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

Rekall Corporation

DC National Cyber Security 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](#_heading=h.1fob9te)

[Contact Information 4](#_heading=h.2et92p0)

[Document History 4](#_heading=h.3dy6vkm)

[Introduction 5](#_heading=h.4d34og8)

[Assessment Objective 5](#_heading=h.3rdcrjn)

[Penetration Testing Methodology 6](#_heading=h.lnxbz9)

[Reconnaissance 6](#_heading=h.1ksv4uv)

[Identification of Vulnerabilities and Services 6](#_heading=h.2jxsxqh)

[Vulnerability Exploitation 6](#_heading=h.z337ya)

[Reporting 6](#_heading=h.3j2qqm3)

[Scope 7](#_heading=h.1y810tw)

[Executive Summary of Findings 8](#_heading=h.2xcytpi)

[Grading Methodology 8](#_heading=h.qsh70q)

[Summary of Strengths 9](#_heading=h.41mghml)

[Summary of Weaknesses 9](#_heading=h.49x2ik5)

[Executive Summary Narrative 10](#_heading=h.2grqrue)

[Summary Vulnerability Overview 1](#_heading=h.vx1227)3

Vulnerability Findings [1](#_heading=h.3fwokq0)4

# 

## Contact Information

|  |  |
| --- | --- |
| **Company Name** | DC National Cyber Security |
| **Contact Name** | Abel Woldemichael |
| **Contact Title** | Pen Tester |

## 

## Document History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author(s)** | **Comments** |
| 001 | Dec 1st 2024 | Abel Woldemichael | Pen Tester |

# 

## 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.

* Firewall protection was active during the reconnaissance phase
* Login credentials were functional
* Several open-source exploits on open ports were not successfully run on the server
* Domain control was isolated from other Windows machines.

### 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.

* Weak credentials were found in various locations, including Rekall’s repository, which was found through Google hacking, in Rekall web app HTML code, and lastly, in weak password configuration.
* Rekall had exposed sensitive information found through open sources. Nmap scans revealed numerous open ports.
* Rekall’s web application was vulnerable to both XSS and SQL attacks
* Rekall’s server and version were available to the public
* Outdated servers
* Scan of IPs showed possible vulnerabilities to IPs and ports

## Executive Summary

## The DC National Security Group (DCNSG) identified several critical vulnerabilities in Rekall's IT infrastructure. These vulnerabilities could severely impact the company's functionality and potentially damage its reputation. Here is an overview of how DCNSG discovered these vulnerabilities.

## 

## DCNSG initiated its penetration test by reviewing the Rekall web application. The first vulnerability identified was on Rekall's VR planning page, where penetration testers successfully executed a Cross-Site Scripting (XSS) attack, allowing malicious actors to run harmful scripts. Further investigation revealed that the same VR planner page was susceptible to Local File Inclusion, enabling the upload of malicious .php files. Testers could insert malicious scripts into the comments section on the comments page. Additionally, after reviewing the HTML of the web application, testers discovered login credentials that provided access to sensitive company data. SQL injection attacks were found to be executable on the login page, the networking page, and even through the URL bar.

## 

## After examining the web application, the penetration testers utilized open-source intelligence (OSINT) to identify additional exposed vulnerabilities. They discovered Rekall’s stored certificate, which exposed sensitive data, and performed a DNS lookup that revealed further vulnerabilities within the company's network infrastructure. Through NMAP scans, DCNSG identified open ports on the network, the host being used, and critical information regarding the server and its version. The scans showed that the running Apache server was not up to date and was vulnerable to known exploits.

## 

## With the data collected from the NMAP scans, DCNSG attempted to exploit several known vulnerabilities using Metasploit. They successfully executed CVE-2017-5638, an Apache exploit, which allowed them to access sensitive company data and execute commands that could alter the data. Another exploit used was CVE-2014-6271 (Shellshock), through which they escalated privileges with stolen credentials to access the company’s sudoers file. They also leveraged CVE-2014-6340, which provided them with Rekall’s server username, www-data. Additionally, CVE-20030264 was executed in Metasploit successfully, allowing access to Rekall’s data through SLMail.

