Rekall Corporation

Penetration Test Report

WCPT Security, 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	WCPT Security
Contact Name	William Cruickshank
Contact Title	Penetration Tester

Document History

Version	Date	Author(s)	Comments
001	2023-07-24	William Cruickshank	

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.

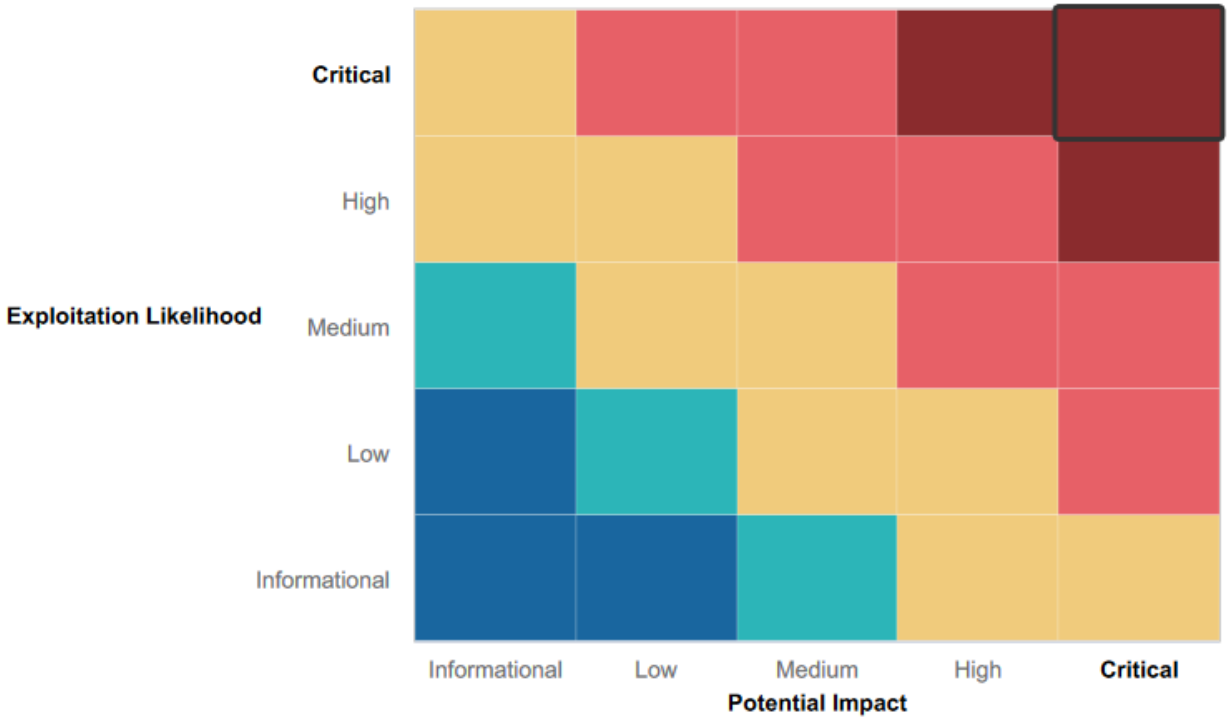
Executive Summary of Findings

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.

- Many fields on totalrekall.xyz had input validation, file names had to be manipulated in different ways to bypass the input validation.
- Many ports were scanned and were found closed and invulnerable.
- Must use extreme nMap or Zenmap searches.

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.

- Able to access administrative credentials through the source page on Login.php page.
- Able to access vendor information through Networking.php page.
- Used existing modules through metasploit to gain access to network machines.

Executive Summary

WCPT Security was able to achieve many objectives outlined in the scope of work. I was able to locate and exfiltrate sensitive information from totalrekall.xyz as well as from the Linux and Windows machines. These tests revealed seven system vulnerabilities, the majority of these vulnerabilities are from areas of the website that allow malicious data to be entered or uploaded. These vulnerabilities are extremely dangerous for Rekall's system and customer information, as well as Rekall's reputation. These vulnerabilities could lead to the possibility for customer data to be stolen and for system data to be changed or deleted if ever conducted by a hacker. In one of the vulnerabilities I was even able to expose administrative credentials through command injection. Because the Apache servers have not been recently updated I was able to find an exploit that permitted access to user credential files from the Linux machine. I was also able to locate an exposed password on a GitHub repository under Rekall's name which I was then able to use to infiltrate the Windows10 machine. From Rekall's public website it did not take much to find hidden files that help to direct online robots. I would highly recommend making some adjustments to these settings to avoid system breaches and attacks.

