//fff

| 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

[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** | Room9 |
| --- | --- |
| **Contact Name** | Cam |
| **Contact Title** | CISO |

## 

## Document History

| **Version** | **Date** | **Author(s)** | **Comments** |
| --- | --- | --- | --- |
| 001 | 16/01/2024 | Salin | First Draft |
| 002 | 18/01/2024 | Liam | Second Draft |
| 003 | 19/01/2024 | Nehaal | Third Draft |
| 004 | 20/01/2024 | Cam | Final Edit |
| 005 | 21/01/2024 | Nehaal, Salin, Cam, Liam | Final Review |

# 

## 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.

* Continual and continuous penetration testing is conducted to identify and address security vulnerabilities.
* The data validation appears to be present, however it can be bypassed.
* Room9 Inc was unable to hack passwords of certain individuals.
* Room9 Inc made several unsuccessful attempts to gain access via Metasploit.

### 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 (Stored and Reflected) and SQL payload injection.
* Credentials found in HTML source code.
* Apache web server was found to be outdated.
* SLMail server is vulnerable leading to multiple exploits.
* Access to hashes allowed for credentials and privilege escalation.
* nmap allowed us to see details of IP range’s and vulnerabilities (open ports, specific vulnerabilities, computer names, etc).
* Open Ports allowed for file enumeration and unauthorized access.

## Executive Summary

During the penetration testing of Rekall's IT assets, Room9 Inc identified critical vulnerabilities that pose a severe threat to the company's revenue and reputation. The web application is susceptible to XSS Reflected and Stored attacks, Local File Inclusion, and SQL Injection on specific pages. Room9 Inc was able to brute force a login and the Networking.php page is also vulnerable to Command Injection.

Open-source data, including a stored certificate, was exposed through OSINT, and user login credentials were shockingly stored in plain view in the HTML source code of the Login.php page. The robots.txt file and a GitHub repository revealed sensitive information, leading to unauthorized access to web hosts files. The Apache server had an outdated Struts vulnerability.

In the Linux environment, Room9 Inc exposed five exposed and vulnerable IP addresses. One host running Drupal was accessed using stolen credentials, and privileges were escalated to root. A common shell RCE execution vulnerability was found, and Shellshock exploit in Metasploit provided access to the sudoers file.

In the Windows environment, open and vulnerable FTP Port 21 and Port 110 (SLMail service) were discovered. Metasploit exposed these vulnerabilities and enabled access to a cracked password hash file. Scheduled tasks were visible in the Task Scheduler.

In conclusion, these vulnerabilities can lead to significant harm to Rekall's assets and business functionality. Room9 Inc has provided detailed recommendations for mitigating each vulnerability to prevent potential damage and loss.

## 

## Summary Vulnerability Overview

| **Vulnerability** | **Severity** |
| --- | --- |
| XSS Reflected | **Critical** |
| XSS Stored | **Critical** |
| Sensitive Data Exposure | **Critical** |
| Local File Inclusion | **Critical** |
| SQL Injection | **Critical** |
| Command Injection | **Critical** |
| Brute Force Attack | **Critical** |
| PHP Injection | **Critical** |
| Open source exposed credentials | **Critical** |
| Port 80 open | **Critical** |
| Windows/pop3/Seattlelab\_pass | **Critical** |
| scheduled tasks access | **Critical** |
| Nmap scan reveals sensitive information | **Critical** |
| Apache Struts | **Critical** |
| Shellshock on port 80 | **Critical** |
| Kiwi | **Critical** |
| Drupal | **Critical** |
| Open FTP port 21 | **Critical** |
| Psexec | **Critical** |
| Access Control | **Critical** |
| Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s) | **High** |
| Directory Traversal | **High** |
| open source exposed data | **High** |
| Session Management | **Medium** |
| Certificate search via crt.sh | **Medium** |

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

| **Scan Type** | **Total** |
| --- | --- |
| Hosts | Web App - 192.168.14.35  Linux - 192.168.13.1, 192.168.13.10, 192.168.13.12, 192.168.13.13 , 192.168.13.14  Windows - 172.22.117.10, 172.22.117.20 |
| Ports | Web app - Port 80, 110 Linux - Port 21, 80, 8080 Windows - Port 80, 445, 8080 |

| **Exploitation Risk** | **Total** |
| --- | --- |
| **Critical** | 20 |
| **High** | 3 |
| **Medium** | 2 |
| **Low** | 0 |

