| Cybersecurity |
| --- |
| Penetration Test Report |

Rekall Corporation

Penetration Test Report

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

# 

## Confidentiality Statement

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

Table of Contents

[Confidentiality Statement 2](#_30j0zll)

[Contact Information 4](#_1fob9te)

[Document History 4](#_3znysh7)

[Introduction 5](#_2et92p0)

[Assessment Objective 5](#_3dy6vkm)

[Penetration Testing Methodology 6](#_2s8eyo1)

[Reconnaissance 6](#_17dp8vu)

[Identification of Vulnerabilities and Services 6](#_3rdcrjn)

[Vulnerability Exploitation 6](#_26in1rg)

[Reporting 6](#_lnxbz9)

[Scope 7](#_35nkun2)

[Executive Summary of Findings 8](#_44sinio)

[Grading Methodology 8](#_z337ya)

[Summary of Strengths 9](#_3j2qqm3)

[Summary of Weaknesses 9](#_1y810tw)

[Executive Summary Narrative](#_4i7ojhp) 10

[Summary Vulnerability Overview 1](#_2xcytpi)3

Vulnerability Findings [1](#_1ci93xb)4

# 

## Contact Information

| **Company Name** | Clark Enterprises |
| --- | --- |
| **Contact Name** | Adam Clark |
| **Contact Title** | Penetration Tester |

## 

## Document History

| **Version** | **Date** | **Author(s)** | **Comments** |
| --- | --- | --- | --- |
| 001 | 10/26/2022 | Adam Clark |  |

# 

## 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:

Chart

Description automatically generated with medium confidence

### 

### 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.

* C2 (Command and Control) servers appear to be secure.
* Essential ports that are not being used are closed and secure.
* Use of encryption makes it more challenging for attackers to utilize stolen data.

### 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.

* Website is vulnerable to malicious scripts and code, thus possibly exposing confidential data within the site.
* Many bits of sensitive data, particularly located in hidden parts of the website, are found to be vulnerable and easy to obtain.
* Rekall’s servers are vulnerable to enumeration.

## Executive Summary

## 

The findings of a penetration test conducted on Rekall’s network on October 14, 2022 are summarized in this report. A team of skilled security consultants from Clark Enterprises, LLC conducted the test.

A security penetration test is a simulated cyber-attack on a computer system or network. The goal of this test is to identify and exploit vulnerabilities in the system in order to assess the system’s security posture. Penetration tests are an important part of a comprehensive security strategy and can help organizations identify and fix vulnerabilities before they are exploited by attackers.

Key Findings:

Clark Enterprises discovered various flaws that may be exploited by a malicious actor. The most important findings were:

* Vulnerabilities in the web server that could allow an attacker to gain access to sensitive data
* Vulnerabilities in the authentication system that could allow an attacker to gain access to user accounts

These flaws could lead to serious breaches of confidentiality and integrity if they are exploited. Unauthorized access to personal identity information, corporate secrets, and inner workings would be gained by adversaries.