Summary Vulnerability Overview

Vulnerability	Severity
Cross Site Scripting (XSS)	Critical
Local File Inclusion (LFI)	Critical
Command Injection	Critical
Apache Struts 2.3.5-2.3.31/2.5x<2.5.10.1 Jakarta Multipart parser RCE (remote)	Critical
Exposed data on totalrekall Public GitHub Repository	High
PHP Object Injection	Medium
Sensitive Data Exposure	Low

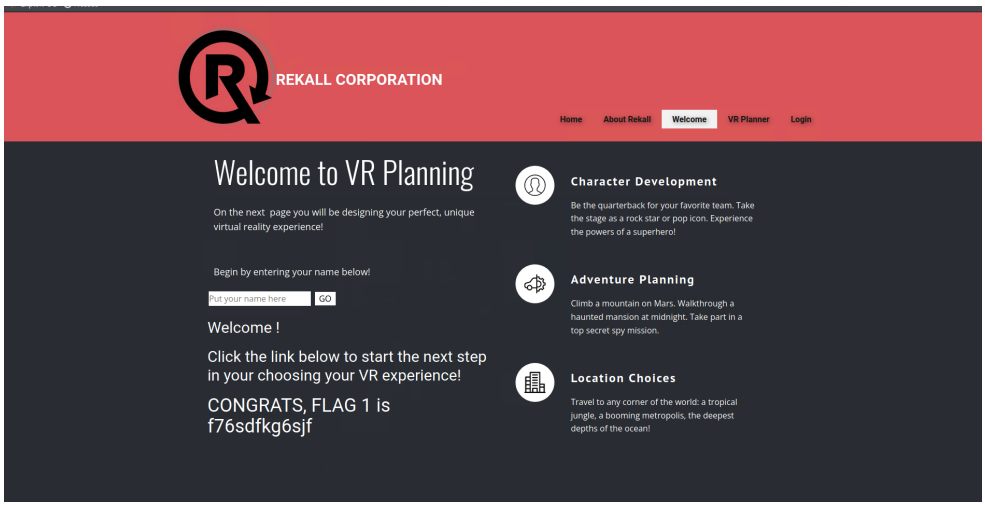
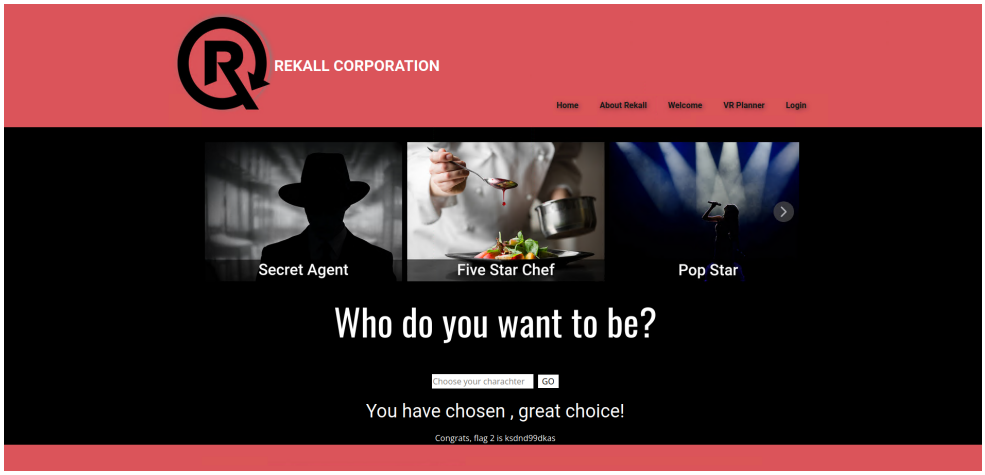
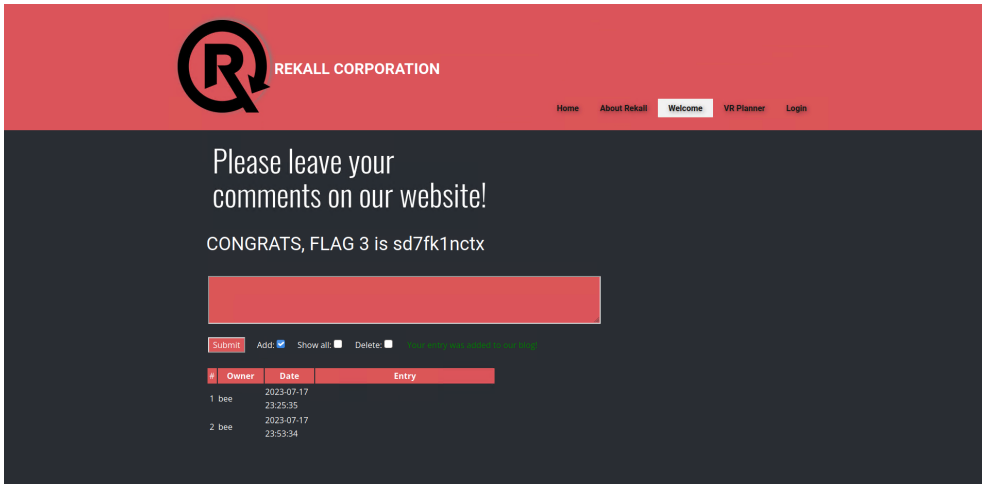
The following summary tables represent an overview of the assessment findings for this penetration test:

Scan Type	Total
Hosts	34.102.136.180 – totalrekall.xyz 192.168.13.10 - Linux 192.168.13.11 - Linux 192.168.13.12 - Linux 192.168.13.13 - Linux 192.168.13.14 - Linux 192.168.13.1 – Linux 172.22.117.20 – Windows10 172.22.117.10 – Windows Domain Controller 172.22.117.100 – Windows host
Ports	80 (HTTP) 21(FTP), 25(SMTP), 110 (POP3), 135 (RPC), 8009 (TCP), 8080

