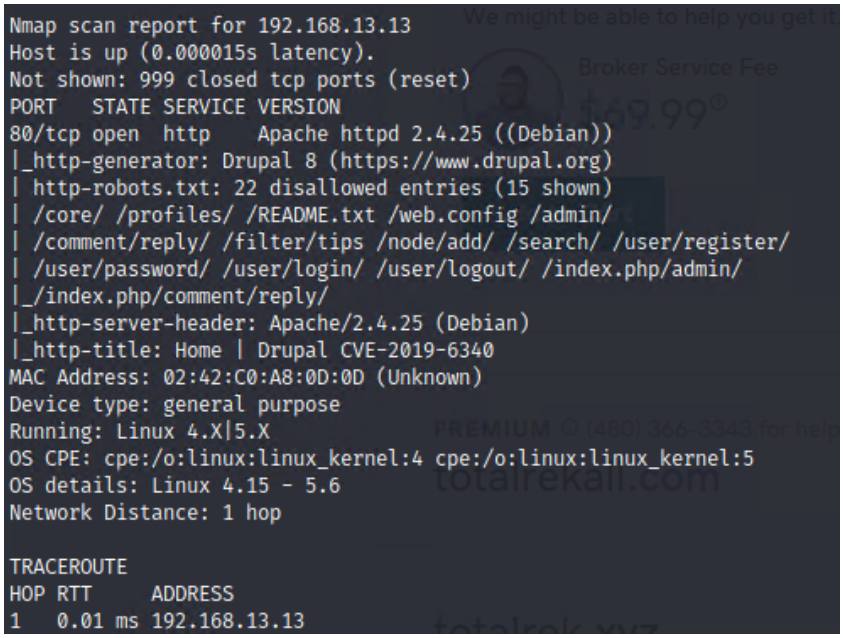
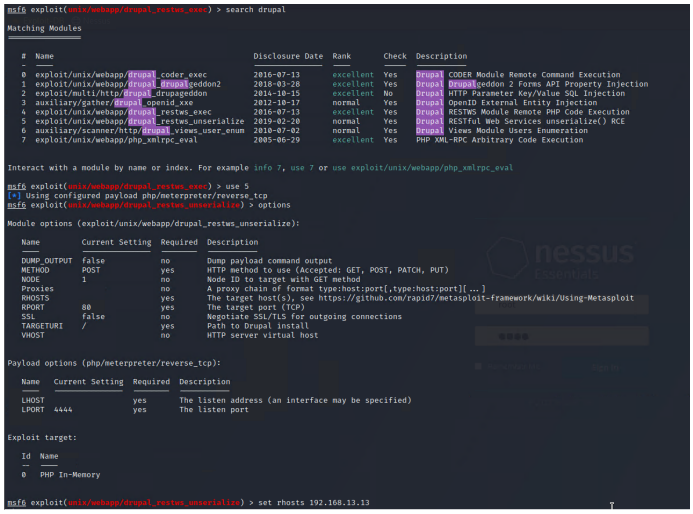
Vulnerability 10	Findings
Title	Open source exposed data
Type (Web app / Linux OS / WIndows OS)	Linux
Risk Rating	**Medium**
Description	On the who.is page we were able to view the information on the domain and it is useful to try and find other vulnerabilities and exploit them.
Images	
Affected Hosts	34.102.136.180
Remediation	• Ensure no sensitive data is being shared publicly, clean up WHOIS records

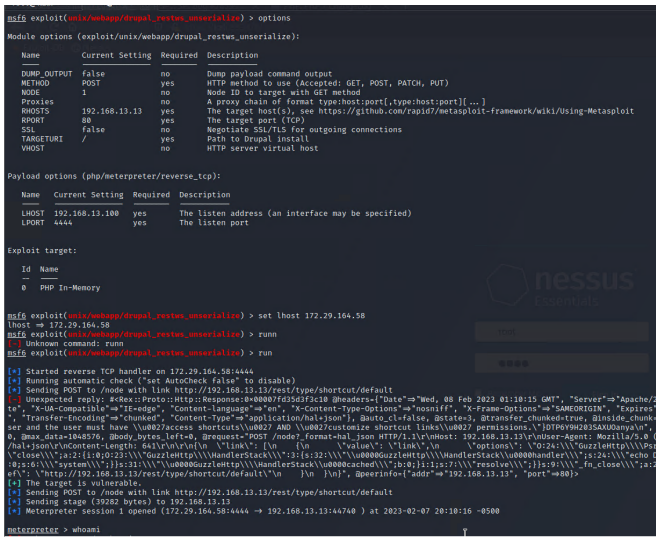
Vulnerability 11	Findings
Title	Open source exposed data- Certificate Search
Type (Web app / Linux OS / WIndows OS)	Linux
Risk Rating	**Medium**
Description	We searched for totalrekall.xyz on crt.sh to reveal sub - domain names
Images	
Affected Hosts	34.102.136.180
Remediation	• Use wildcard certificates instead so you can have a single certificate that is valid for all subdomains, in turn not revealing the sub-domain names in CT logs.