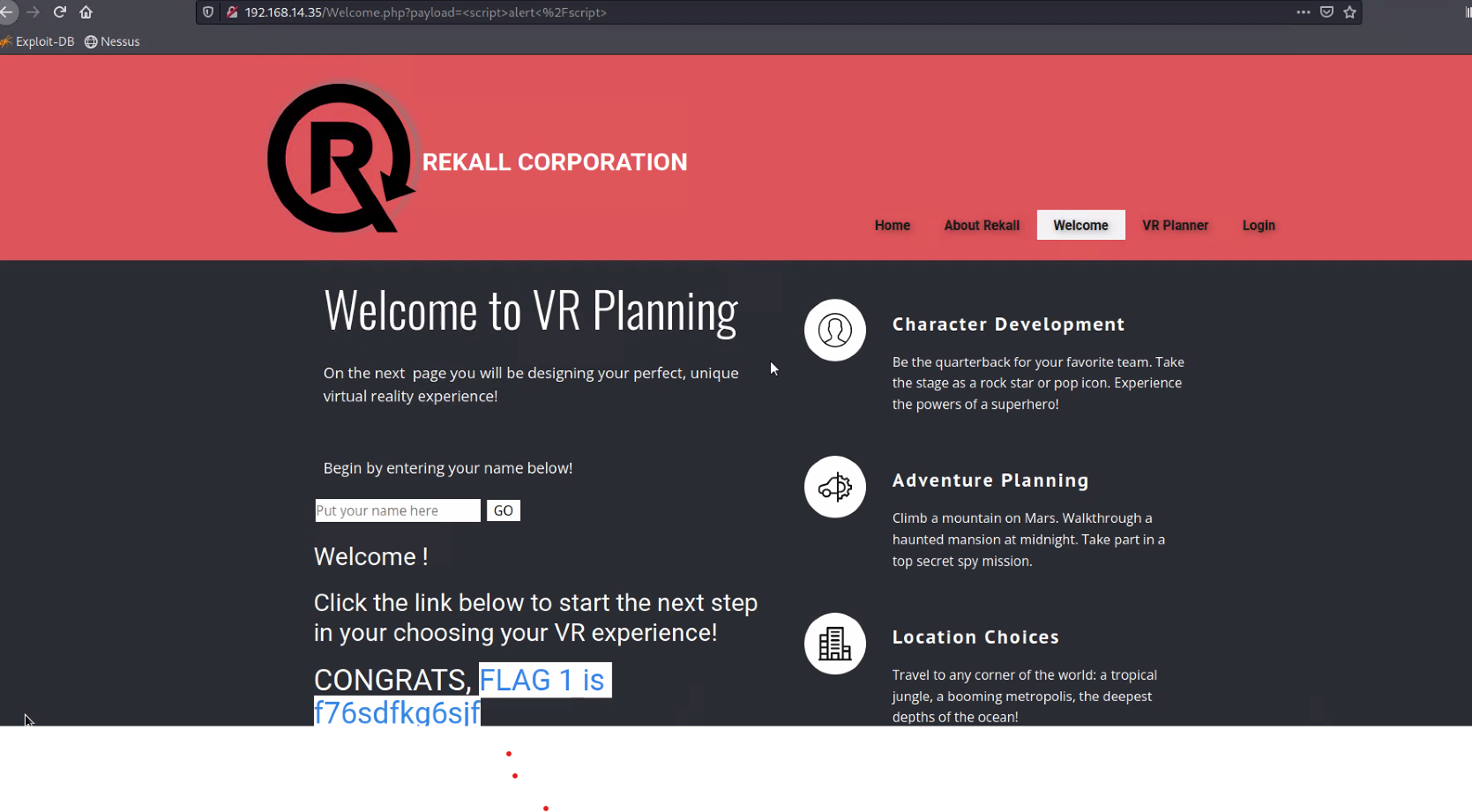
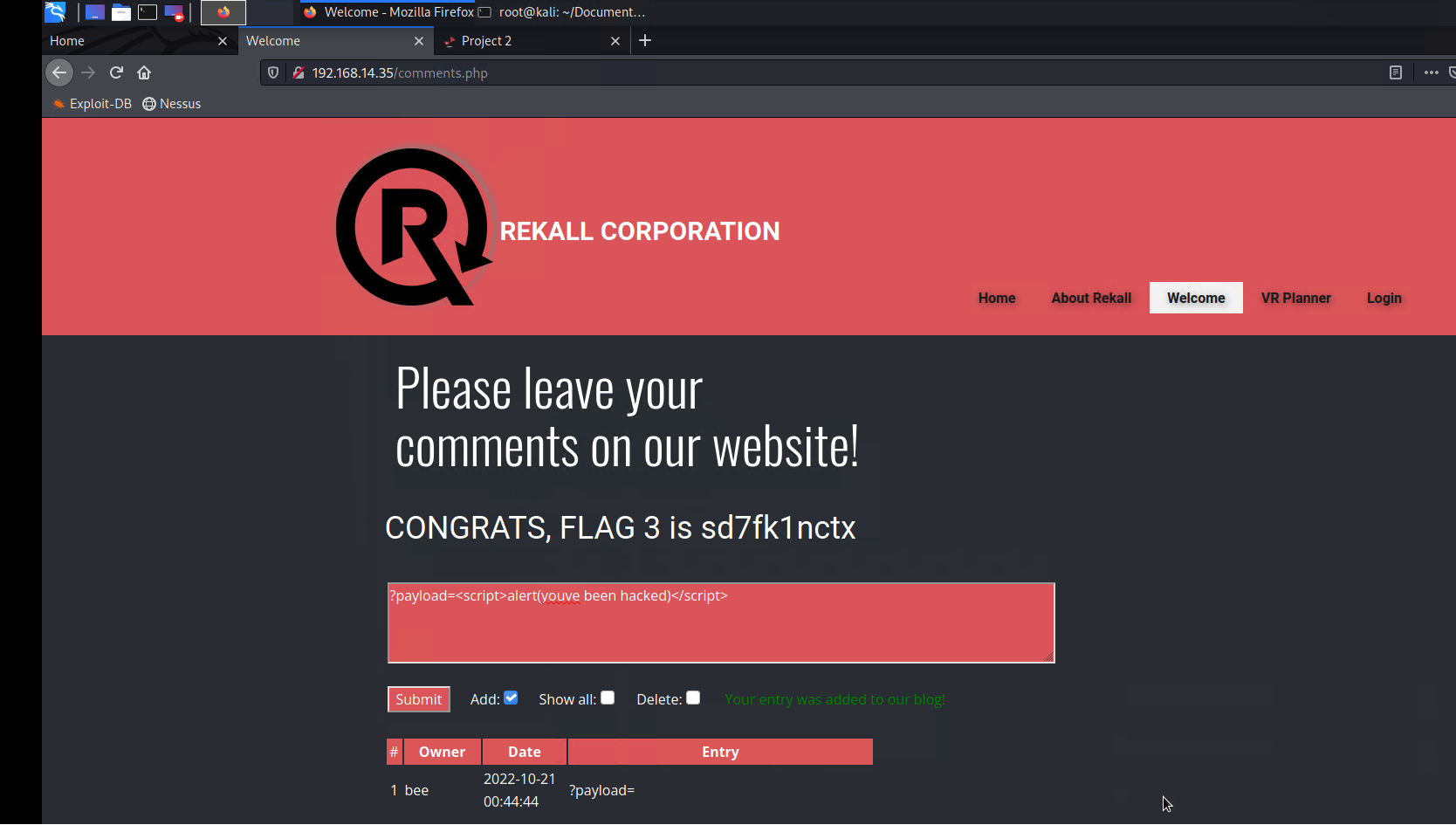
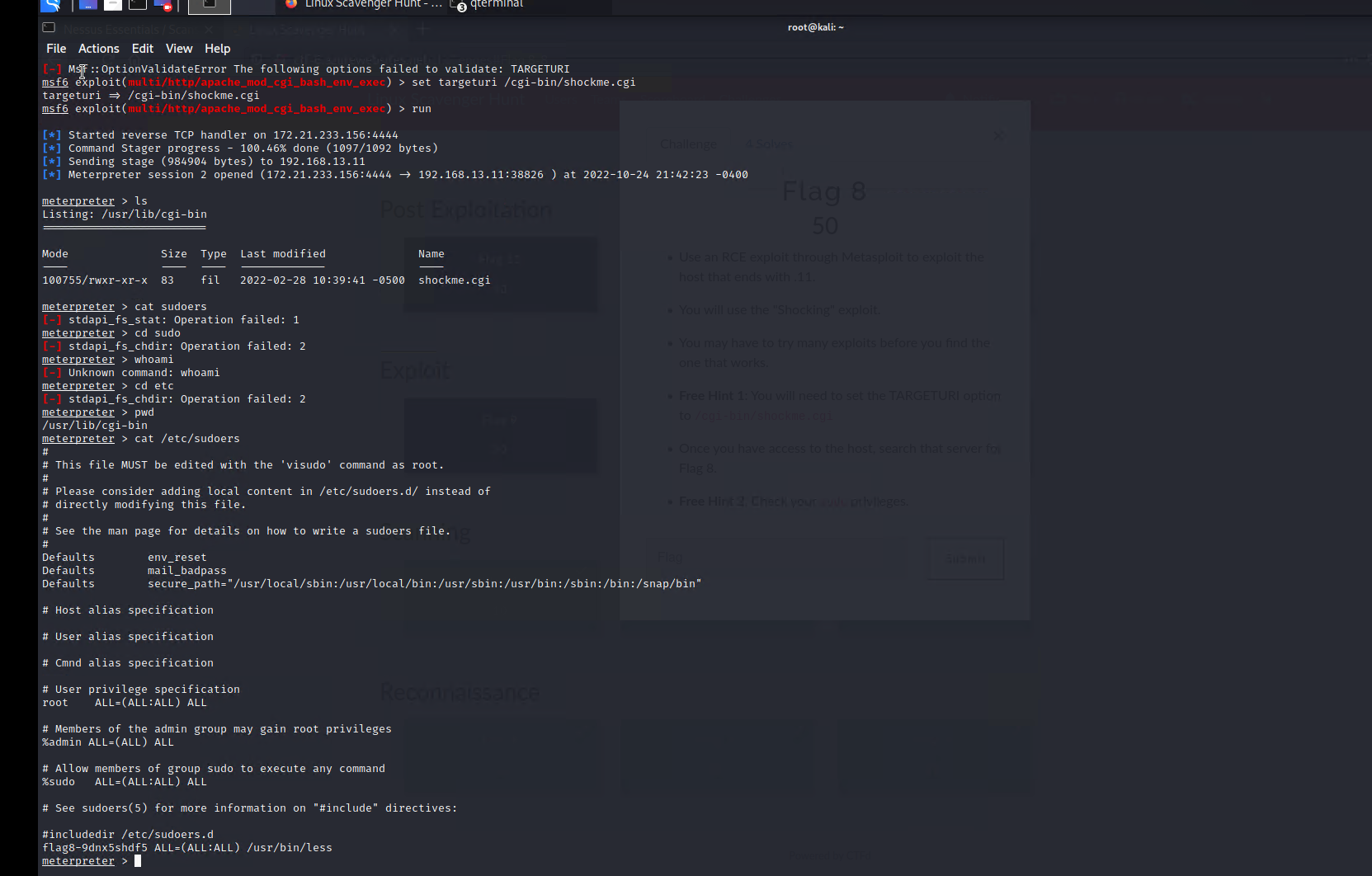
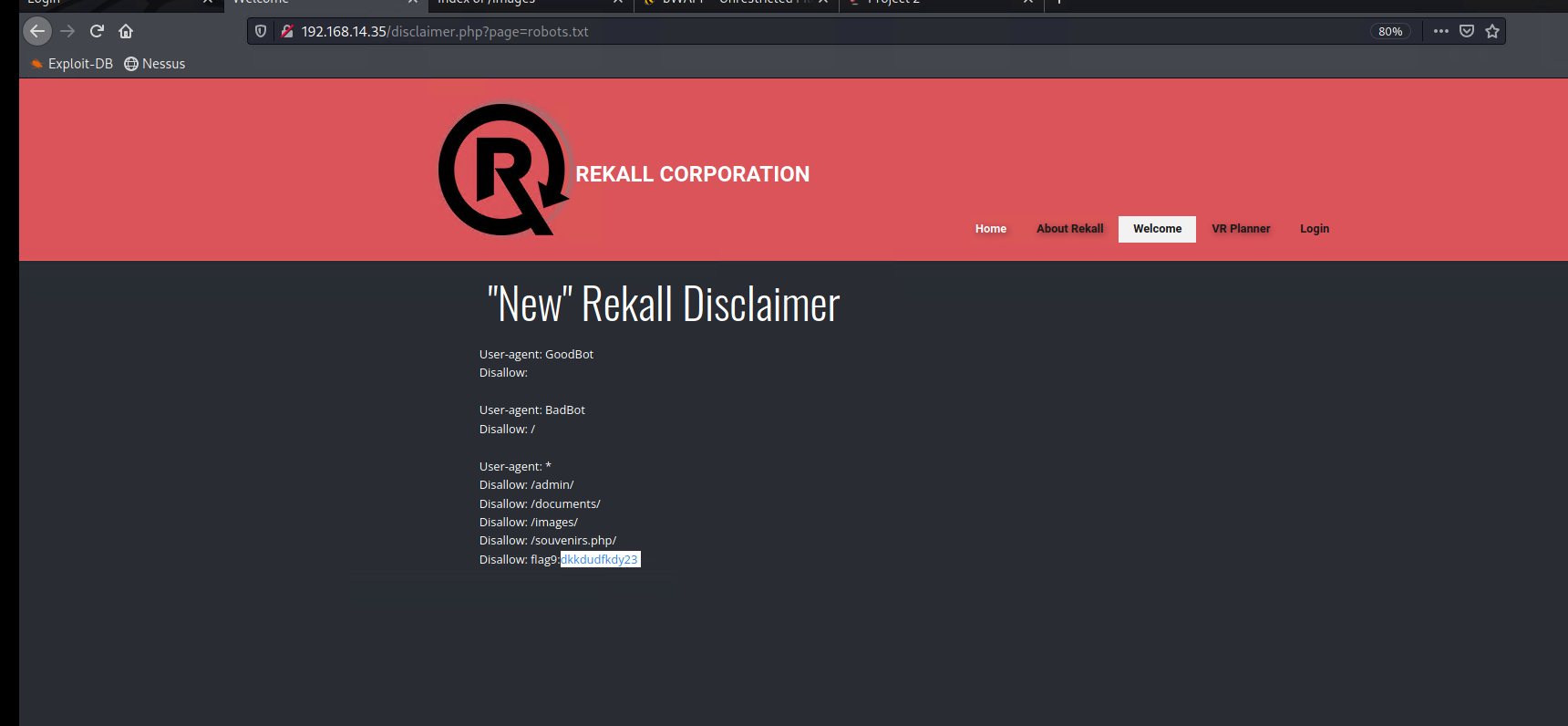