## Vulnerability Findings

| **Vulnerability 1** | **Findings** |
| --- | --- |
| **Title** | XSS Reflected |
| **Type (Web app / Linux OS / WIndows OS)** | Web APP |
| **Risk Rating** | **Critical** |
| **Description** | Malicious script reflected successfully on the home page |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Input Validation and removing the ability to input special characters such as >,/,etc |

| **Vulnerability 2** | **Findings** |
| --- | --- |
| **Title** | XSS stored |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | **Critical** |
| **Description** | We were able to comment on the malicious script on the comment box which was then stored in the database. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Input Validation and removing the ability to input special characters such as >,/,etc |

| **Vulnerability 3** | **Findings** |
| --- | --- |
| **Title** | Sensitive Data Exposure |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | **Critical** |
| **Description** | we could find the credentials for the admin account on the html page source on the login.php |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Removing the sensitive data which can be accessed from the html source |

| **Vulnerability 4** | **Findings** |
| --- | --- |
| **Title** | Local File Inclusion |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | **Critical** |
| **Description** | Local file inclusion successfully done by uploading the .php file |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Input validation and sanitization, Whitelist allowed files, Disable PHP remote file inclusion |

| **Vulnerability 5** | **Findings** |
| --- | --- |
| **Title** | SQL Injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | **Critical** |
| **Description** | we were able to do SQL injection attack on the user login for the login.php |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Input Validation and removing the ability to input special characters such as >,/,etc, and sanitization |

| **Vulnerability 6** | **Findings** |
| --- | --- |
| **Title** | Command Injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | **Critical** |
| **Description** | we were able to inject command on dns check toolbar by using [www.splunk.com](http://www.splunk.com) && cat /etc/passwd to view the users credentials |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Implement input validation and remove unintended access |

| **Vulnerability 7** | **Findings** |
| --- | --- |
| **Title** | Bruteforce |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | **Critical** |
| **Description** | We were able to guess the password for user melina which was same as username and were able to login into admin |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Using strong passwords consisting of uppercase,lowercase, numbers and special characters.  Set login attempt limits |

| **Vulnerability 8** | **Findings** |
| --- | --- |
| **Title** | PHP injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | **Critical** |
| **Description** | We were able to input a php injection to display userid |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Avoid passing user-submitted input to any filesystem/framework API |

| **Vulnerability 9** | **Findings** |
| --- | --- |
| **Title** | Open source exposed credentials |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | **Critical** |
| **Description** | We used OSINT to discover a user’s credential from Total rekall |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Educate users to not put there credentials on the web openly |

| **Vulnerability 10** | **Findings** |
| --- | --- |
| **Title** | Port 80 open |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | **Critical** |
| **Description** | we used trivera cracked hash to get into 172.22.117.20 and reveal sensitive information |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | using strong passwords consisting of uppercase,lowercase, numbers and special characters |

| **Vulnerability 11** | **Findings** |
| --- | --- |
| **Title** | Windows/pop3/Seattlelab\_pass |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | **Critical** |
| **Description** | We were able to access meterpreter shell by using the seattlelab\_pass module in metasploit |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Update all relevant software, including the Windows operating system and POP3 email server, with the latest security patches; change compromised or weak passwords, using strong, unique alternatives |

| **Vulnerability 12** | **Findings** |
| --- | --- |
| **Title** | Scheduled Access |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | **Critical** |
| **Description** | we were able to view scheduled task after getting access to the machine |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | restricting access to unauthorized users |

| **Vulnerability 13** | **Findings** |
| --- | --- |
| **Title** | Nmap scan reveals sensitive information |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | **Critical** |
| **Description** | Nmap scan on 192.168.13.0/24 revealed sensitive information |
| **Images** |  |
| **Affected Hosts** | 192.168.13.0/24 |
| **Remediation** | Implement IP blocking for unauthorized users |

| **Vulnerability 14** | **Findings** |
| --- | --- |
| **Title** | Apache Struts (CVE-2017-5638) |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | **Critical** |
| **Description** | Used exploit module multi/http/struts2\_content\_type\_ognl with a payload of linux/x86/shell\_reverse\_tcp to gain Meterpreter access. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.12 |
| **Remediation** | Update Apache Struts and have patch management |