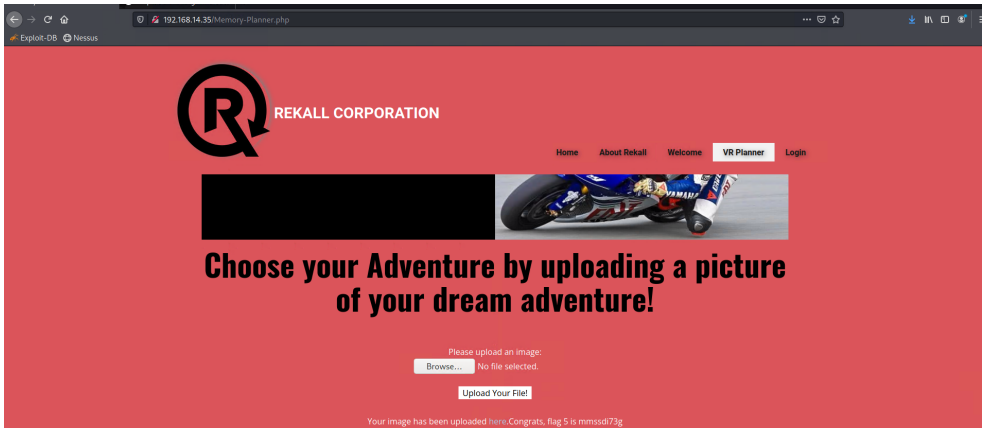
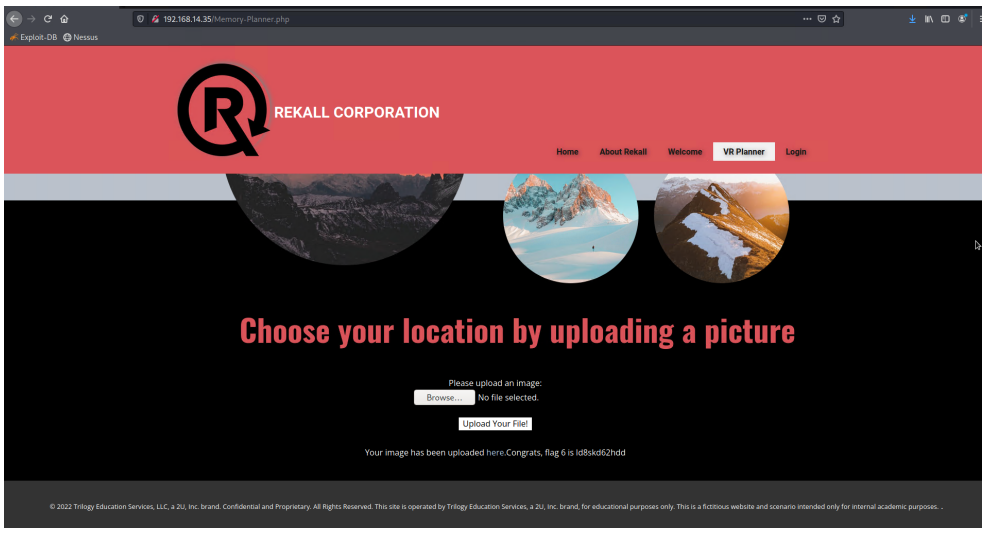
Exploitation Risk	Total
Critical	4
High	1
Medium	1
Low	1