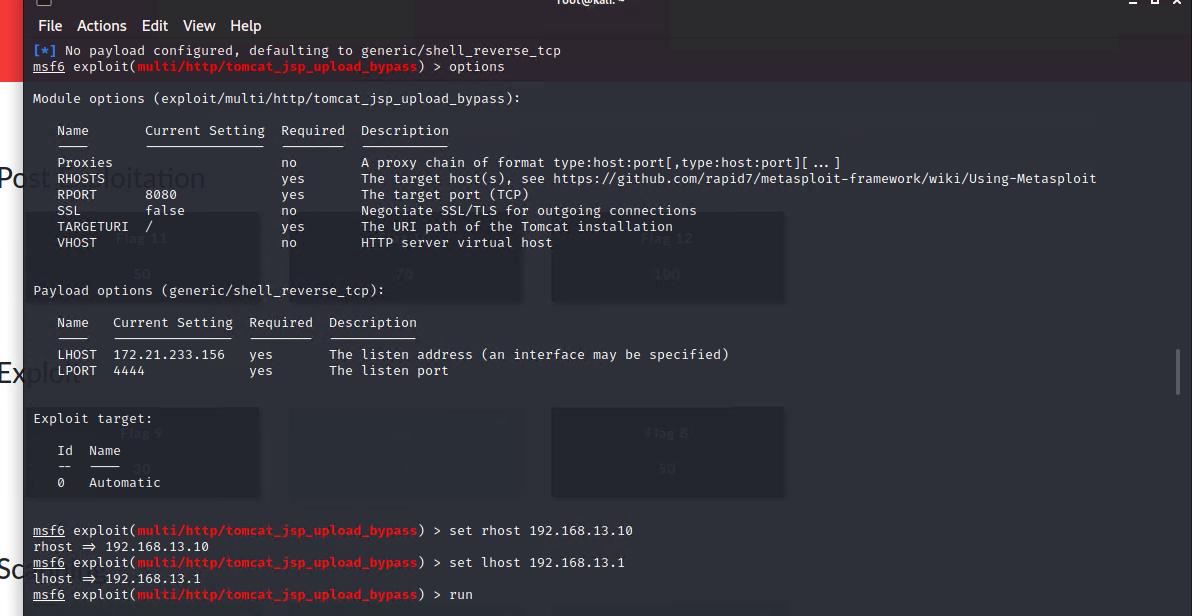
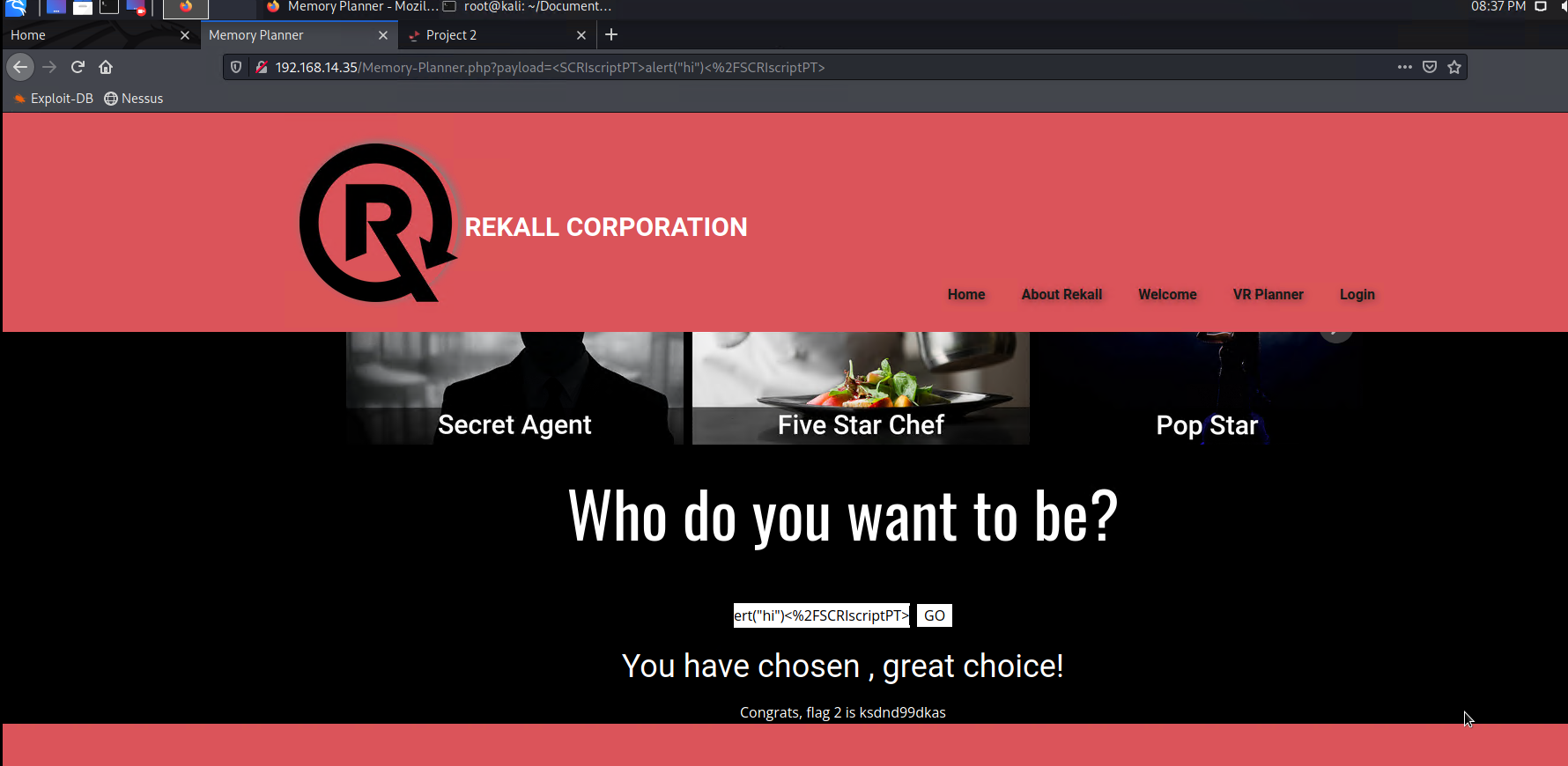
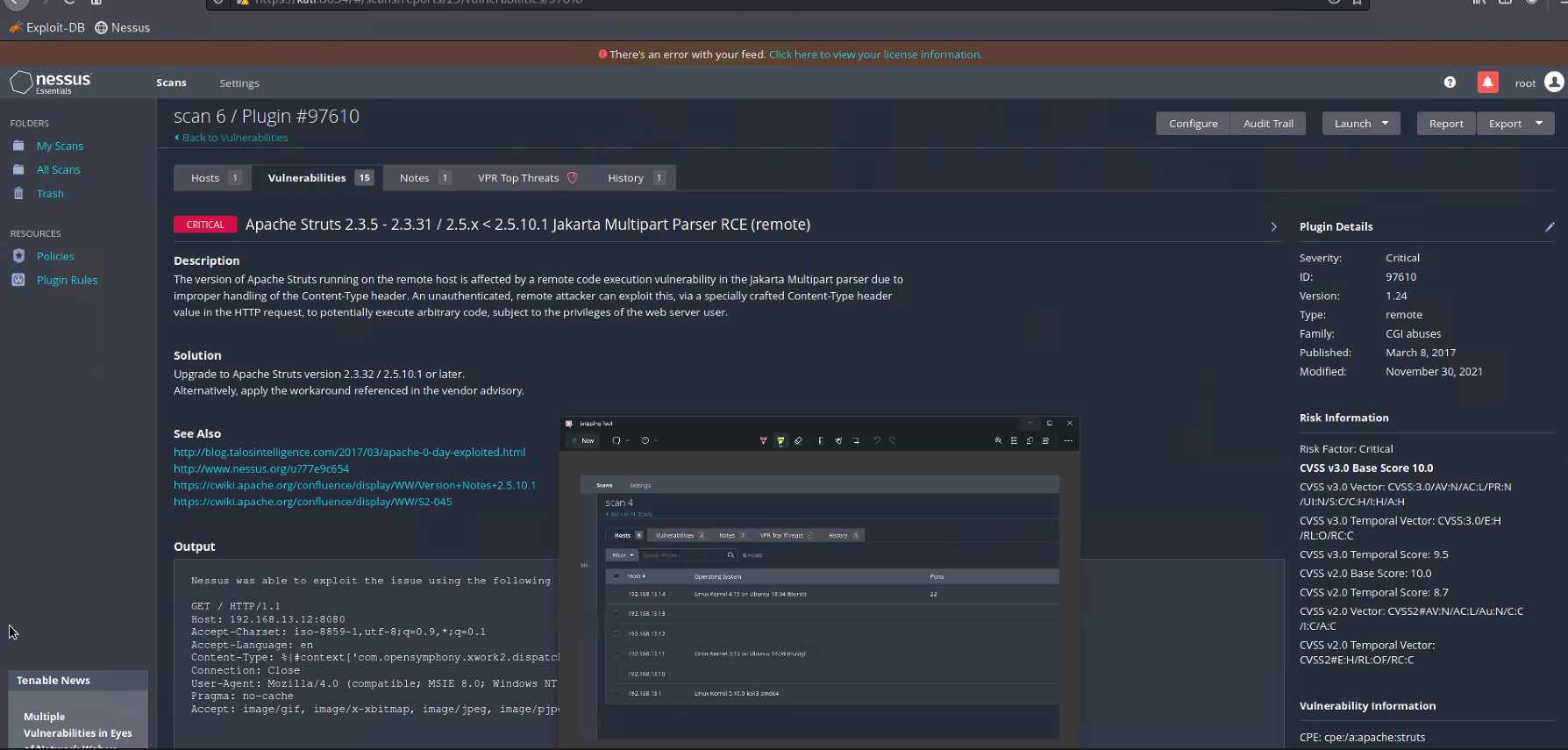
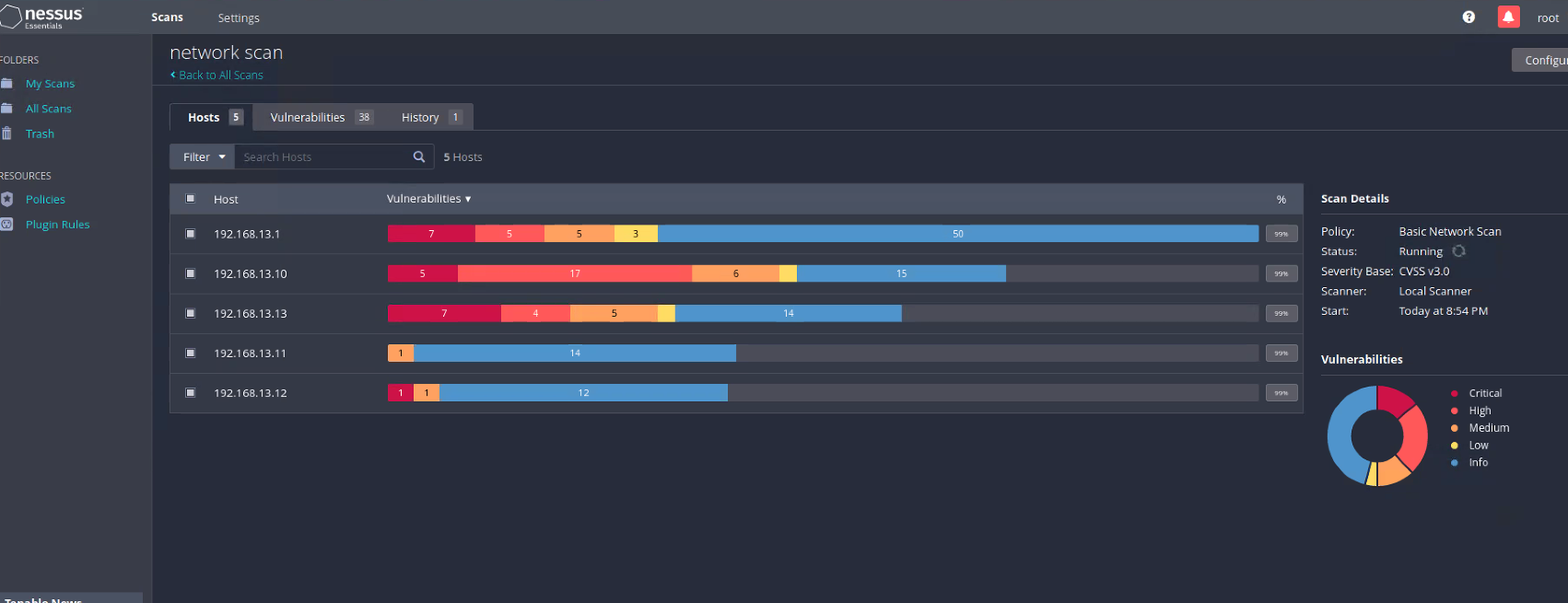