## 

## The team employed alternative methods to access Rekall's data as well. For example, they used SSH to access Alice’s account via the IP address 192.168.13.14. Alice’s account had a weak password, which made it easy to guess the credentials. Another method utilized was Google hacking, which led to the discovery of the Rekall repository containing Trivera's username and hashed password. Using John the Ripper, they cracked the password and successfully logged into the database with those credentials. Further analysis of the NMAP scan revealed that 172.22.117.20 was using an open port (port 80), allowing the team to access the Index of / and uncover sensitive information. They also found that the information in the NMAP scan pointed to using a file transfer protocol where sensitive data could be stored in plain text files.

## 

## In summary, DCNSG was able to uncover 11 critical vulnerabilities that could significantly harm the company's reputation, disrupt day-to-day operations, and potentially lead to financial damage. Below, the team has outlined mitigation strategies to address these vulnerabilities and minimize Rekall’s attack surface.

## 

## Summary Vulnerability Overview

|  |  |
| --- | --- |
| **Vulnerability** | **Severity** |
|  | **Critical** |
| SQL Injection | **Critical** |
| Local File Inclusion | **Critical** |
| Credentials in the HTML | **Critical** |
| NMAP Scan Vulnerabilities | **Critical** |
| CVE 2017-5638 – Apache Struts Vulnerability RCE | **Critical** |
| CVE-2014-6271 – Apache Mod\_cgi Bash Environment Variable Code Injection (Shell Shock) | **Critical** |
| CVE-2019-6340 - Drupal RESTful Web Services unsterilized RCE | **Critical** |
| Remote SSH | **Critical** |
| Directory Traversal | **Critical** |
| CVE-2003-0264 – Multiple Buffer Overflows in SLMail | **Critical** |
| FTP Protocol Vulnerabilities | **High** |
| XSS Stored | **High** |
| XSS Reflected | **Medium** |
| HTML Command Input | **Medium** |
| Unencrypted Traffic | **Medium** |
| Command Injection | **Medium** |
| Open Source data | **Medium** |
| DNS Lookup Record | **Medium** |
| Nessus Scan | **Medium** |
| Windows Task Scheduler | **Medium** |
|  |  |

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

|  |  |
| --- | --- |
| **Scan Type** | **Total** |
| Hosts | 172.22.117.10  172.22.117.20  192.168.13.10  192.168.13.11  192.168.13.12  192.168.13.13  192.168.13.14  192.168.13.35 |
| Ports | 21, 22, 80, 106, 110 |

|  |  |
| --- | --- |
| **Exploitation Risk** | **Total** |
| **Critical** | 11 |
| **High** | 2 |
| **Medium** | 8 |
| **Low** | 0 |

## Vulnerability Findings

|  |  |
| --- | --- |
| **Vulnerability 1** | **Findings** |
| **Title** | XSS Reflected |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Medium |
| **Description** | Scripts were allowed to be injected on the Rekall web app which malicious scripts can exploit. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Input Validation |

|  |  |
| --- | --- |
| **Vulnerability 2** | **Findings** |
| **Title** | HTML Command Input |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Medium |
| **Description** | The "Who do you want to be?" page of the web application's HTML has a vulnerability that allows malicious scripts to be injected. This occurs due to insufficient input validation and sanitization, enabling attackers to execute XSS attacks, which can compromise user data and the application's security. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Input Validation |

|  |  |
| --- | --- |
| **Vulnerability 3** | **Findings** |
| **Title** | XSS Stored |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | High |
| **Description** | Another vulnerability identified in Rekall's comment page is the lack of input validation and sanitization, which allows attackers to inject and execute malicious XSS scripts within the comments section. This can lead to unauthorized access to user data, session hijacking, and other security risks for users interacting with the comment page.. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Input Validation |

|  |  |
| --- | --- |
| **Vulnerability 4** | **Findings** |
| **Title** | Unencrypted Traffic |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Medium |
| **Description** | Using the Curl command-line tool, we discovered that port 80 on the server was open and transmitting data without encryption. This vulnerability allows attackers to intercept and potentially manipulate web traffic, posing a significant security risk. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Redirect all web traffic to HTTPS (Port 443) to help mitigate potential risks. |

