| 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** | JaPen |
| --- | --- |
| **Contact Name** | Penetration Tester |
| **Contact Title** | Jalen Hutchinson |

## 

## Document History

| **Version** | **Date** | **Author(s)** | **Comments** |
| --- | --- | --- | --- |
| 001 | 07/19/2023 | Jalen Hutchinson | - |

# 

## 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.

# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\*Web application: 192.168.14.35

\*Windows servers: 192.168.14.35, 192.168.13.10, 172.22.117.20

\*Linux servers: 192.168.13.1, 192.168.13.10 192.168.13.11, 192.168.13.12, 192.168.13.13

## 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.

Web application filtering for malicious code injection

The web application filters for malicious code injection by following these two practices:

* Only allowing .jpg file extensions for picture uploads. This prevents attackers from uploading files with malicious code extensions, such as .php or .exe.
* Filtering for command injections such as && and ;. These characters can be used by attackers to execute commands on the server. The web application filters for these characters and prevents them from being used in file names or other input fields.

These two practices help to protect the web application from malicious code injection attacks. With these practices followed, the web application helps to keep its users' data safe.

### 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.

Critical Vulnerabilities and Exploits:

Network Compromise Prevention Measures:

* Web Application Vulnerabilities:
  + Local file inclusion vulnerability in the web application.
  + Local file XSS vulnerability in the web application.
  + SQL injection vulnerability in the web application's login page.
  + Admin credentials found in the source code.
  + Sensitive data exposed in robots.txt.
  + PHP injection allowed for misuse of admin tools.
* Session Management Abuse:
  + Able to abuse session management using Burp Suite.

Exploited Network Assets:

* Exploits against IP Address 192.168.13.10:
  + Successful exploit using an Apache Tomcat vulnerability.
* Exploits against IP Address 192.168.13.11:
  + Successful exploit using a Shellshock vulnerability.
* Exploits against IP Address 192.168.13.12:
  + Successful exploit using an Apache Struts vulnerability.
* Exploits against IP Address 192.168.13.13:
  + Successful exploit using a Drupal vulnerability.
  + Able to SSH into 192.168.13.13 and escalate to root privileges.
* File Transfer and Exploit on Host 172.22.117.20:
  + Able to transfer files from the host 172.22.117.20 using FTP.
  + Exploited 172.22.117.20 with a POP3 exploit.
* Lateral Movement:
  + Able to move laterally using the PsExec exploit.

These critical vulnerabilities and exploits have been identified, and immediate action is recommended to address them in order to prevent unauthorized access and compromise of the network.

## Executive Summary

Penetration Test Findings and Recommendations:

Objective: The penetration test conducted by JaPen for Rekall aimed to assess the company's vulnerability to targeted attacks. The test focused on determining if remote attackers could breach ReKall's defenses and evaluating the impact of a security breach on the confidentiality of the company's data. The findings and vulnerabilities identified during the test are outlined below.

* Vulnerability Findings: The assessment revealed numerous vulnerabilities across Rekall's systems, including web applications, Linux servers, and Windows servers. The security position of Rekall was found to be critical, indicating a high risk of compromise. Specific

vulnerabilities discovered include:

* Web application vulnerabilities, such as local file inclusion, local file XSS, and SQL injection.
* Admin credentials found in the source code.
* Sensitive data exposed in robots.txt.
* User credentials obtained through command injection.
* PHP injection allows misuse of admin tools.
* Session management abuse using Burp Suite.
* Exploited vulnerabilities on specific IP addresses:
  + Successful exploits on IP address 192.168.13.10 using an Apache Tomcat vulnerability.
  + Successful exploit on IP address 192.168.13.11 using a Shellshock vulnerability.
  + Successful exploit on IP address 192.168.13.12 using an Apache Struts vulnerability.
  + Successful exploit on IP address 192.168.13.13 using a Drupal vulnerability.
  + Able to SSH into 192.168.13.13 and escalate to root privileges.
  + Able to transfer files from host 172.22.117.20 using FTP.
  + Exploited host 172.22.117.20 with a POP3 exploit.
  + Lateral movement achieved using the PsExec exploit.
* Remediation Recommendations: To improve Rekall's security posture and mitigate the identified vulnerabilities, the following actions are recommended:
* Address web application vulnerabilities promptly, including local file inclusion, local file XSS, and SQL injection.
* Remove admin credentials from source code and ensure secure credential management practices.
* Secure sensitive data by reviewing and revising robots.txt configuration.
* Investigate and remediate command injection vulnerabilities to prevent unauthorized access to user credentials.
* Implement proper input validation and sanitization to prevent PHP injection attacks.
* Strengthen session management mechanisms to prevent abuse.
* Patch or update systems with known vulnerabilities, such as Apache Tomcat, Shellshock, Apache Struts, and Drupal, to the latest versions.
* Review and enhance SSH configuration to prevent unauthorized access and privilege escalation.
* Improve FTP security and consider alternative secure file transfer methods.
* Address vulnerabilities on host 172.22.117.20 by patching or applying appropriate security measures.
* Implement network segmentation and access controls to limit lateral movement.
* Positive Aspects and Recommendations: While Rekall's security posture was found to be critical overall, certain strengths were identified during the penetration test. It is recommended to continue implementing and reinforcing these positive aspects. Further details about these strengths should be provided by JaPen.

Taking immediate and comprehensive actions based on the vulnerability findings and remediation recommendations will enable Rekall to significantly improve its security position. Effective planning, timely execution, and ongoing security measures are essential to safeguarding Rekall's systems and protecting the confidentiality of the company's data.