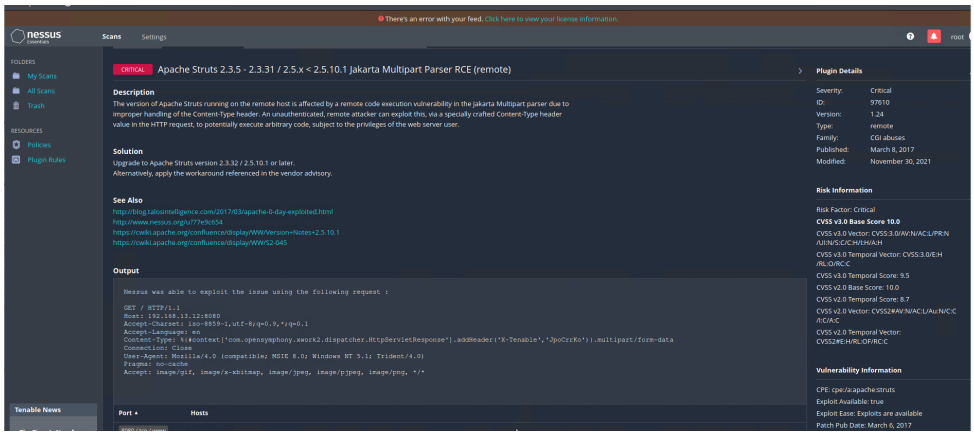
	• Secure your servers and application endpoints better if subdomains get listed

Vulnerability 12	Findings
Title	Nmap Scan- Linux
Type (Web app / Linux OS / WIndows OS)	Linux
Risk Rating	**Medium**
Description	An Nmap scan was ro on the subnet of the network, by running 192.169.13.100/24, and found 5 hosts running excluding the host we were scanning from.
Images	
Affected Hosts	192.168.13.10, 192.168.13.11, 192.168.13.12, 192.168.13.13, 192.168.13.14
Remediation	• Proactively scanning would provide the opportunity to find and fix vulnerabilities before attackers find them • Block and Slow Nmap with Firewalls • Detect Nmap Scans

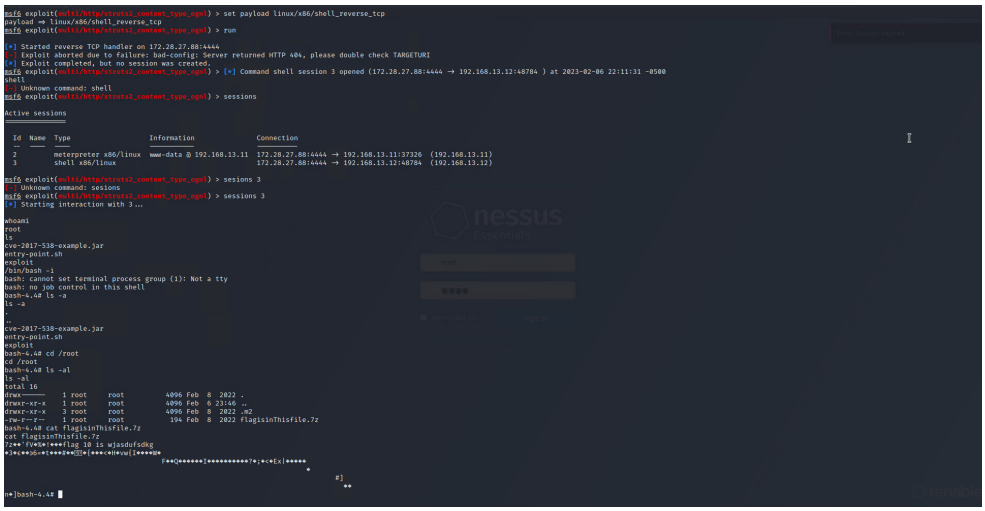
Vulnerability 13	Findings
Title	Drupal (CVE-2019-6340)
Type (Web app / Linux OS / WIndows OS)	Linux
Risk Rating	**High**

