| 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** | NetWork Ninjas |
| --- | --- |
| **Contact Name** | Ali, Mahfoud |
| **Contact Title** | Vulnerability Analyst |

## 

## Document History

| **Version** | **Date** | **Author(s)** | **Comments** |
| --- | --- | --- | --- |
| 001 | 18/04/2024 | Ali, Mahfoud |  |

# 

## 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.

* Protect against denial-of-service attacks to guarantee uninterrupted network accessibility.
* To avoid unwanted access, map network architecture and secure open-source data.
* To improve security, make use of programmes like Nmap, Hashcat, and Metasploit.
* Put both offensive and defensive plans into action for potential threats.
* To find and fix vulnerabilities, carry out continuous penetration testing.

### 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.

* Vulnerable to XSS and SQL injection attacks.
* Inappropriate storage of credentials in HTML source code.
* Outdated Apache web server exposed to multiple exploits.
* Susceptible to SLMail server exploits allowing unauthorized shell access.
* Unauthorized access to password hashes facilitates password cracking and privilege escalation.
* Rekall server's physical address is publicly accessible.
* IP addresses within Rekall's range show potential vulnerabilities.
* Open ports create unauthorized access and file enumeration.
* Weak user passwords susceptible to cracking or guessing.
* Company information accessible through OSINT.
* Web vulnerabilities include XSS, sensitive data exposure, local file inclusion, SQL injection, command injection, PHP injection, brute force attacks, session management flaws, and directory traversal.

## Executive Summary

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

## 

## The penetration test revealed critical, high, medium, and low-severity vulnerabilities across various components, including web applications, Linux, and Windows operating systems.

## 

## Critical vulnerabilities included Apache Struts CVE 2017 5638, PHP injection, session management flaws, and privilege escalation issues. Exploitation risks were identified, such as web server Shellshock, Admin Server Passwords Exposed by Kiwi, and the execution of system shells using found admin server passwords. These vulnerabilities show serious threats to the security and integrity of the systems.

## 

## High-risk vulnerabilities included SQL injection, directory traversal, XSS, local file inclusion, and open-source data exposure. These vulnerabilities could lead to unauthorized access, data breaches, and compromise of sensitive information.

## 

## Medium-risk findings included the exposure of IP addresses, DNS lookup results, and open FTP ports. While not as severe as critical or high-risk vulnerabilities, they still require attention to prevent potential exploitation and data exposure.

## 

## Low-risk vulnerabilities, such as the HTTP response headers containing sensitive information, while less severe, should still be addressed to enhance overall security posture and prevent potential leakage of sensitive data.

## 

## The findings highlight the importance of implementing effective remediation strategies, including input validation, access controls, patch management, and security awareness training. Regular security audits and monitoring are essential to detect and mitigate vulnerabilities promptly, reducing the risk of exploitation and minimizing potential impact on organizational security.

## 

## 

## 

## Summary Vulnerability Overview

| **Vulnerability** | **Severity** |
| --- | --- |
| Apache Struts CVE 2017 5638 | **CRITICAL** |
| PHP injection | **CRITICAL** |
| Session management | **CRITICAL** |
| Privilege Escalation | **CRITICAL** |
| Metasploit is used to gain access to SLMail Port 110 | **CRITICAL** |
| Web Server Shellshock (Port 80) | **CRITICAL** |
| Admin Server Passwords Exposed by Kiwi | **CRITICAL** |
| Executed System Shell using Found Admin Server Passwords | **CRITICAL** |
| Metasploit is used to gain access to SLMail Port 110 | **CRITICAL** |
| Apache Tomcat Vulnerability Enabling Remote Code Execution | **CRITICAL** |
| Exposure of Sensitive Data to SQL Injection | **HIGH** |
| Directory traversal | **HIGH** |
| Open source exposed data | **HIGH** |
| Local file inclusion | **HIGH** |
| Apache Tomcat Vulnerability Enabling Remote Code Execution | **HIGH** |
| XSS Reflected | **HIGH** |
| XSS Stored | **HIGH** |
| Apache Tomcat Vulnerability Enabling Remote Code Execution | **HIGH** |
| Drupal | **HIGH** |
| User Credentials exposed on website | **HIGH** |
| Command Injection | **HIGH** |
| Important information stored in Public/Documents directory | **HIGH** |
| Dig TXT totalrekall.xyz | **MEDIUM** |
| crt.sh | **MEDIUM** |
| Port 21 open | **MEDIUM** |
| Running Nmap exposed IPS | **MEDIUM** |
| curl -v http://192.168.14.35/About-Rekall.ph | **LOW** |