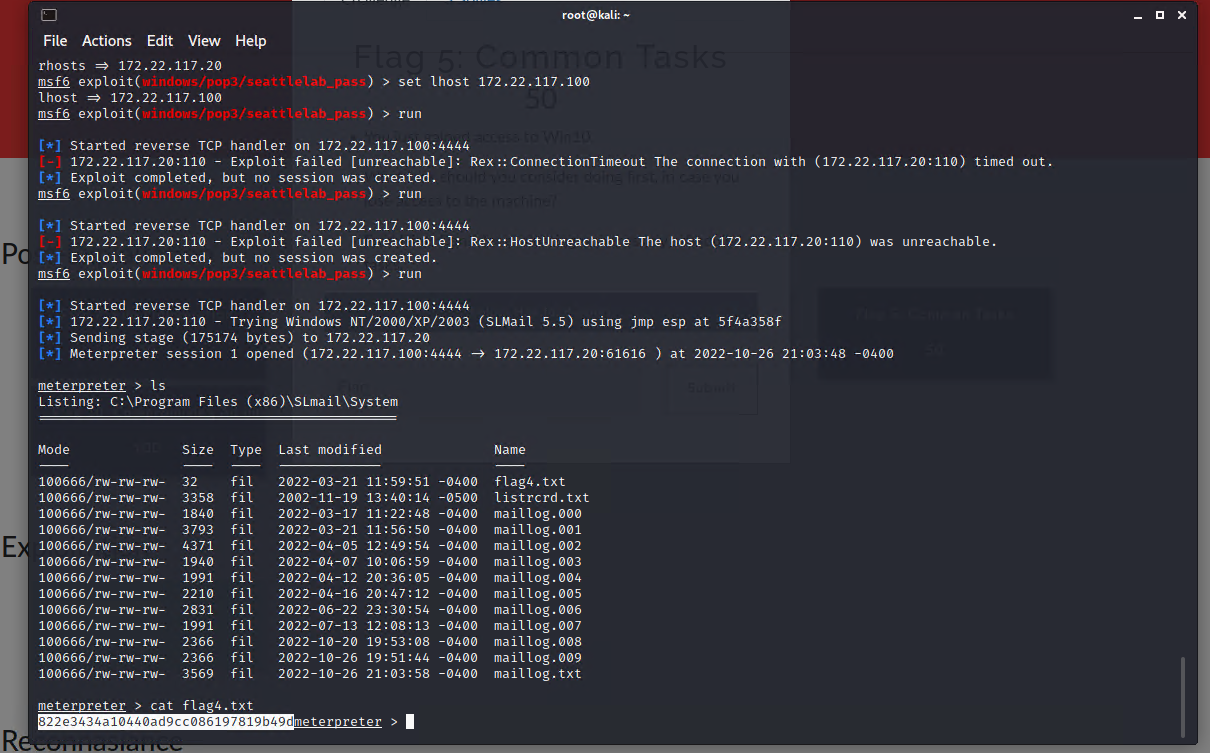
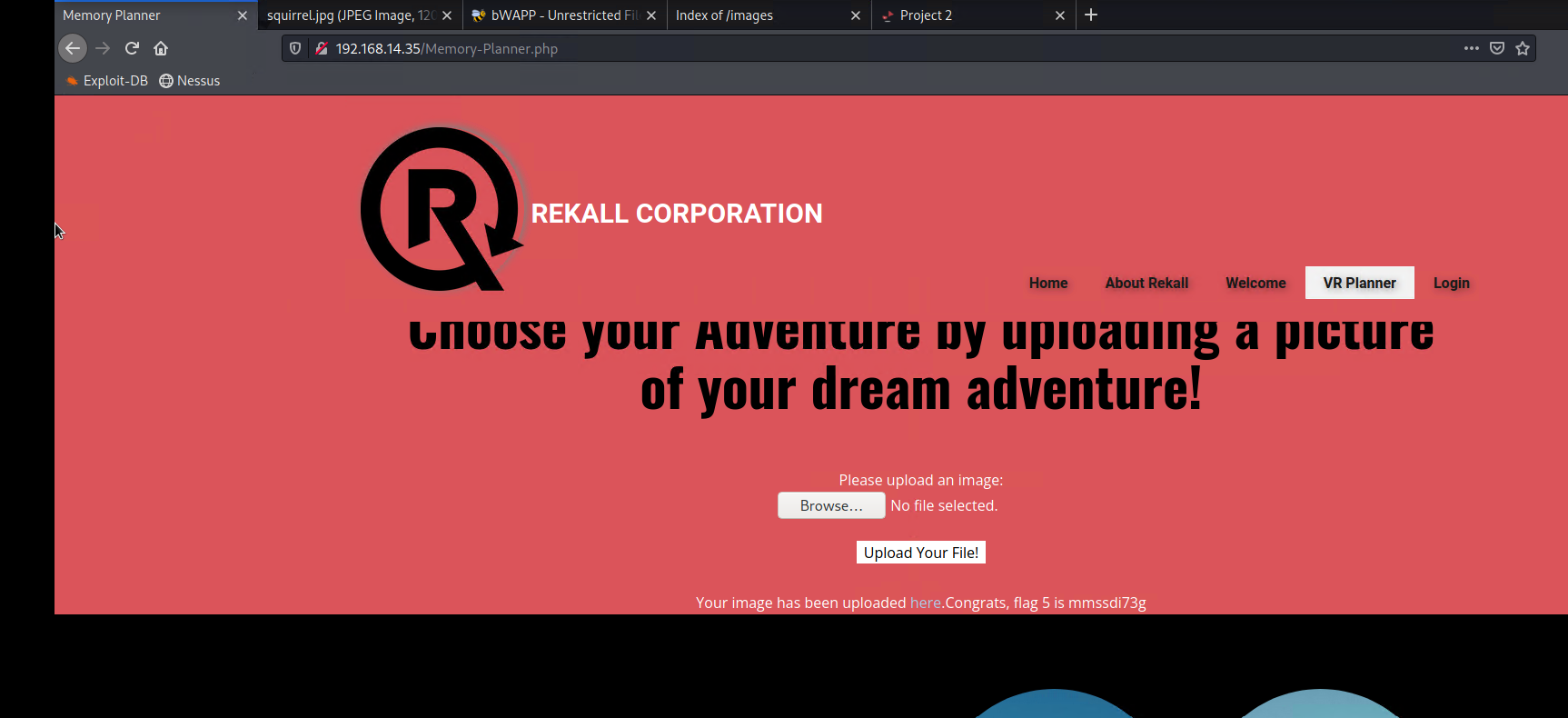
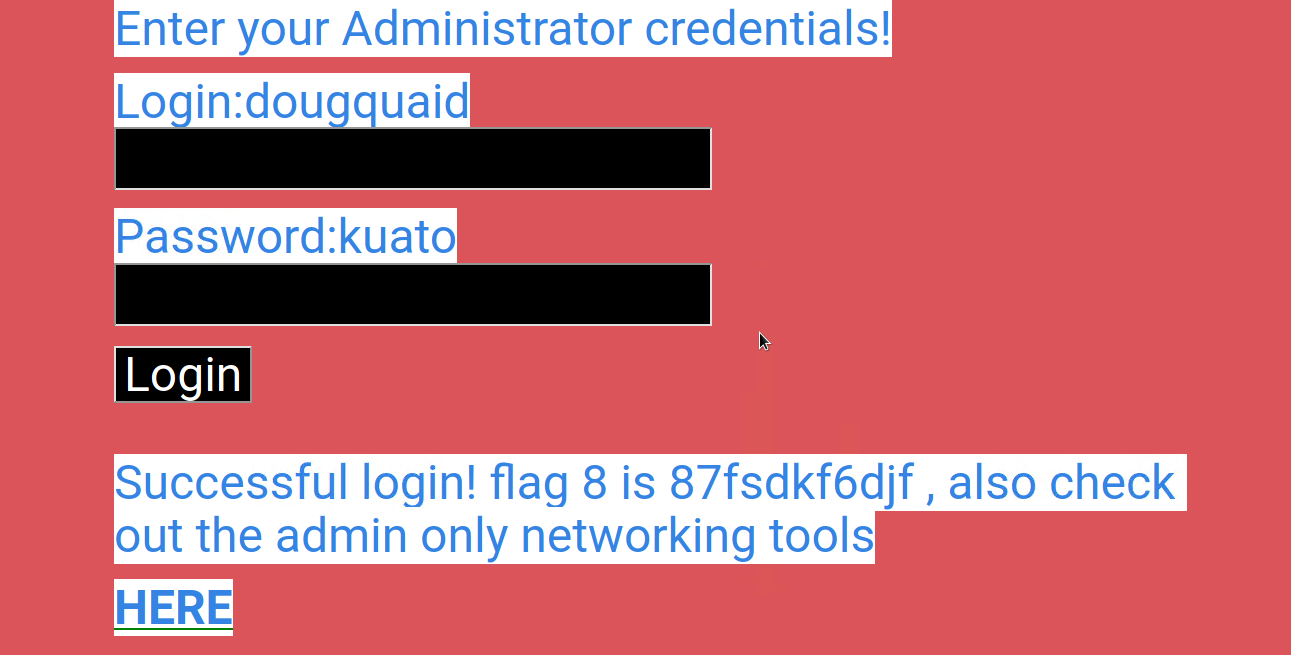