Description	The team ran an aggressive Nmap scan on the network, nmap -A 192.168.13.100/24, to find open ports and services running on each host. We found that the host 1992.168.13.13 was running Drupal 8 and that could be exploited through metasploit. After searching for Durpal exploits, we found unix/webapp/drupal_restws_unserialize which allowed us to get a meterpreter shell.
Images	 

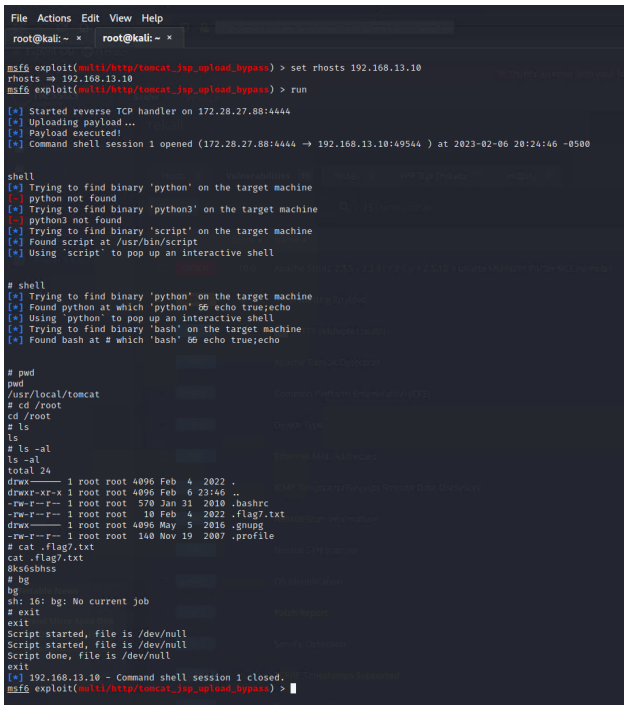
	
Affected Hosts	192.168.13.13
Remediation	• Upgrade Drupal to newer version

Vulnerability 14	Findings
Title	Nessus Scan
Type (Web app / Linux OS / WIndows OS)	Linux
Risk Rating	High
Description	A Nessus scan was run on host 192.168.13.12 and an Apache Struts critical vulnerability appeared.
Images	
Affected Hosts	192.168.13.12
Remediation	

