| 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** | LetUsHackYou |
| --- | --- |
| **Contact Name** | Anthony S. |
| **Contact Title** | Pen Tester |

## 

## Document History

| **Version** | **Date** | **Author(s)** | **Comments** |
| --- | --- | --- | --- |
| 001 | 12/10/2022 | Anthony S. | First Version |

# 

## 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.

* Current and ongoing penetration testing to identify and mitigate vulnerabilities.
* DDOS mitigation strategy in place to prevent network denial.
* Tools such as Metasploit and Nmap used to prevent unauthorized access.
* Forward thinking offensive and defensive strategy.

### 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.

* Web Application is vulnerable to XSS and SQL injection.
* Credentials are visible in open source locations.
* Physical Address is visible on WHOSIS.
* Linux systems are vulnerable to Metasploit exploits resulting in privilege escalation.
* Windows Servers are vulnerable due to Ports 80, 21, and 110 being open resulting in shell access.

## Executive Summary

LetUsHackYou, during Penetration Testing of Rekall’s IT assets, was able to identify several vulnerabilities that could potentially cause catastrophic impact on the reputation or revenue of Rekall. LetUsHackYou was able to infiltrate Rekall’s assets, exfiltrate sensitive data, and escalate privileges within systems shown below:

LetUsHackYou tested Rekall’s Web Application first. We determined that the home page is vulnerable to an XSS Reflected attack because a malicious script can be run on the home page. An XSS Stored vulnerability was found on the Comments page, as script is able to be run on the page. The Loging.php toolbar is vulnerable to SQL injection, while the Networking.php page is vulnerable to a Command Injection attack. The VR Planner is susceptible to Local File Inclusion as files can be uploaded on the page.

Using OSINT, LetUsHackYou was able to search crt.sh to find a stored certificate. Admin user login credentials were stored directly in the HTML source code of the Login.php and was able to be seen by highlighting the page in a web browser. Apache server was found to be out-of-date with an Apache Struts vulnerability. The robots.txt file was shown to be exposed and readily accessible. Upon inspection of the Github repository, LetUsHackYou found access to user credentials that resulted in web hosts files and directories.

LetUsHackYou tested the Linux environment next. Nmap was used to find exposed 5 IP addresses and shown them as vulnerable to attack. Obtained credentials were used to access one host and escalate privileges to root. Another common shell RCE execution vulnerability was discovered with Meterpreter. The sudoers file was accessible with a Shellshock exploit found in Metasploit.

Lastly, LetUsHackYou tested the Windows OS environment. Using Metasploit, LetUsHackYou discovered FTP Port 21 and Port 110 (SLMail)were open and vulnerable. Metasploit was then used to exploit these vulnerabilities to gain access to a password hash file, which was cracked and enabled the creation of reverse shell. Scheduled tasks were readily visible within the Windows 10 Machine Task Scheduler. It was also found in Metepreter could be used to display directories on public Windows directories.

In summary, the found vulnerabilities could be exploited maliciously to cause catastrophic damage to assets and functionality of Rekall in general. LetUsHackYou has provided remediation recommendations detailing how to mitigate each vulnerability to prevent damage or loss that could result.

## 

## 

## 

## 

## Summary Vulnerability Overview

| **Vulnerability** | **Severity** |
| --- | --- |
| Sensitive Data Exposure | **Critical** |
| Local File Inclusion | **Critical** |
| SQL Injection | **Critical** |
| User Credentials Exposure | **Critical** |
| Command Injection | **Critical** |
| Shellshock on Web Servers (Port 80) | **Critical** |
| PHP Injection | **Critical** |
| Apache Struts CVE-2017-5638 | **Critical** |
| Drupal - CVE-2019-6340 | **Critical** |
| Linux Privilege Escalation | **Critical** |
| SLMail Port 110 Exploited via Metasploit (SeattleMail) | **Critical** |
| lsa\_dump\_sam via Kiwi Shows Password Hashes | **Critical** |
| IPs visible with Nmap | **Critical** |
| Admin Server Credentials Dumped via Kiwi | **Critical** |
| Open Source Exposed Data | **High** |
| Open FTP Port 21 | **High** |
| Apache Tomcat Remote Code Execution Vulnerability (CVE-2017-12617) | **High** |
| Sensitive Data Saved in Public/Documents | **High** |
| Run as ALL Sudoer (CVE-2019-14287) | **High** |
| XSS Stored | **Medium** |
| XSS Reversed | **Medium** |
| Weak Passwords | **Medium** |