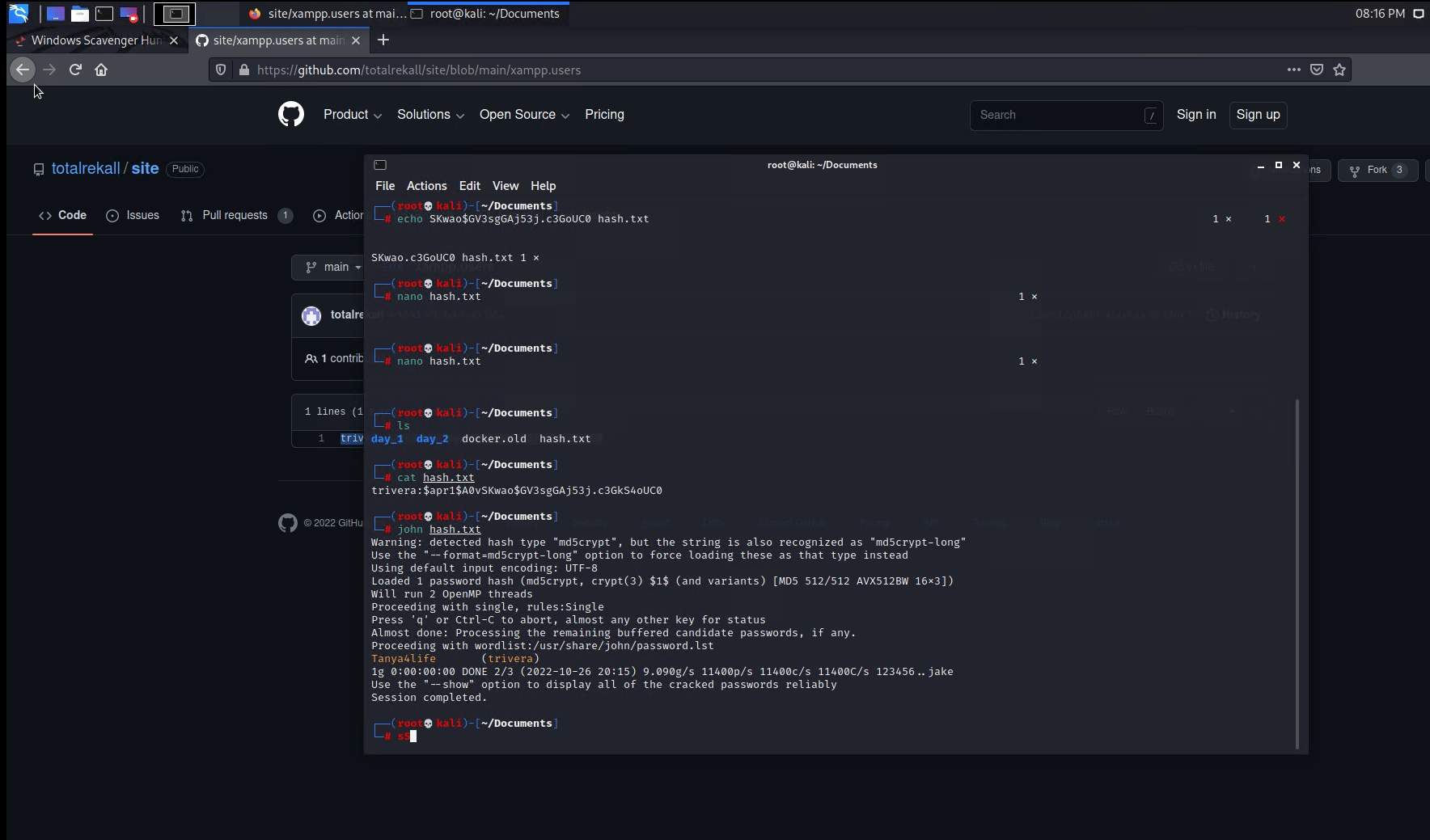
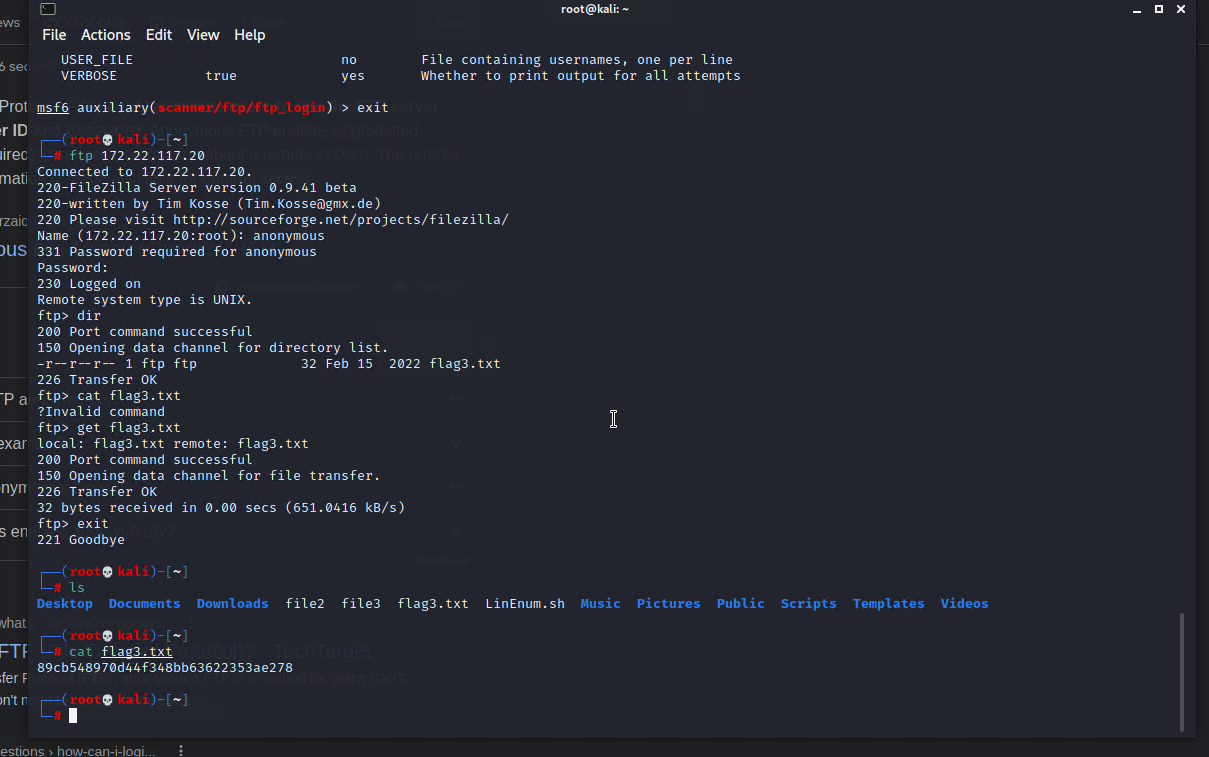
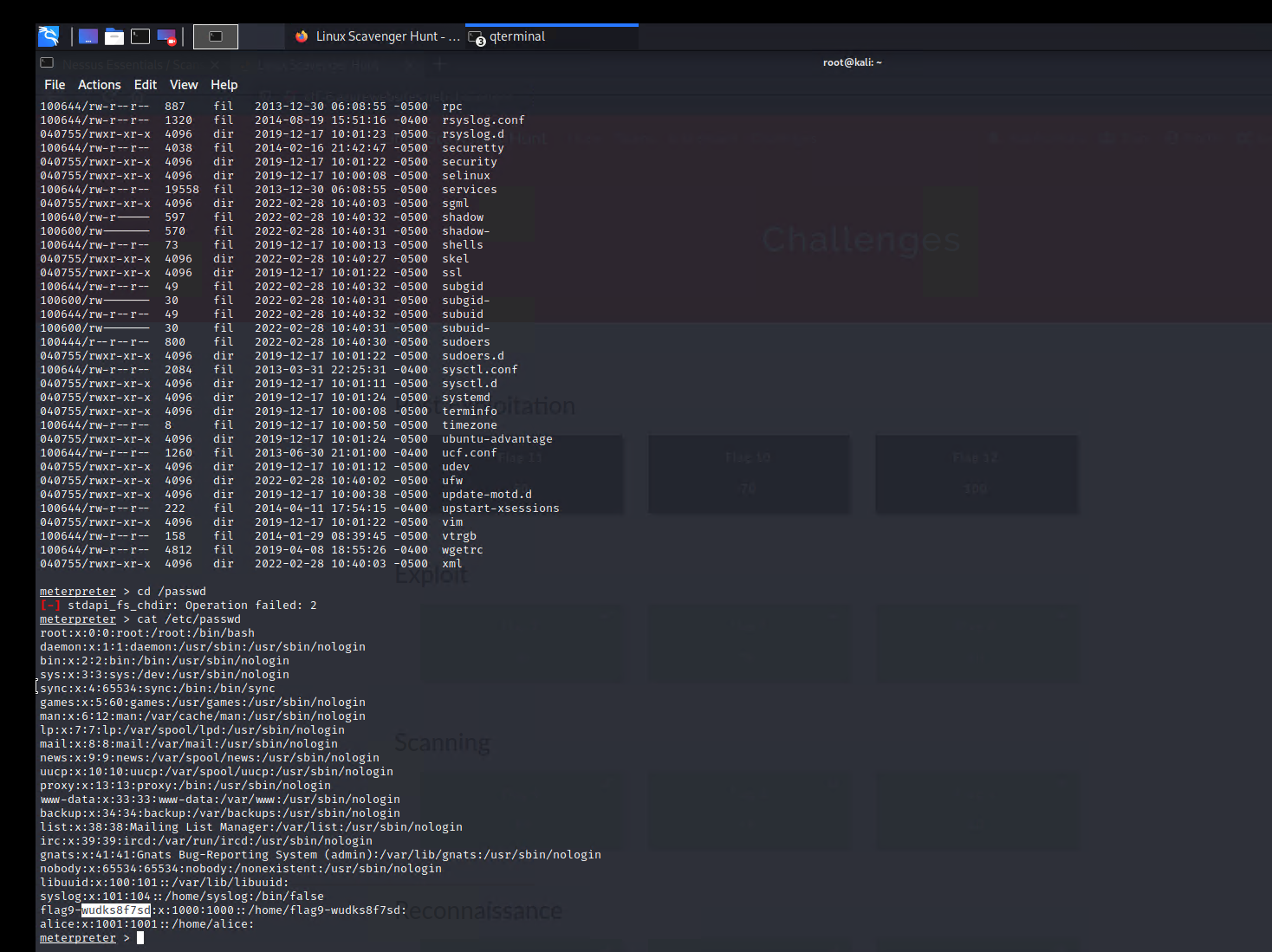
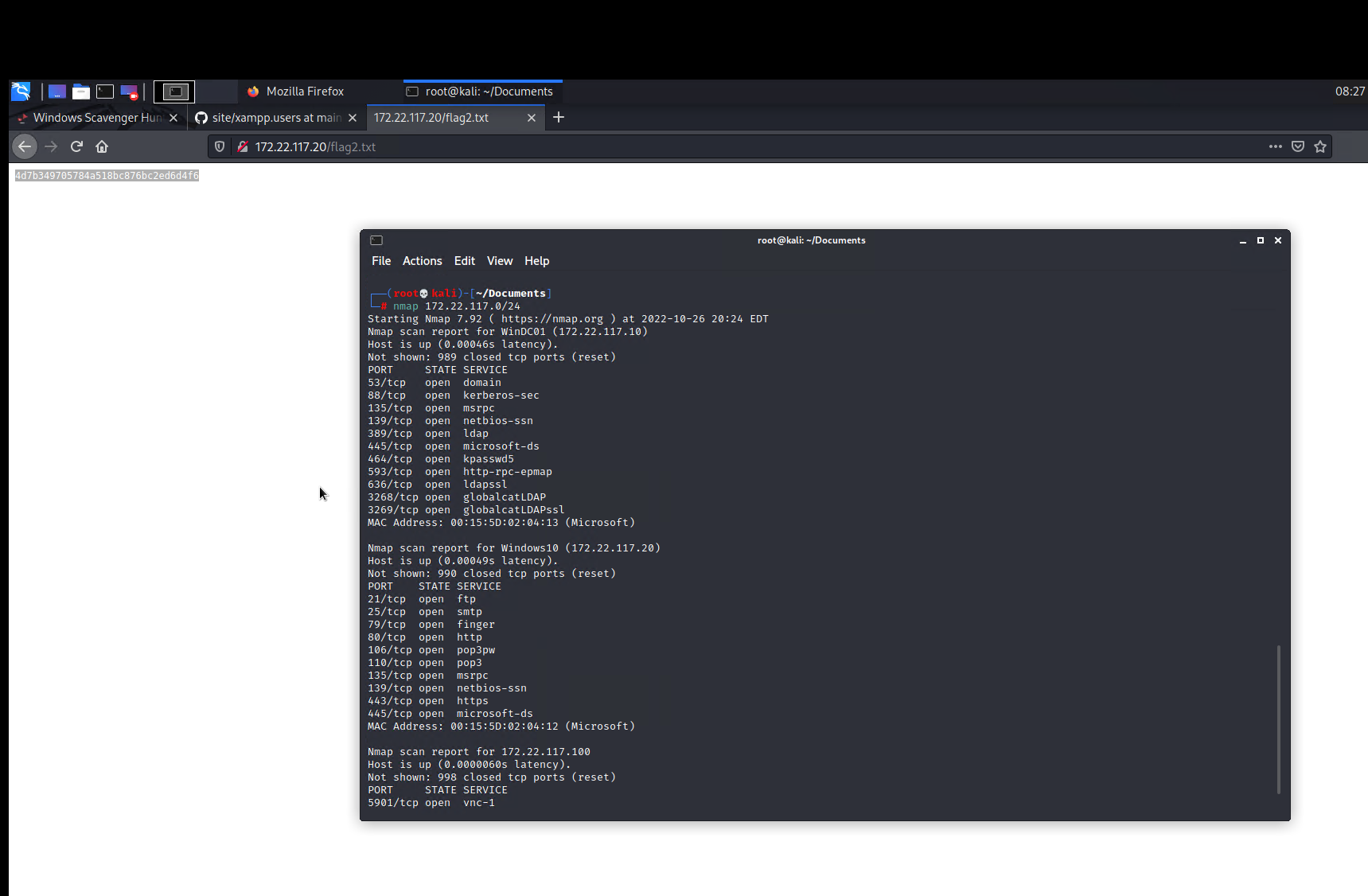