Vulnerability Findings

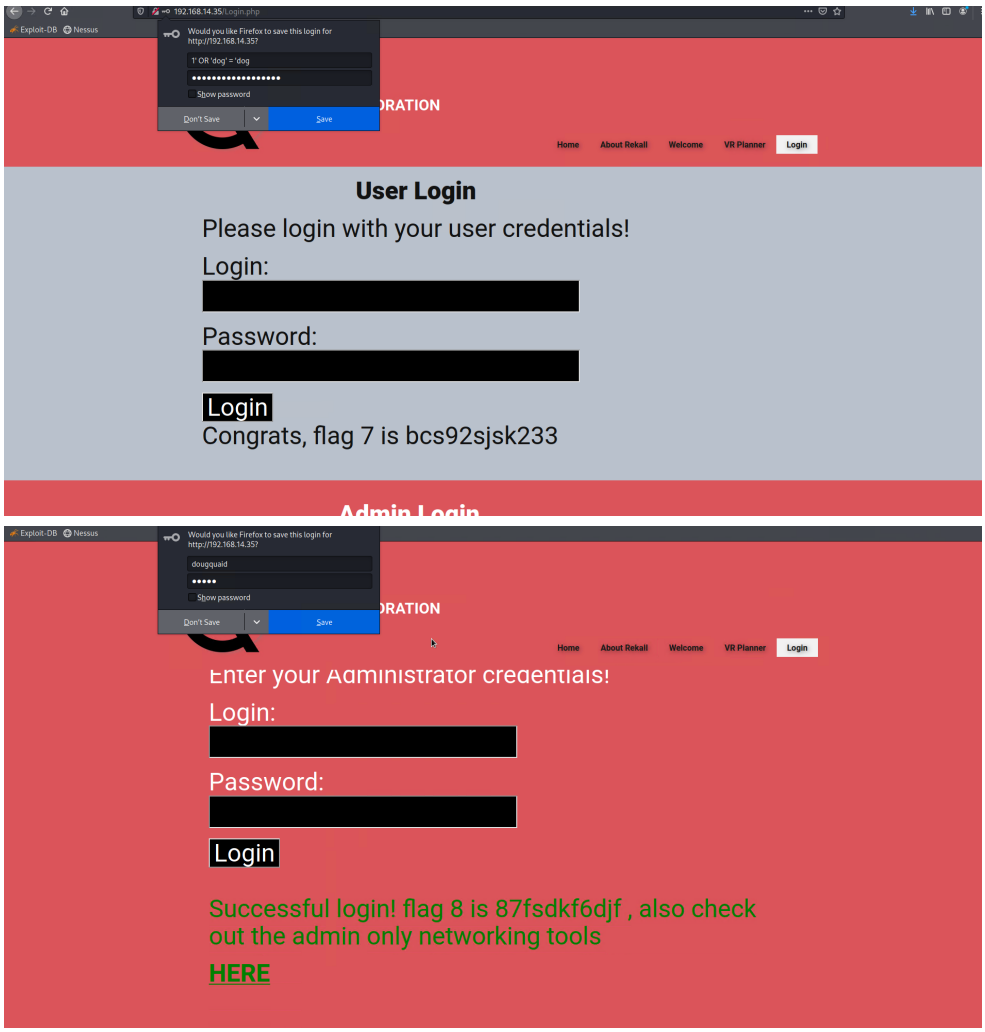
Title	Cross Site Scripting (XSS)

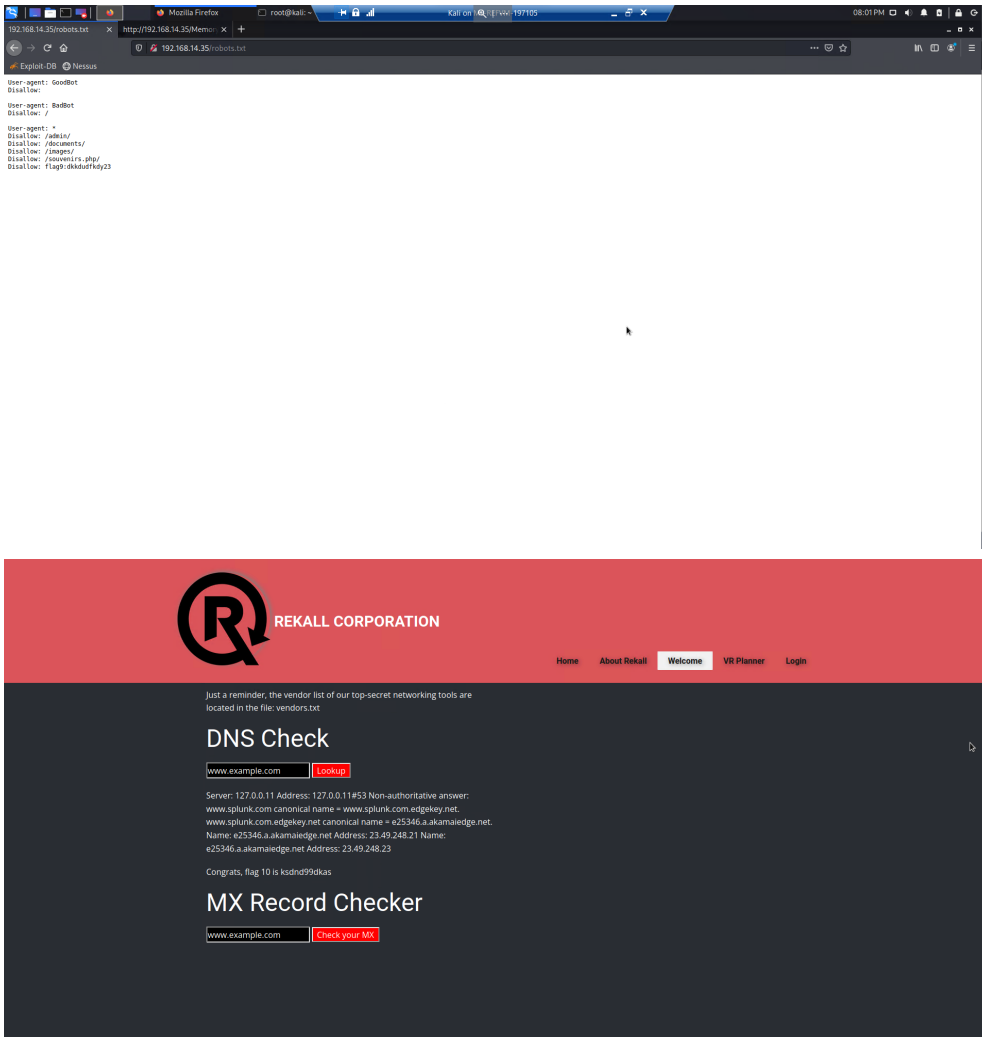
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	**Critical**
Description	I was able to successfully insert alerts into the input fields on totalrekall.xyz. For the input field “Put your name here” I used the XXS payload "<script>Hello!</script>". For the input field “Choose your character” I used the XXS payload "<SCRscriptIPT> Secret Agent </SCRscriptIPT>". For the input field “Comments” I used the XXS script "<script>alert"Hello!"</script>”.
Images	  