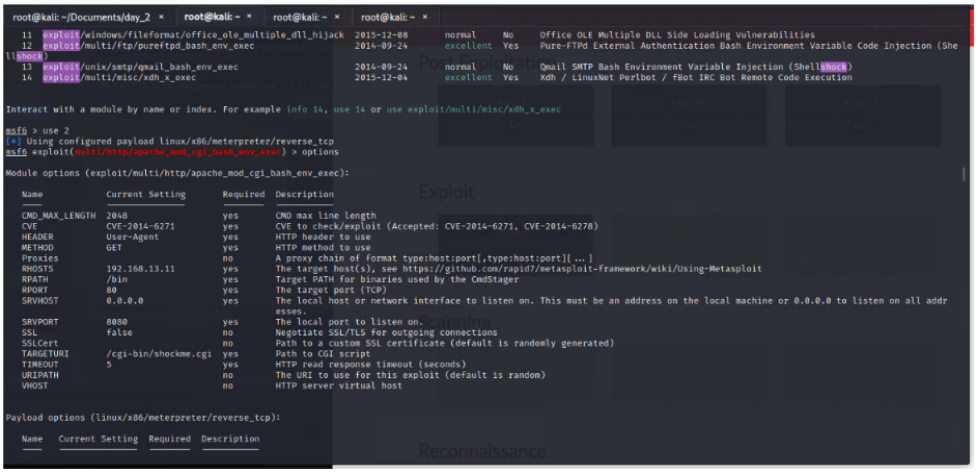
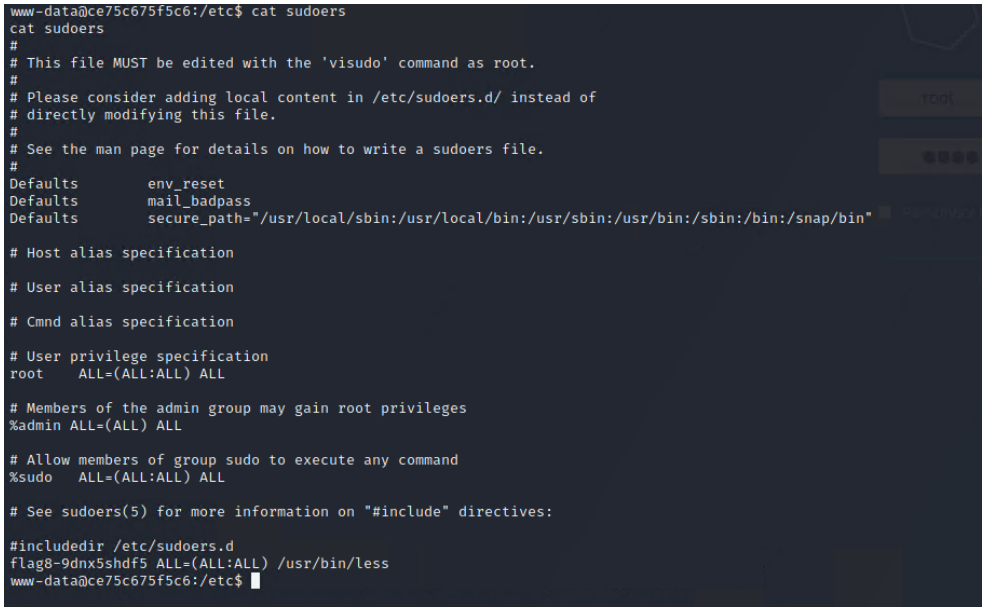
Vulnerability 15	Findings
Title	Apache Struts (CVE-2017-5638)

Type (Web app / Linux OS / WIndows OS)	Linux
Risk Rating	**Critical**
Description	The previous Nessus scan pinpointed that we needed to search for Struts exploits on Metasploit. We ended up successfully using the following exploit: multi/http/struts2_content_type_ognl and obtained a meterpreter shell.
Images	
Affected Hosts	192.168.13.12
Remediation	

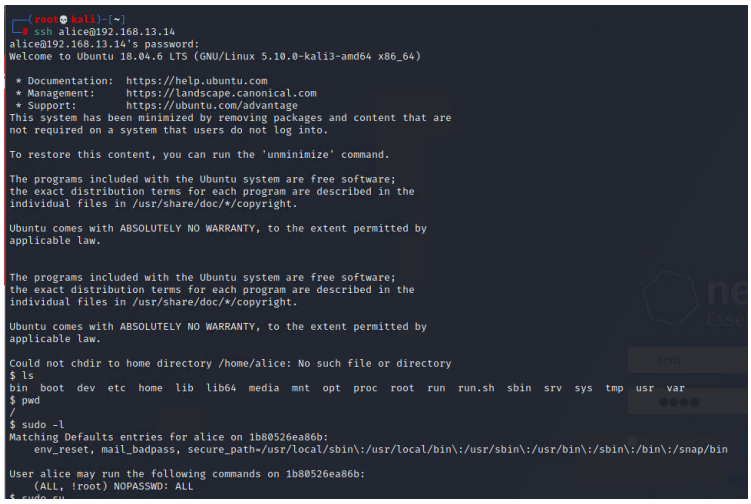
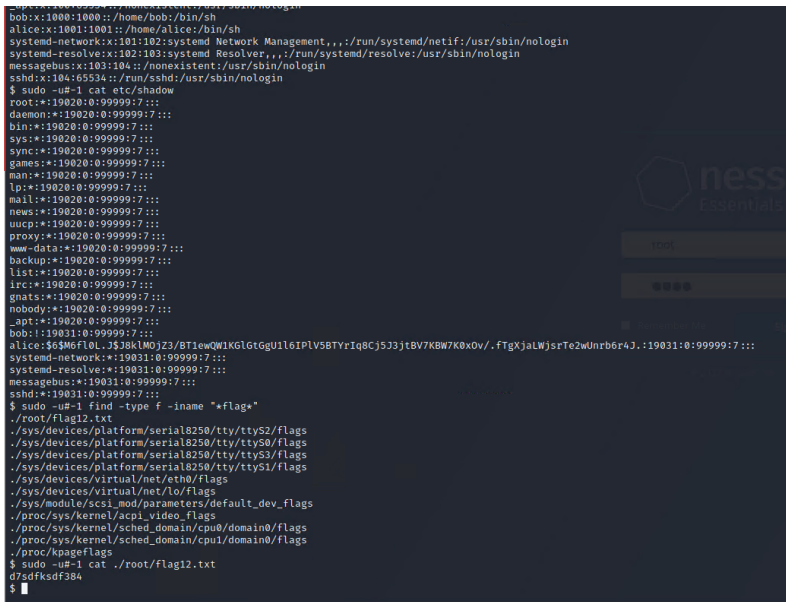
Vulnerability 16	**Findings**
Title	Apache Tomcat RCE (CVE-2017-12617)
Type (Web app / Linux OS / WIndows OS)	Linux
Risk Rating	**High**
Description	We referenced the aggressive Nmap scan run earlier and found that Apache Tomcat was running on host 192.168.13.10. Then using Metasploit we searched for exploits containing Tomcat and JSP. We then obtained a meterpreter shell by successfully running the following exploit: multi/http/tomcat_jsp_upload_bypass.