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

| **Scan Type** | **Total** |
| --- | --- |
| Hosts | 192.168.13.12  192.168.13.13  192.168.13.11  192.168.13.10  172.22.117.20  192.168.13.14  192.168.14.35  172.22.117.10 |
| Ports | 22  21  80  106  110 |

| **Exploitation Risk** | **Total** |
| --- | --- |
| **Critical** | 10 |
| **High** | 12 |
| **Medium** | 4 |
| **Low** | 1 |

## Vulnerability Findings

| **Vulnerability 1** | **Findings** |
| --- | --- |
| **Title** | XSS reflected |
| **Type (Web app / Linux OS / WIndows OS)** | Webapp |
| **Risk Rating** | HIGH |
| **Description** | On welcome page in the text box where it asks for you name I put in this script <script>alert('XSS');</script> |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Effective input validation prevents malicious content from entering the system, reducing the threat of XSS attacks. |

| **Vulnerability 2** | **Findings** |
| --- | --- |
| **Title** | XSS reflected Advanced |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | HIGH |
| **Description** | this exploit was executed on the memory planner page in the text field where it says choose your character the script was <SCRIPscriptT>alert("hi")</SCRIPscripTt> |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | input validation |

| **Vulnerability 3** | **Findings** |
| --- | --- |
| **Title** | XSS Stored |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | High |
| **Description** | this exploit was done on the comments page and the script was reflected with this script <script>alert(“Funtimes”)</script> |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | input validation  content security policy (CSP) |

| **Vulnerability 4** | **Findings** |
| --- | --- |
| **Title** | curl -v http://192.168.14.35/About-Rekall.ph |
| **Type (Web app / Linux OS / WIndows OS)** | Linux os |
| **Risk Rating** | LOW |
| **Description** | The HTTP response headers contain the flag. BURP or a cURL request can be used to view these headers |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Access Controls: Ensure only authorized users can access sensitive information on the About page by implementing access controls. |

| **Vulnerability 5** | **Findings** |
| --- | --- |
| **Title** | Local file Inclusion |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | HIGH |
| **Description** | uploaded php files on the VR Planner page |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Prevent attackers from manipulating file paths by avoiding direct appending. Instead, retrieve files using safer methods like database lookups or predefined mappings. |

| **Vulnerability 6** | **Findings** |
| --- | --- |
| **Title** | Local file Inclusion |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | HIGH |
| **Description** | uploaded php files on the VR Planner page |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Avoid Direct File Path Appending |

| **Vulnerability 7** | **Findings** |
| --- | --- |
| **Title** | SQL injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | HIGH |
| **Description** | on the login page where it prompts me for a username and password I instead entered ok' or 1=1-- |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Input validation |

| **Vulnerability 8** | **Findings** |
| --- | --- |
| **Title** | SQL injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | HIGH |
| **Description** | found credentials when highlighted |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Remove or mask sensitive data displayed on the webpage, especially when it's unnecessary for the user to see the full details. This can help prevent unauthorized access even if the data is highlighted. |

| **Vulnerability 9** | **Findings** |
| --- | --- |
| **Title** | Sensitive data exposure |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | HIGH |
| **Description** | Just typed robots.txt in the URL |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Implement access control |

| **Vulnerability 10** | **Findings** |
| --- | --- |
| **Title** | Command injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | HIGH |
| **Description** | typing [www.example.com](http://www.example.com) exposed the flag |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Implement input validation |

| **Vulnerability 11** | **Findings** |
| --- | --- |
| **Title** | Command injection |
| **Type (Web app / Linux OS / WIndows OS)** | web app |
| **Risk Rating** | HIGH |
| **Description** | typing [www.example.com](http://www.example.com) exposed the flag |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Implement input validation |

| **Vulnerability 12** | **Findings** |
| --- | --- |
| **Title** | Brute force attack |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | Critical |
| **Description** | found passwords than used them more specifically melina for username and password |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Multi-Factor Authentication (MFA): Implement multi-factor authentication (MFA) to add an extra layer of security on top of passwords. MFA requires users to provide additional verification factors, such as a one-time password sent to their mobile device phone |

| **Vulnerability 13** | **Findings** |
| --- | --- |
| **Title** | PHP injection |
| **Type (Web app / Linux OS / WIndows OS)** | web app |
| **Risk Rating** | CRITICAL |
| **Description** | altered the URL to http://192.168.13.35/souvenirs.php?message=""; system('cat /etc/passwd') which gave me the flag |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Web Application Firewall (WAF): Use a Web Application Firewall (WAF) to filter and block malicious requests that may attempt PHP injection attacks. make the WAF detect and block known PHP injection attacks |

| **Vulnerability 14** | **Findings** |
| --- | --- |
| **Title** | Session management |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | CRITICAL |
| **Description** | I used this URLl <http://192.168.13.35/admin_legal_data.php?admin=87> and just guessed the session ID |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Use Secure Session IDs: Generate session IDs securely using cryptographically strong random number generators. Ensure that session IDs are sufficiently long and unpredictable to resist guessing attacks. |