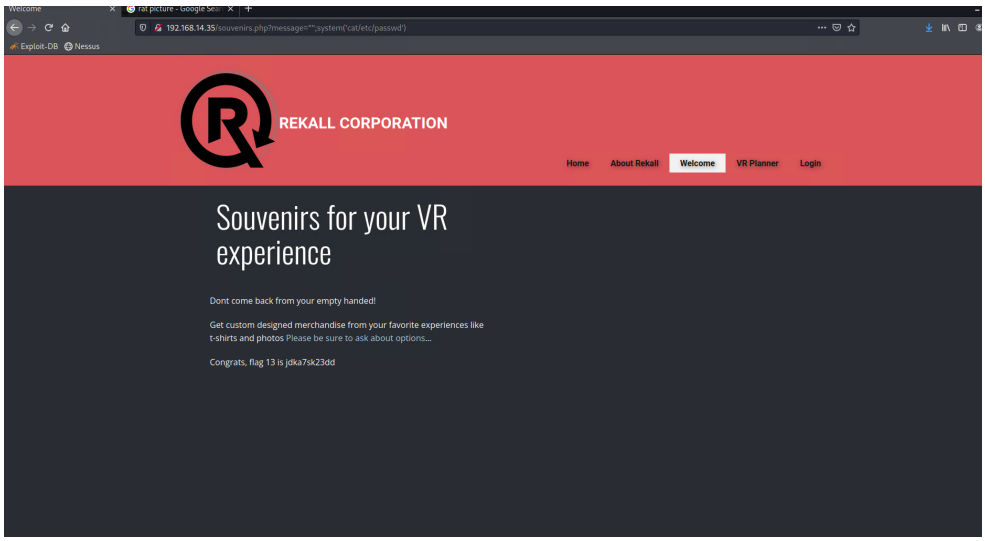
Affected Hosts	34.102.136.180 – totalrekall.xyz
Remediation	Input validation and sanitization Limit inline scripts Secure cookies and session handling

Title	Local File Inclusion (LFI)
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	**Critical**
Description	I was able to successfully insert a .php file into the input field in the second field on the Memory Planner page. I used a file named “hello.php” to manipulate the field. I then continued to the field below that and also manipulated that field by uploading a file named “hello.jpg.php”
Images	 
Affected Hosts	34.102.136.180 – totalrekall.xyz
Remediation	Input validation and sanitization Whitelisting and Identifier-based access

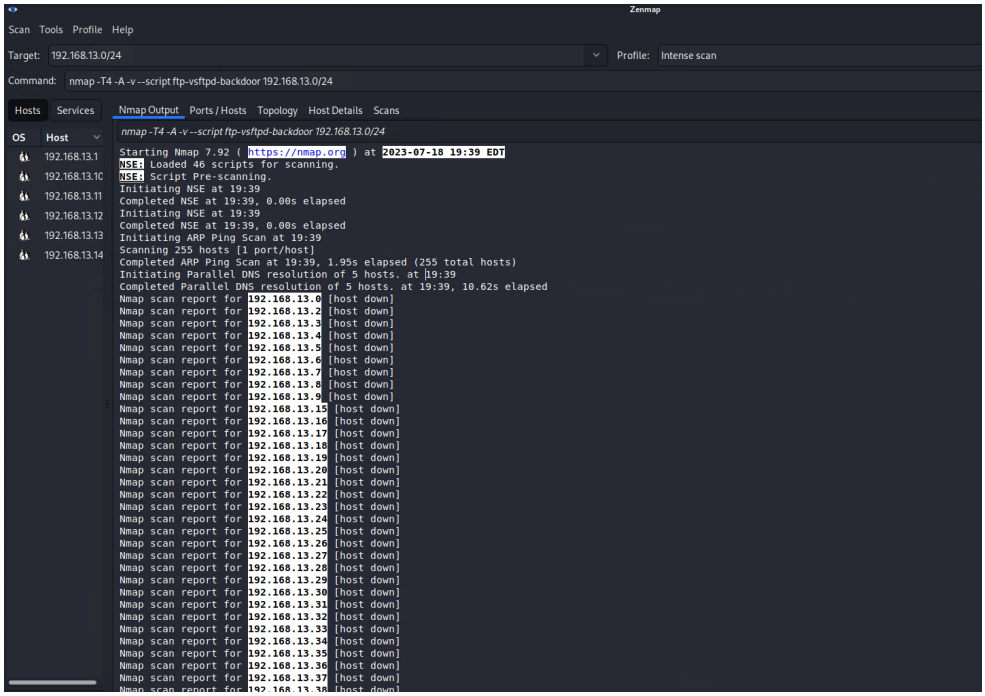
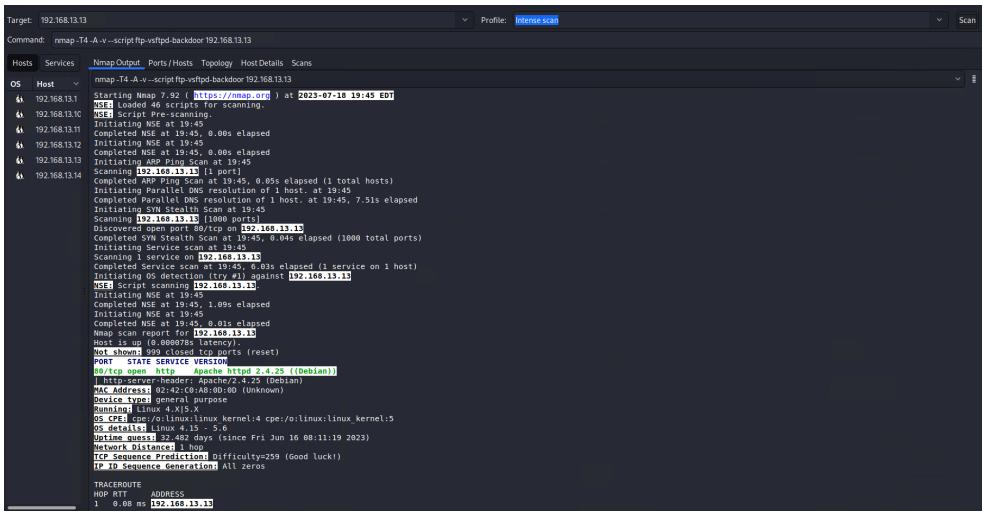
	Web Application Firewall (WAF)