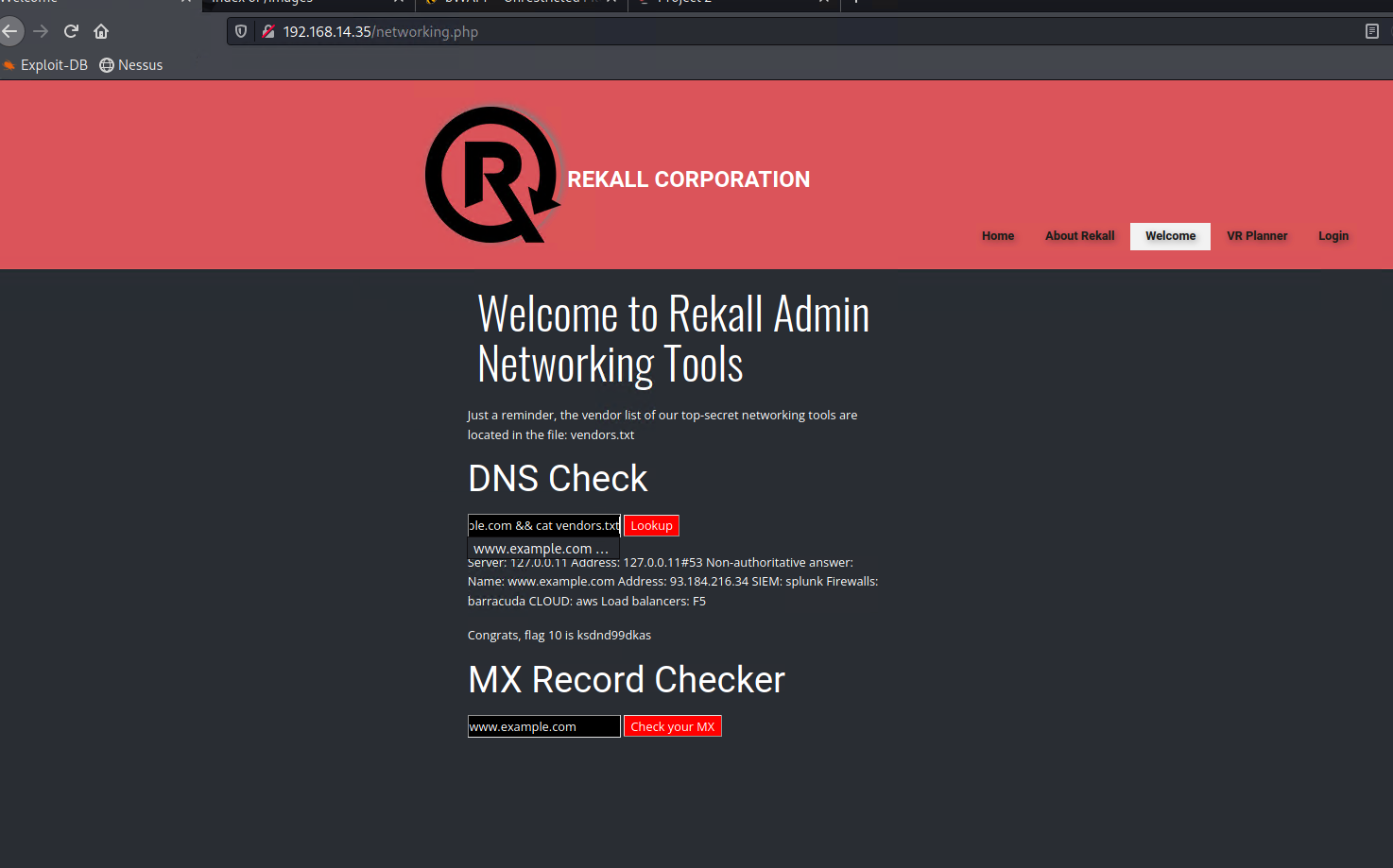
The consequences of a major cyber breach caused by these flaws could include lawsuits and financial losses. In one situation, Clark Enterprises could tamper with the real-time flow of business information. The effects of a real-world attack could put the business’s reputation at risk.