| **Vulnerability 15** | **Findings** |
| --- | --- |
| **Title** | Directory traversal |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | HIGH |
| **Description** | <http://192.168.13.35/disclaimer.php?page=old_disclaimers/disclaimer_1.txt> I ran this url and it gave me the output |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | filter and validate user inputs make sure that the file system is using a sanitizer for the users |

| **Vulnerability 16** | **Findings** |
| --- | --- |
| **Title** | Open source exposed data |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | HIGH |
| **Description** | <https://centralops.net/co/DomainDossier.aspx> went to this website and when prompted for domain and ip address I used totalrekall.xyz viewed the whois records and found the flag there |
| **Images** |  |
| **Affected Hosts** |  |
| **Remediation** | Register domains privately to avoid exposing sensitive information |

| **Vulnerability 17** | **Findings** |
| --- | --- |
| **Title** | dig TXT totalrekall.xyz |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | MEDIUM |
| **Description** | ran dig TXT totalrekall.xyz and gave me the flag |
| **Images** |  |
| **Affected Hosts** | 172.31.176.1 |
| **Remediation** | Use a DNS lookup tool to check the TXT records for totalrekall.xyz. Look for unusual entries that could signal security risks |

| **Vulnerability 18** | **Findings** |
| --- | --- |
| **Title** | crt.sh |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | MEDIUM |
| **Description** | searched totalrekall.xyz on crt.sh |
| **Images** |  |
| **Affected Hosts** | 34.102.136.180 |
| **Remediation** | don't expose information on crt.sh |

| **Vulnerability 19** | **Findings** |
| --- | --- |
| **Title** | nmap 192.168.13.0/24 |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | MEDIUM |
| **Description** | ran nmap 192.168.13.0/24 |
| **Images** |  |
| **Affected Hosts** | 192.168.13.0/24  192.168.13.10  192.168.13.11  192.168.13.12  192.168.13.13  192.168.13.14 |
| **Remediation** | IP blocking |

| **Vulnerability 20** | **Findings** |
| --- | --- |
| **Title** | nmap -A 192.168.13.0/2 |
| **Type (Web app / Linux OS / WIndows OS)** | LInux OS |
| **Risk Rating** | CRITICAL |
| **Description** | Ran nmap -A 192.168.13.0/2 |
| **Images** |  |
| **Affected Hosts** | 192.168.13.0/2  192.168.13.12 |
| **Remediation** | IP blocking show less information to an aggressive scan |

| **Vulnerability 21** | **Findings** |
| --- | --- |
| **Title** | Nessus scan results |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | Medium |
| **Description** | Ran nessus scan revealed Struts vulnerability |
| **Images** |  |
| **Affected Hosts** | 192.168.13.12 |
| **Remediation** | Make sure Apache is always up to date |

| **Vulnerability 22** | **Findings** |
| --- | --- |
| **Title** | Apache Tomcat Remote Code Execution Vulnerability (CVE-2017-12617) |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | Ran msfconsole to to use exploit multi/http/tomcat\_jsp\_upload\_bypass set RHOSTS to 192.168.13.10 than within the session i ran shell than ran / cat /root/.flag7.txt |
| **Images** |  |
| **Affected Hosts** | 192.168.13.10 |
| **Remediation** | Update apache tomcat  Limit access to the Apache Tomcat server by configuring proper access controls. Use firewalls, network security groups |

| **Vulnerability 23** | **Findings** |
| --- | --- |
| **Title** | Shellsock |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | CRITICAL |
| **Description** | Ran msfconsole than used the exploit exploit/multi/http/apache\_mod\_cgi\_bash\_env\_exec and set the TARGERTURI to /cgi-bin/shockme.cgi and RHOSTS to 192.168.13.11 than ran shell and cat /etc/sudoers to get flag 8 |
| **Images** |  |
| **Affected Hosts** | 192.168.13.11 |
| **Remediation** | Always make sure that the your software is up to date |

| **Vulnerability 24** | **Findings** |
| --- | --- |
| **Title** | Shellsock |
| **Type (Web app / Linux OS / WIndows OS)** | LIinux OS |
| **Risk Rating** | CRITICAL |
| **Description** | stayed on the same shell as flag 8 and ran cat /etc/passwd |
| **Images** |  |
| **Affected Hosts** | 192.168.13.11 |
| **Remediation** | Always make sure that the your software is up to date and having audits every couple of months to make sure sensitive files are not shown publicly |

| **Vulnerability 25** | **Findings** |
| --- | --- |
| **Title** | Struts - CVE-2017-5638 |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | CRITICAL |
| **Description** |  |
| **Images** |  |
| **Affected Hosts** | 192.168.13.12 |
| **Remediation** | Perform updates on Apache and apply least privilege rule where you give the least amount of privilege needed |