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

| **Scan Type** | **Total** |
| --- | --- |
| Hosts | 192.168.13.10, 192.168.13.11, 192.168.13.12, 192.168.13.13, 192.168.13.14, 192.168.13.15, 172.22.117.10, 172.22.117.20 |
| Ports | 21, 25, 79, 80, 106, 110 |

| **Exploitation Risk** | **Total** |
| --- | --- |
| **Critical** | 14 |
| **High** | 5 |
| **Medium** | 3 |
| **Low** | 0 |

## Vulnerability Findings

| **Vulnerability 1** | **Findings** |
| --- | --- |
| **Title** | User Credentials Exposure |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | User Credentials are visible within the HTML of the Login.php and is visible when highlighting in a web browser. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Remove the User Credentials from HTML and implement 2-factor authentication. |

| **Vulnerability 2** | **Findings** |
| --- | --- |
| **Title** | Sensitive Data Exposure |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Medium |
| **Description** | Unrestricted access to robots.txt page. |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Restrict access to robots.txt to authorized users |

| **Vulnerability 3** | **Findings** |
| --- | --- |
| **Title** | Sensitive Data Exposure/Credentials Dump |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | Critical |
| **Description** | Use of previous successful exploit with Metasploit/Meterpreter session gave access to vulnerable passwords file obtained. Sensitive data was also found under Public\Documents\flag7.txt |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Update permissions on files and user permissions to restrict access to vulnerable files. Move files to an non-public domain. |

| **Vulnerability 4** | **Findings** |
| --- | --- |
| **Title** | Credentials Dump |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | Critical |
| **Description** | lsa\_dump\_sam was able to be used with Kiwi to gain Hashed NTLM. Hashed NTLM was cracked with john. |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Recommend salting the Hash. |

| **Vulnerability 5** | **Findings** |
| --- | --- |
| **Title** | SQL Injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | Accessing /Login.php page, a payload with Name or “1=1” was entered in the toolbar intended for a password successfully resulted in exploitation. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Remove ability for web app to accept direct input. |

| **Vulnerability 6** | **Findings** |
| --- | --- |
| **Title** | XSS Reflected |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Medium |
| **Description** | Malicious script was successfully reflected on host home page with <script>alert(“1”)</script> |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Input Validation |

| **Vulnerability 7** | **Findings** |
| --- | --- |
| **Title** | XSS Stored |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Medium |
| **Description** | Accessing /Login.php page and entering <script>alert(“Hi”)</script> to reveal the 3rd flag. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Implementation of XSS protection to disallow injection of script. |

| **Vulnerability 8** | **Findings** |
| --- | --- |
| **Title** | Local File Inclusion |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | Local File Inclusion was used on the VR Planner page by uploading .php file in the toolbar |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Prevent file paths from being appended directly. If possible, restrict API to allow inclusion only from a directory and the directories below it. |

| **Vulnerability 9** | **Findings** |
| --- | --- |
| **Title** | Command Injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | [www.welcometorecall.com](http://www.welcometorecall.com) && cat vendors.txt was used to gain access to sensitive data in vendors.txt. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Add input validation. |

| **Vulnerability 10** | **Findings** |
| --- | --- |
| **Title** | FTP Enumeration |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | Critical |
| **Description** | Port 21 was found open, which allowed for FTP enumeration through FTP connection on host IP. THis resulted in the successful transfer and access/download of vulnerable files. |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Restrict Access to Port 21 |

| **Vulnerability 11** | **Findings** |
| --- | --- |
| **Title** | Open Source Exposed Data |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | High |
| **Description** | On the Domain Dossier webpage, viewed the WHOIS data for totalrekall.xyz using OSINT. |
| **Images** |  |
| **Affected Hosts** | https://centralops.net/co/DomainDossier.aspx |
| **Remediation** | Remove any sensitive data from public records and clear this information in WHOIS records. |