Recommendations:

Clark Enterprises recommends that Rekall:

* Undertakes a security review of all of its firewalls for security misconfiguration issues.
* Performs automated vulnerability scans of all its Internet-facing servers and patches all critical vulnerabilities.
* Request an urgent security patch from the vendor that provides you with single-sign-on (SSO) authentication.
* Perform a review of your email server’s security settings and harden the server in accordance with industry best practices.
* Update all software to the latest versions of themselves to fix bugs, protect against hackers, and improve performance.

Within 15 days, these activities listed above should be completed, and any high or critical risk findings should be resolved within 45 days.



## Summary Vulnerability Overview

| **Vulnerability** | **Severity** |
| --- | --- |
| PHP Injection on website | **Critical** |
| Advanced Cross-Site Scripting (XSS) | **High** |
| Sensitive Data Exposure | **Critical** |
| DNS networking check | **Medium** |
| SSL Certificate research | **Low** |
| Zenmap Scan | **High** |
| ShellShock vulnerability | **Critical** |
| RCE Exploit | **Critical** |
| Local File Inclusion (LFI) | **Critical** |
| Reflected XSS | **High** |
| Stored XSS | **High** |
| SQL Injection | **Critical** |
| Command Injection | **Critical** |
| Nessus Basic Network Scan | **Critical** |
| Open Ports 21, 22, 80, and 443 (Port Enumeration) | **Critical** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

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

| **Scan Type** | **Total** |
| --- | --- |
| Hosts | 3 |
| Ports | 4 |

| **Exploitation Risk** | **Total** |
| --- | --- |
| **Critical** | 9 |
| **High** | 4 |
| **Medium** | 1 |
| **Low** | 1 |

## Vulnerability Findings

| **Vulnerability 1** | **Findings** |
| --- | --- |
| **Title** | Reflected Cross-Site Scripting |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | High |
| **Description** | Occurs when an application receives data in an http request and includes data within an immediate response in an unsafe way. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.0/24 |
| **Remediation** | Use a client-side input validation, in which the code might not allow a malicious user to input scripts |

| **Vulnerability 2** | **Findings** |
| --- | --- |
| **Title** | SQL Injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Critical |
| **Description** | A submitted user input can run SQL commands against a database. Depends on the application running queries against a SQL database. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.0/24 |
| **Remediation** | validate user inputs, sanitize data, and raise web application firewalls. |

| **Vulnerability 3** | **Findings** |
| --- | --- |
| **Title** | Local File Inclusion (LFI) |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Critical |
| **Description** | An attacker tricks the application to run unintended back-end code or scripts that are local to the application’s file system. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.0/24 |
| **Remediation** | Avoid passing user-submitted input to the web server; this can be accomplished via input validation. |

| **Vulnerability 4** | **Findings** |
| --- | --- |
| **Title** | Port Enumeration |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | Open ports 21(FTP), 22(SSH), 80 (HTTP) and 443 (HTTPS) are vulnerable to unauthorized users using stolen credentials. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.0/24, 172.22.117.20 |
| **Remediation** | Close any and all ports that are not in use and secure critical ports 22 and 443. |

| **Vulnerability 5** | **Findings** |
| --- | --- |
| **Title** | RCE Exploit |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | bug that threat actors can exploit by injecting user input into a file or string meant to be executed by the target program’s parser. |
| **Images** |  |
| **Affected Hosts** | 172.22.117.0/24 |
| **Remediation** | Keep OS and 3rd-party software updated, sanitize user input, use Access control lists. |

| **Vulnerability 6** | **Findings** |
| --- | --- |
| **Title** | ShellShock Vulnerability |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | Critical |
| **Description** | vulnerability in the bash shell that lets attackers execute commands and run malicious scripts on OS. |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Use IDS/IPS to detect any type of network communication. |

| **Vulnerability 7** | **Findings** |
| --- | --- |
| **Title** | Command Injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Critical |
| **Description** | Attacker can execute arbitrary command in the host OS and typically injects command by exploiting an application vulnerability. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.10 |
| **Remediation** | Install input validation and implement ‘least privilege’ rule. |