| **Vulnerability 15** | **Findings** |
| --- | --- |
| **Title** | Shellshock on port 80 |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | **Critical** |
| **Description** | Used exploit module multi/http/tomcat\_jsp\_upload\_bypass with a payload of linux/x86/shell\_reverse\_tcp to gain Meterpreter access. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.1 |
| **Remediation** | Web Server Patching and have patch management |

| **Vulnerability 16** | **Findings** |
| --- | --- |
| **Title** | Kiwi |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | **Critical** |
| **Description** | Post exploitation, we were able to extract the NTLM hash associated with the user administrator |
| **Images** |  |
| **Affected Hosts** | 192.168.13.1 |
| **Remediation** | Moving the hashstore somewhere they couldn’t be extracted - also monitoring authentication/logon activity that could indicate an attack in progress |

| **Vulnerability 17** | **Findings** |
| --- | --- |
| **Title** | Drupal CVE |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | **Critical** |
| **Description** | We were able to create a meterpreter shell using exploit unix/webapp/drupal ws unserialize. Once accessed we were able to get uid. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.13 |
| **Remediation** | Update all relevant software, including the Linux operating system and |

| **Vulnerability 18** | **Findings** |
| --- | --- |
| **Title** | Open FTP port 21 |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | **Critical** |
| **Description** | A secure shell was created on the user Alice’s account through the open port 21 giving access to the host 192.168.13.14. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.14 |
| **Remediation** | Close port 21 and update the password for Alice. |

| **Vulnerability 19** | **Findings** |
| --- | --- |
| **Title** | Psexec |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | **Critical** |
| **Description** | We were able to escalate our privilege to the WINDC machine using windows/smb/psexec to create a meterpreter shell. Once done a shell account into the C:\\ drive was created granting access to the host 172.22.117.10. Once the shell was created we were able to find all the users in the system. |
| **Images** |  |
| **Affected Hosts** | 172.22.117.10 |
| **Remediation** | Immediately update the Admin password as it was easy to access and is not strong additionally. Update the password policy to increase both password strength and add a regular password update policy to have passwords regularly changed. |

| **Vulnerability 20** | **Findings** |
| --- | --- |
| **Title** | Access Control |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | **Critical** |
| **Description** | We were able to escalate privilege with meterpreter, and access the sudoers file |
| **Images** |  |
| **Affected Hosts** | 192.168.13.1 |
| **Remediation** | Harden access control - while preventing access to sudoers entirely might not be feasible, logging/alerting any changes/updates could flag any threat actors accessing sudoers. |

| **Vulnerability 21** | **Findings** |
| --- | --- |
| **Title** | Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s) |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | **High** |
| **Description** | Able to view sensitive information via "X-Powered-By" HTTP Response Header Field(s) |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Remove or Customize "X-Powered-By" Header |

| **Vulnerability 22** | **Findings** |
| --- | --- |
| **Title** | open source exposed data |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | **High** |
| **Description** | On the OSINT Framework webpage we were able to view the WHOIS data with OSINT for Totalrekall.xyz to access sensitive information |
| **Images** |  |
| **Affected Hosts** | totalrekall.xyz |
| **Remediation** | remove sensitive data shared publicly in WHOIS records |

| **Vulnerability 23** | **Findings** |
| --- | --- |
| **Title** | Directory traversal |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | **High** |
| **Description** | Through the use of directory traversal to access the disclaimer\_1.txt file, which should not be accessible. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Input Validation and removing the ability to input special characters such as >,/,etc |

| **Vulnerability 24** | **Findings** |
| --- | --- |
| **Title** | Session Management |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | **Medium** |
| **Description** | Through the use of burpsuite to test out different session ids in the admin\_legal\_data.php directory |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Encrypt the session id. |

| **Vulnerability 25** | **Findings** |
| --- | --- |
| **Title** | Certificate search via crt.sh |
| **Type (Web app / Linux OS / WIndows OS)** | Web app |
| **Risk Rating** | **Medium** |
| **Description** | searched for totalrekall.xyz in crt.sh found stored certificate |
| **Images** |  |
| **Affected Hosts** | 3.33.130.190,15.197.148.33 |
| **Remediation** | Protect information from being exposed by crt.sh site |