Images	
Affected Hosts	192.168.13.10
Remediation	• Update Tomcat to the latest version where the vulnerability is fixed.

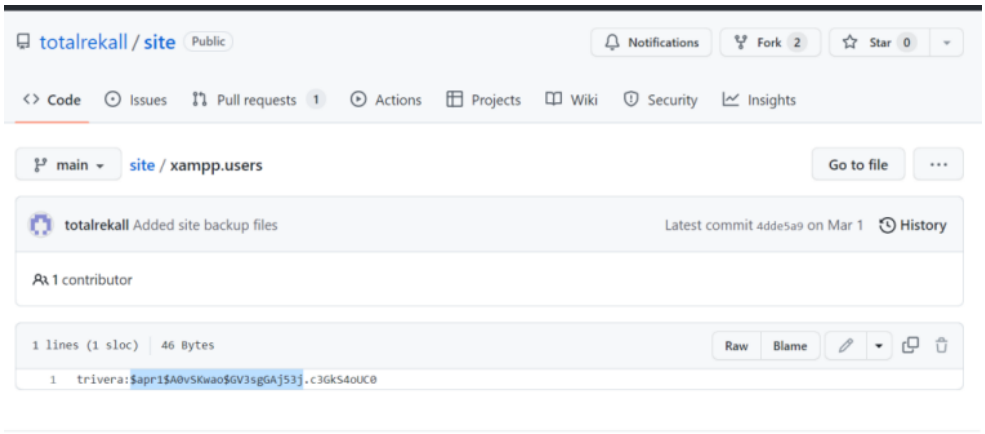
Vulnerability 17	**Findings**
Title	Shellshock
Type (Web app / Linux OS / WIndows OS)	Linux
Risk Rating	**Critical**
Description	On Metasploit msfconsole we searched Shellshock exploits and the exploit that was successful in returning a meterpreter shell was the exploit: exploit/multi/http/apache_mod_cgi_bash_env_exec

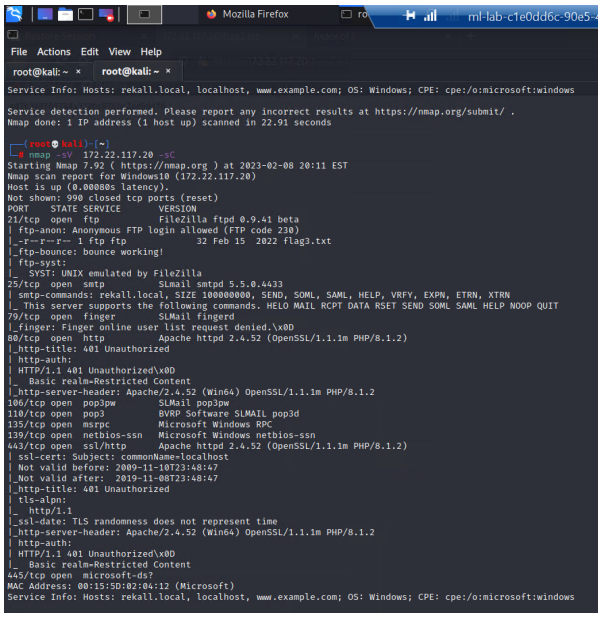
Images	 
Affected Hosts	192.168.13.11
Remediation	• Update the version of Bash • Keep servers up to date with the latest security updates