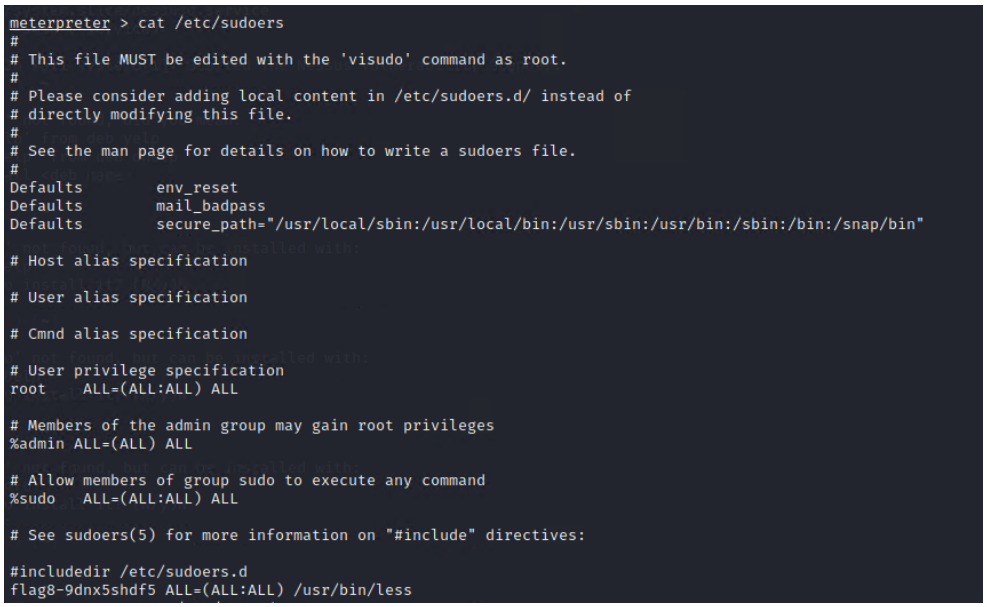
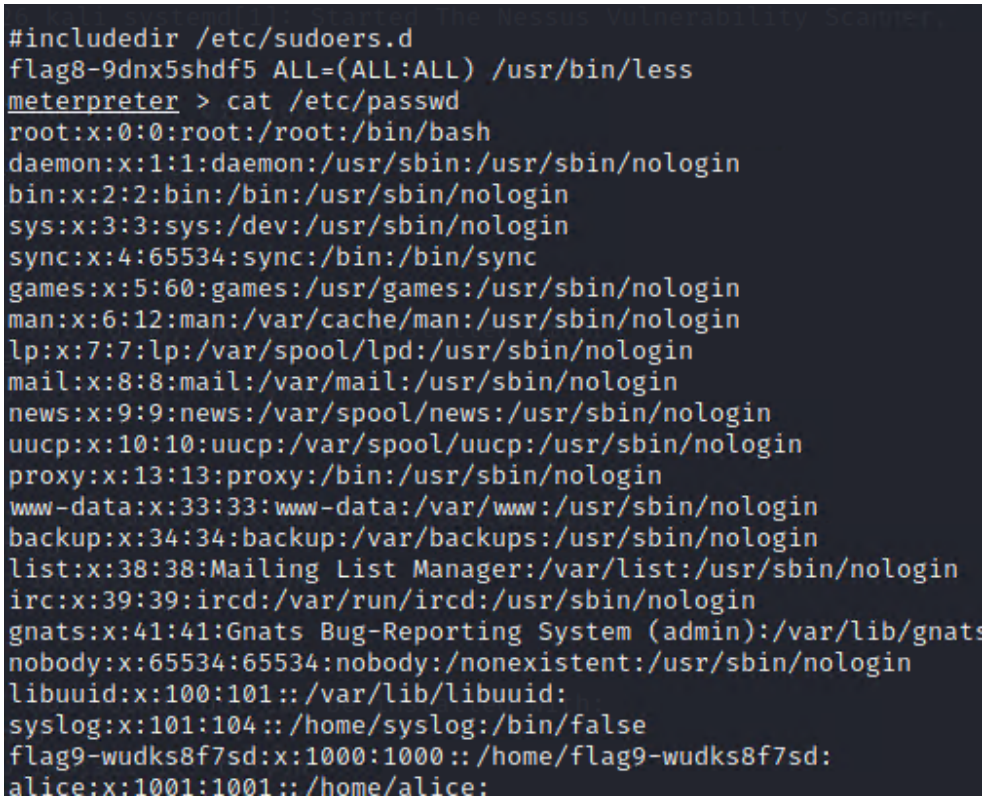
Title	Command Injection
Type (Web app / Linux OS / Wlndows OS)	Web app
Risk Rating	**Critical**
Description	I was able to successfully inject a command ("1' OR 'dog' = 'dog") inside the login and password field to manipulate the field and get login access. Once I gained login access I viewed the source page on the Login.php page and found credentials (Login:"dougquaid" Password: kuato). I used those to login.
Images	
Affected Hosts	34.102.136.180 – totalrekall.xyz
Remediation	Input validation and sanitization Use parameterized queries Escape special characters