## 

## 

## 

## 

## Summary Vulnerability Overview

| **Vulnerability** | **Severity** |
| --- | --- |
| Cross-Site Scripting Reflected |  |
| Cross-Site Scripting Stored |  |
| Exposed Sensitive Data |  |
| Local File Intrusion |  |
| SQL Injection |  |
| Command Injection |  |
| Attack via Brute Force |  |
| PHP Injection |  |
| Session Management |  |
| Traversal of Directory |  |
| LLMNR Broadcasts |  |
| Secure Shell (SSH) |  |
| CVE-2019-14287 |  |
| OSINT Credentials |  |
| Weak Passwords |  |
| Windows Management Infrastructure |  |

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

| **Scan Type** | **Total** |
| --- | --- |
| Hosts | 8 |
| Ports | 28 |

| **Exploitation Risk** | **Total** |
| --- | --- |
| **Critical** | 11 |
| **High** | 5 |
| **Medium** | 0 |
| **Low** | 0 |

## Vulnerability Findings

Note: In my opinion, there is no such thing as a medium or low risk. I believe everything is at least high, and I believe this because if it is not seen as such it will be abused more and more, eventually becoming a high if not critical exploit on the risk chart. Please make sure to take all risks JaPen found as important and urgent, and use my remediations as a guide for what needs to be done to ensure these exploits are exterminated.

| **Vulnerability 1** | **Findings** |
| --- | --- |
| **Title** | Cross-Site Scripting Reflected |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | On the Welcome page (welcome.php), you are able to enter payloads into the “Begin by entering your name below!” section. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | * Prevent users from entering malicious characters into input fields. * Filter user input to remove special characters that could be used for malicious purposes. * Encode special characters in user input so that they are not interpreted as HTML or JavaScript code. * Use a web application firewall to block malicious traffic before it reaches your website. |

| **Vulnerability 2** | **Findings** |
| --- | --- |
| **Title** | Cross-Site Scripting Stored |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | Similar to the Welcome page, on the Comments page (comments.php) you are able to enter the same exact payload and get the same results. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | * Prevent users from entering malicious characters into input fields. * Filter user input to remove special characters that could be used for malicious purposes. * Encode special characters in user input so that they are not interpreted as HTML or JavaScript code. * Use a web application firewall to block malicious traffic before it reaches your website. |

| **Vulnerability 3** | **Findings** |
| --- | --- |
| **Title** | Local File Inclusion |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | High |
| **Description** | On the Memory Planner page (memory-planner.php), you are able to upload .php files with malicious scripts inside. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.35 |
| **Remediation** | As this page is for uploading images, only allow image associated files such as .jpeg and .png. |

| **Vulnerability 4** | **Findings** |
| --- | --- |
| **Title** | Exposed Sensitive Data |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | When looking into the HTML page source located on the login page (login.php), you are able to see the login for an admin. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Remove all sensitive data that is located in the HTML source page. |

| **Vulnerability 5** | **Findings** |
| --- | --- |
| **Title** | Cross-Site Scripting Reflected |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | Similar to the Welcome page and Comments page, you are able to enter the same exact payload on the Memory Planner page (memory-planner.php) where it asks “Who do you want to be?”, and get the same results once again. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | * Prevent users from entering malicious characters into input fields. * Filter user input to remove special characters that could be used for malicious purposes. * Encode special characters in user input so that they are not interpreted as HTML or JavaScript code. * Use a web application firewall to block malicious traffic before it reaches your website. |

| **Vulnerability 6** | **Findings** |
| --- | --- |
| **Title** | Command Injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | When on the Networking Tools page (networking.php), you are able to inject commands. Keep in mind we should not even be able to see this page, but due to the admin login credentials being located on the HTML page source of the login page, we can. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | * Prevent users from entering malicious characters into input fields. * Filter user input to remove special characters that could be used for malicious purposes. * Encode special characters in user input so that they are not interpreted as HTML or JavaScript code. * Use a web application firewall to block malicious traffic before it reaches your website. |

| **Vulnerability 7** | **Findings** |
| --- | --- |
| **Title** | SQL injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | By using a Boolean-based SQL injection on the second field of the login page (login.php), you are able to exploit. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | * Prevent users from entering malicious characters into input fields. * Filter user input to remove special characters that could be used for malicious purposes. * Encode special characters in user input so that they are not interpreted as HTML or JavaScript code. * Use a web application firewall to block malicious traffic before it reaches your website. |

| **Vulnerability 8** | **Findings** |
| --- | --- |
| **Title** | Whois Misuse |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | When using an OSINT tool such as Domain Dossier to find more information about the domain, you are able to see a vulnerability in the registrants' streets that could be used to exploit. |
| **Images** |  |
| **Affected Hosts** | totalrekall.xyz |
| **Remediation** | To prevent Whois misuse, registrants should use a privacy service to hide their contact information from the public.  They should also be careful about what information they share online, as this information could be used by attackers to exploit them. |

| **Vulnerability 9** | **Findings** |
| --- | --- |
| **Title** | Certificate Misuse |
| **Type (Web app / Linux OS / WIndows OS)** | Web App |
| **Risk Rating** | Critical |
| **Description** | When using an OSINT tool such as crt.sh to find more information about the certificate used for the site, you are able to find a vulnerability that could be used to exploit. |
| **Images** |  |
| **Affected Hosts** | totalrekall.xyz |
| **Remediation** | * Use a reputable certificate authority (CA). * Regularly check the status of their certificates. * Use a web application firewall (WAF). |