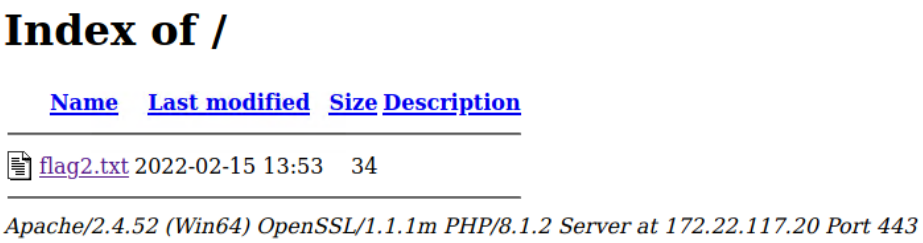
Vulnerability 18	Findings
Title	SSH and Sudo CVE-2019-14287
Type (Web app / Linux OS / WIndows OS)	Linux
Risk Rating	High
Description	The open source exposed data on who.is gave us a clue about being able to SSH into the server as alice. We quickly guessed the password since it was the same as the user. Then by checking sudo privileges we saw a vulnerability in the user's sudo privileges that we were able to exploit to gain root access.

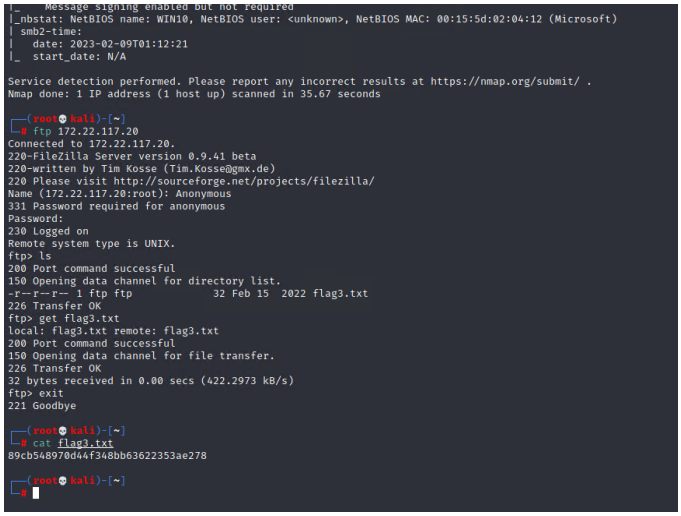
Images	 
Affected Hosts	192.168.13.14
Remediation	• Close port 22 • Alice needs to change weak password • Update Sudo so this vulnerability doesn’t exist

Vulnerability 19	Findings
Title	Exposed Data on Github
Type (Web app / Linux OS / WIndows OS)	Windows
Risk Rating	High
Description	On totalrekalls public GitHub I, we easily discovered an exposed hash for a user. We were then able to crack the hash and obtain credentials.

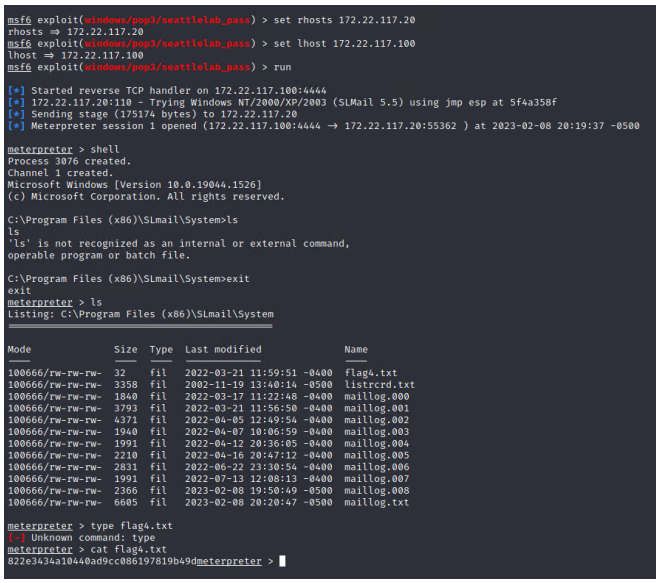
Images	
Affected Hosts	172.22.117.20
Remediation	• Use stronger and complex passwords • Remove password hashes and sensitive data from github page • Use strong salted hash functions