Title	Sensitive Data Exposure
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	**Low**
Description	I was able to manipulate the search field to add “robots.txt” to the end of the IP Address which directed me to the robots.txt page. I was then able to reveal the robots exclusion standard settings for the website. From the Networking.php page I found vendors.txt which contains vendor info. From there I found that the SIEM was splunk. I then went back to the Networking.php where I did a DNS check on www.splunk.com.
Images	
Affected Hosts	34.102.136.180 – totalrekall.xyz
Remediation	Input validation and sanitization Access control and Geolocation restrictions Regularly monitor network traffic to detect any suspicious activities related to IP manipulation.

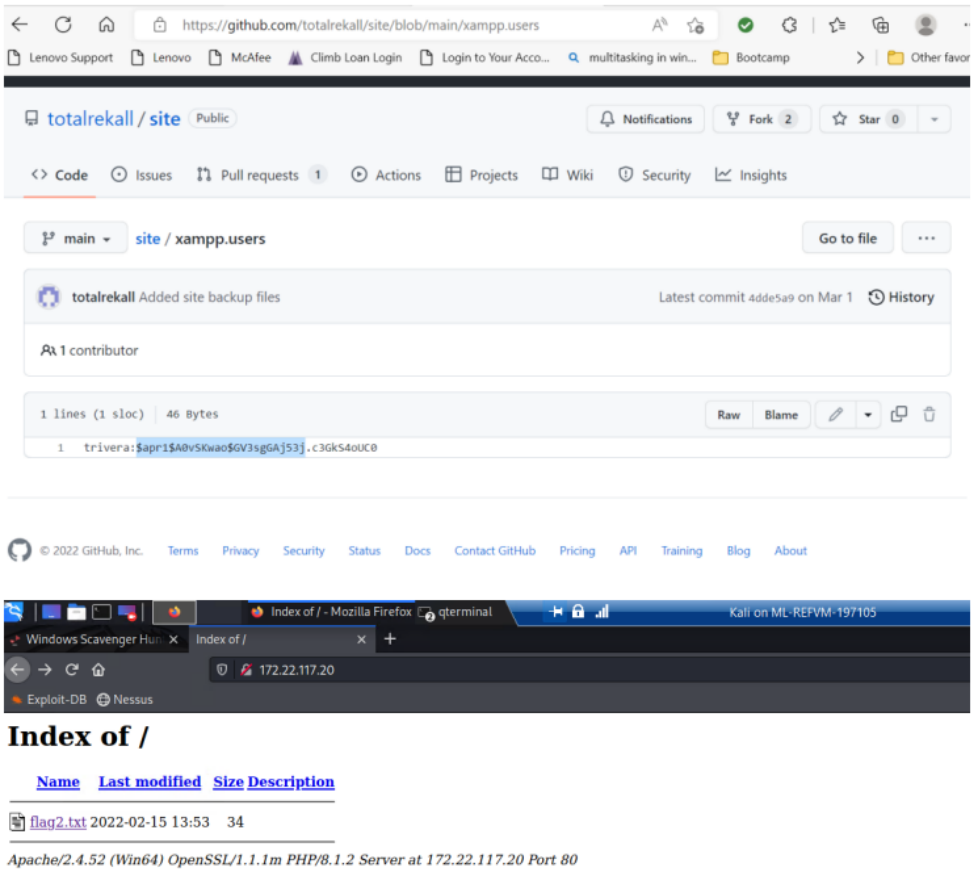
Title	PHP Object Injection
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	**Medium**
Description	From the souvenirs.php page I was able to exploit the search to add other parameters. The PHP Injection I used was "?message="";system('cat/etc/passwd') to access the /etc/passwd file.
Images	
Affected Hosts	34.102.136.180 – totalrekall.xyz
Remediation	Input validation and sanitization Avoid using vulnerable functions Secure dynamic evaluation Error handling