| **Vulnerability 12** | **Findings** |
| --- | --- |
| **Title** | SLMail Exploit |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | Critical |
| **Description** | Open Port 110 resulted in successful exploitation through use of windows/pop3/seattlelab\_pass exploit. This was used within Metasploit to gain a successful Meterpreter session. |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Remove access to Port 110 and remove/replace SLmail service. |

| **Vulnerability 13** | **Findings** |
| --- | --- |
| **Title** | Certificate Exposure |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Medium |
| **Description** | Found stored certificate for totalrekall.xyz on crt.sh. |
| **Images** |  |
| **Affected Hosts** | 34.102.136.180 |
| **Remediation** | Remove exposed information on crt.sh and prevent future information from being exposed. |

| **Vulnerability 14** | **Findings** |
| --- | --- |
| **Title** | Nessus |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Medium |
| **Description** | Use of a Nessus scan showed the Apache Struts vulnerability. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.12 |
| **Remediation** | Regularly update Apache. |

| **Vulnerability 15** | **Findings** |
| --- | --- |
| **Title** | Nmap Scan |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | Nmap was used to scan 192.168.13.0/24, which revealed 5 exposed host IPs. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.10, 192.168.13.11, 192.168.13.12, 192.168.13.13, 192.168.13.14 |
| **Remediation** | Push IP blocking for all unauthorized users. |

| **Vulnerability 16** | **Findings** |
| --- | --- |
| **Title** | Aggressive Nmap Scan |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | Using nmap -A 192.168.13.0/24 we were able to discover the host that runs Drupal |
| **Images** |  |
| **Affected Hosts** | 192.168.13.13 |
| **Remediation** | Block probe attempts, restrict returned information or return misleading information, and slow down the aggressive nmap scan. |

| **Vulnerability 17** | **Findings** |
| --- | --- |
| **Title** | Privilege Escalation |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | Using SSH and stolen credentials escalated privileges were able to be obtained. sudo -u#-1 cat /root/flag12.txt |
| **Affected Hosts** | 192.168.13.14 |
| **Remediation** | Enforce stronger passwords, implement 2-factor authentication, and close port 22. |

| **Vulnerability 18** | **Findings** |
| --- | --- |
| **Title** | Port Scan of Subnet |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | Using a file found on the Github repository, stolen credentials were used to open 172.22.117.20 in a web browser due to port 80 being open. This allowed access to the flag2.txt file in the root directory |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Require stronger password and require 2-factor authentication |

| **Vulnerability 19** | **Findings** |
| --- | --- |
| **Title** | Github Username and Password Hash |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | Using credentials found in xampp.users from Github repository, the hash was cracked and access to the web server was obtained. |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Restrict access to Port 80 and remove xampp.users from Github repository. |

| **Vulnerability 20** | **Findings** |
| --- | --- |
| **Title** | Meterpreter Shell REC Execution (CVE 2017-5638) |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | With Meterpreter, multi/http/struts2\_content\_type\_ognl exploit was used to gain access to /root/flagisinThisfile.7z |
| **Affected Hosts** | 192.168.13.12 |
| **Remediation** | Apply vendor updates. |

| **Vulnerability 21** | **Findings** |
| --- | --- |
| **Title** | Shellshock on Port 80 |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | exploit/multi/http/apache\_mod\_cgi\_bash\_env\_exec with the target URL /cgi-bin/shockme.cgi and RHOST 192.168.13.11 were used to gain access to escalated privileges. |
| **Affected Hosts** | 192.168.13.11 |
| **Remediation** | Edit /etc/sudoers/ to limit access for all sudo accounts. |

| **Vulnerability 22** | **Findings** |
| --- | --- |
| **Title** | Public Directory Search |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | Medium |
| **Description** | Using ls in Meterpreter while in Users\Public\Documents directory, files in the directory were able to be displayed. |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Restrict unauthorized access and move files to non-public areas. |