Vulnerability 20	Findings
Title	Server Details Exposed
Type (Web app / Linux OS / WIndows OS)	Windows
Risk Rating	**Medium**
Description	With the credentials obtained on Rekall's Github page we were able to log onto 172.22.117.20 because the Nmap scan revealed that port 80 was open.
Images	

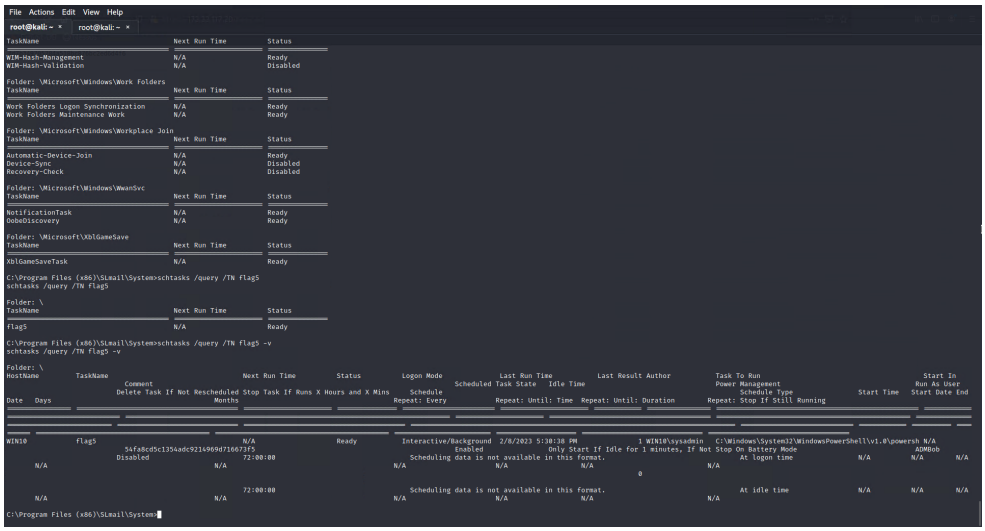
	
Affected Hosts	172.22.117.20
Remediation	• Use stronger and complex passwords • Remove password hashes and sensitive data from github page

Vulnerability 21	**Findings**
Title	FTP Enumeration
Type (Web app / Linux OS / WIndows OS)	Windows
Risk Rating	**High**
Description	The Nmap scan of 172.22.117.20 revealed that FTP was open on port 21. It was also revealed that FTP anonymous access was possible and did not require a password.
Images	
Affected Hosts	172.22.117.20
Remediation	• Close port 21 • Disable FTP and switch to FTPS or SFTP which are more secure

Vulnerability 22	**Findings**
Title	SLmail

Type (Web app / Linux OS / WIndows OS)	Windows
Risk Rating	**High**
Description	The Nmap scan of 172.22.117.20 revealed SLmail was running on SMTP port 25 and on TCP port 110. We then searched Metasploit for SLmail exploits and successfully got a meterpreter shell using: windows/pop3/seattlelab_pass
Images	
Affected Hosts	172.22.117.20
Remediation	• Only allowing access to the POP3 server from "inside" the firewall

Vulnerability 23	Findings
Title	Scheduled Tasks
Type (Web app / Linux OS / WIndows OS)	Windows
Risk Rating	**Critical**
Description	In the same meterpreter shell directly above, we looked at the scheduled tasks by using schtasks command then a schtasks /query and lastly schtasks /query /TN flag5 -v

Images	
Affected Hosts	172.22.117.20
Remediation	• Perform audits so toolkits like the PowerSploit framework contain PowerUp modules that can be used to explore systems for permission weaknesses in scheduled tasks that could be used to escalate privileges.