| **Vulnerability 27** | **Findings** |
| --- | --- |
| **Title** | Drupal - CVE-2019-6340 |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | HIGH |
| **Description** | Ran msfconsole I searched for Drupal exploits used unix/webapp/drupal\_restws\_unserialize and set RHOSTS to 192.168.13.13 after getting shell I ran getuid |
| **Images** |  |
| **Affected Hosts** | 192.168.13.13 |
| **Remediation** | Deploy a Web Application Firewall (WAF) to filter and block malicious requests that may exploit the CVE-2019-6340 vulnerability. Always make sure the software is up to date |

| **Vulnerability 28** | **Findings** |
| --- | --- |
| **Title** | CVE-2019-14287 ssh |
| **Type (Web app / Linux OS / WIndows OS)** | LinuxOS |
| **Risk Rating** | CRITICAL |
| **Description** | I accesses alice's account by running ssh alice@192.168.13.14 the password was alice and while in her account I ran sudo -u#-1 cat /root/flag12.txt |
| **Images** |  |
| **Affected Hosts** | 192.168.13.14 |
| **Remediation** | use more complex passwords and have a policy so that every user has to change their password every couple of months |

| **Vulnerability 29** | **Findings** |
| --- | --- |
| **Title** | GitHub |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | Medium |
| **Description** | accessed the totatrekall github page by searching for it I than found the hash for user trivera than used john to crack the hash and found the password |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | make sure that credentials are not easily accessible on the internet |

| **Vulnerability 30** | **Findings** |
| --- | --- |
| **Title** |  |
| **Type (Web app / Linux OS / WIndows OS)** |  |
| **Risk Rating** |  |
| **Description** | with the credentials found in the last flag I was able to find the next one searching for <http://172.22.117.20> I was than prompted to put in credentials they were username: trivera password: Tanya4life |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | use more complex passwords and have a policy so that every user has to change their password every couple of months |

| **Vulnerability 31** | **Findings** |
| --- | --- |
| **Title** | ftp 172.22.117.20 |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | CRITICAL |
| **Description** | Running ftp 172.22.117.20 I was then asked for a password I used anonymously for both user and password when logged on. I moved the file flag3.txt to my machine then used the cat flag3.txt |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | use stronger password and dont use easily guessable password or the set the password the same as username |

| **Vulnerability 32** | **Findings** |
| --- | --- |
| **Title** | Seattlelab\_pass |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | HIGH |
| **Description** | Ran msfconcole found the right exploit which was exploit/windows/pop3/seatllelab\_pass set RHOSTS to 172.22.117.20 |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | make sure everything is up to date |

| **Vulnerability 33** | **Findings** |
| --- | --- |
| **Title** | SL\_mail / schtasks |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | HIGH |
| **Description** | searched for schtasks /query /tn flag5 |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | dont allow all users to make tasks also update schtasks |

| **Vulnerability 34** | **Findings** |
| --- | --- |
| **Title** | hashdump |
| **Type (Web app / Linux OS / WIndows OS)** | Window OS |
| **Risk Rating** | HIGH |
| **Description** | when in meterpreter I loaded kiwi within kiwi i ran lsa\_dump\_sam than found the hash for flag 6 than used john to crack it |
| **Images** |  |
| **Affected Hosts** | 172.22.117,20 |
| **Remediation** | Use antivirus/anti-malware, intrusion detection/prevention, and endpoint detection and response tools to protect endpoints. These tools should help detect and block unauthorized access and malicious activities on devices. |

| **Vulnerability 35** | **Findings** |
| --- | --- |
| **Title** | Directory traversal |
| **Type (Web app / Linux OS / WIndows OS)** | WIndows OS |
| **Risk Rating** | HIGH |
| **Description** | using cd to go into Users/Public/Documents than used cat on flag.txt |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Don't put important information in easily accessible files |

| **Vulnerability 36** | **Findings** |
| --- | --- |
| **Title** | kiwi |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | CRITICAL |
| **Description** | used lsadump::cache to retrieve the than used the hash to login into admbobs account to find flag 8 |
| **Images** |  |
| **Affected Hosts** | 172.22.117.100  172.22.117.10 |
| **Remediation** | don't store hash into lsadump::cache implement MFA |

| **Vulnerability 37** | **Findings** |
| --- | --- |
| **Title** |  |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | CRITICAL |
| **Description** | by navigating into the root directory I was able to find lag 9. I then used command cat on it. |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Secure sensitive files by implementing proper access controls. This involves setting permissions and restrictions to limit who can access, modify, or delete these files. |

| **Vulnerability 38** | **Findings** |
| --- | --- |
| **Title** | dcsnc\_ntlm |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | HIGH |
| **Description** | Loaded kiwi in meterpreter than ran dcsync\_ntlm administrator |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | perform regular security audits implement access controls and privilege management |