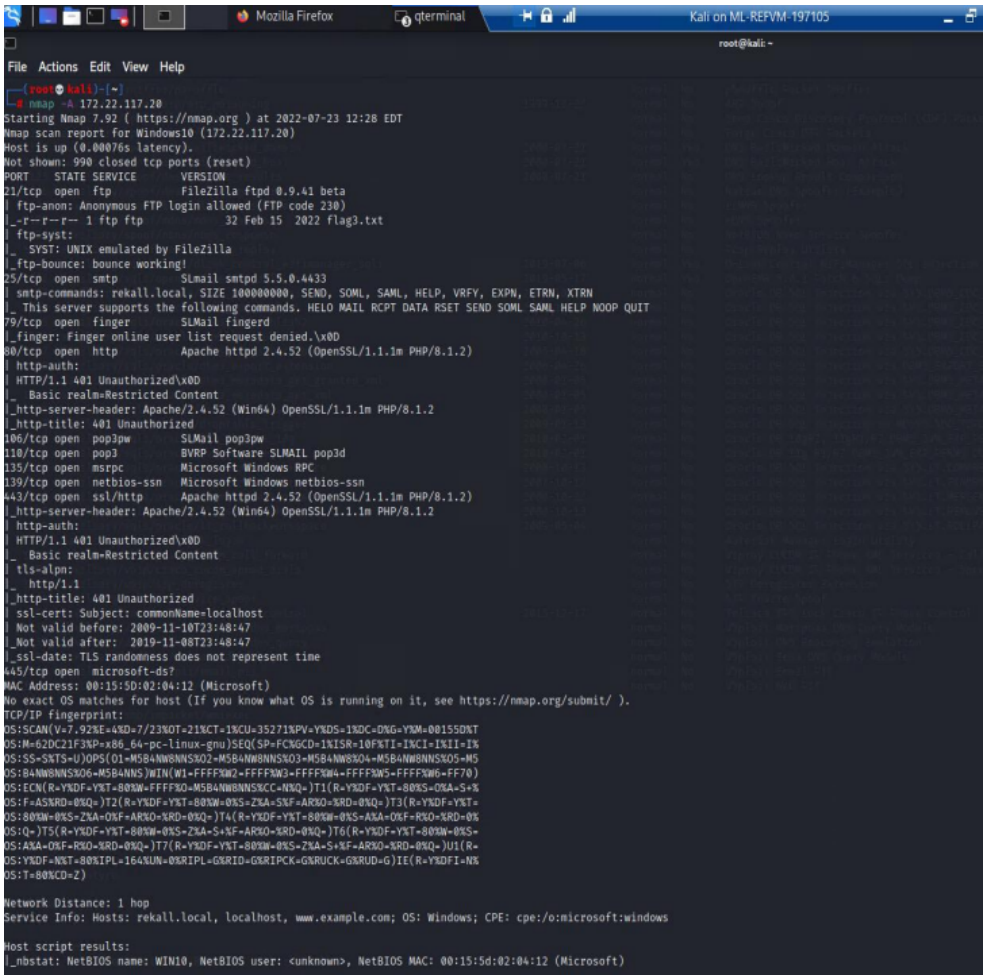
Title	Apache Struts 2.3.5-2.3.31/2.5x<2.5.10.1 Jakarta Multipart parser RCE (remote)
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	**Critical**
Description	I ran Zenmap scans on 192.168.13.0/24 to find the count of hosts (found 5 hosts). I then ran an aggressive scan on the discovered hosts. The host running Drupal was 192.168.13.13. I then used the module "mutli/http/apache_mod_cgi_bash_env_exec" and manipulated the payload to

	access the host by gaining shell through 192.168.13.10 again using RCE exploit through metasploit. Once inside I was then able to access the /etc/sudoers file as well as the /etc/passwd file.
Images	 

	 
Affected Hosts	192.168.13.10 - Linux
Remediation	Upgrade or update to the latest version of Apache Struts.

Title	Exposed data on totalrekall Public GitHub Repository
Type (Web app / Linux OS / WIndows OS)	Windows OS

Risk Rating	High
Description	I discovered that the public GitHub Repository for totalrekall contained a exposed hash for a user. I then cracked the hash. Then I performed Zenmap scans of the IP range for the network and was able to determine the IPs for Windows host, Windows10 machine and Domain Controller. Through this process I also discovered that port 80 (http) was open on the Windows10 machine. I was then able to use the credentials that were found on the GitHub Repository to log in to the Windows10 machine. I performed an aggressive Zenmap scan and found that port 21 (FTP) was open and allowed anonymous access. I was then able to log in to the Windows10 machine through port 21.
Images	

	
Affected Hosts	172.22.117.20 – Windows10 172.22.117.10 – Windows Domain Controller 172.22.117.100 – Windows host
Remediation	Remove sensitive login information on GitHub Repository Do not allow any anonymous access