|  |  |
| --- | --- |
| **Vulnerability 5** | **Findings** |
| **Title** | Local File Inclusion |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Critical |
| **Description** | A vulnerability was identified on Rekall's VR Planner page. The application permits the upload of files without proper validation, leading to a Local File Inclusion attack. This allows attackers to upload and execute malicious files on the server, potentially gaining unauthorized access to sensitive data and system resources. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Implemented restrictions to only allow the upload of JPG files and enhanced input validation to prevent the injection of malicious PHP files. |

|  |  |
| --- | --- |
| **Vulnerability 6** | **Findings** |
| **Title** | Local File Inclusion |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Critical |
| **Description** | Rekall's second file upload feature was configured to accept only JPG files. However, we discovered a vulnerability by renaming a PHP file to have a .JPG extension. This bypassed the file type restriction and allowed us to upload and execute a malicious PHP script, indicating a lack of proper content validation beyond file extension checks. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Ensure that the server validates the MIME type of the uploaded file to confirm it matches the expected content type (image/jpeg for JPG files). Implement additional checks to inspect the file content and confirm it adheres to the JPG file structure, regardless of the file extension. Verify that the file extension and the MIME type are consistent with each other to prevent mismatches. Sanitize and validate all inputs on the server side to ensure no malicious content is executed. |

|  |  |
| --- | --- |
| **Vulnerability 7** | **Findings** |
| **Title** | SQL Injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Critical |
| **Description** | On the user login page, we identified a vulnerability by injecting an SQL payload containing an always-true statement (1=1). This payload bypassed the authentication process and overrode all other SQL queries, indicating that the application is susceptible to SQL Injection attacks. This flaw allows attackers to gain unauthorized access to the system by manipulating the SQL queries executed by the application. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Input Validation: Ensure only the intended login credentials can be used. |

|  |  |
| --- | --- |
| **Vulnerability 8** | **Findings** |
| **Title** | Login Credentials in the HTML |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Critical |
| **Description** | In the HTML code of the Rekall login page, the penetration tester discovered login credentials for the user "dougquiad." This security lapse allowed the tester to successfully log in using these credentials, indicating that sensitive information was improperly stored within the client-side code. This vulnerability exposes the application to unauthorized access and potential misuse. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Removal of credentials |

|  |  |
| --- | --- |
| **Vulnerability 9** | **Findings** |
| **Title** | Command Injection |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Medium |
| **Description** | On the Rekall Welcome page, we identified a vulnerability where malicious commands could be injected into the HTML. This allowed unauthorized access to files in the database, indicating insufficient input validation and sanitization. This flaw exposes the application to potential data breaches and unauthorized data manipulation. |
| **Images** |  |
| **Affected Hosts** | 192.168.14.35 |
| **Remediation** | Input Validation, limiting user input, running web server under a unique user account |

|  |  |
| --- | --- |
| **Vulnerability 10** | **Findings** |
| **Title** | Open Source data |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Medium |
| **Description** | Using open-source intelligence (OSINT) techniques, we discovered that Rekall's sensitive data was publicly accessible. This vulnerability indicates a failure to properly secure and restrict access to sensitive information, exposing the data to potential misuse and unauthorized access by malicious actors. |
| **Images** |  |
| **Affected Hosts** | totalrekall.xyz |
| **Remediation** | Remove sensitive data that could potentially risk the company's functionality. Limit access to DNS records. Regularly monitor WHOIS changes. |

|  |  |
| --- | --- |
| **Vulnerability 11** | **Findings** |
| **Title** | DNS Lookup Record |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Medium |
| **Description** | During our assessment, we discovered Rekall's DNS records, which exposed sensitive information about the company's servers, internal systems, and infrastructure. This vulnerability reveals details such as IP addresses, server locations, and internal network configurations, potentially aiding attackers in planning targeted attacks and exploiting weaknesses within Rekall's network. |
| **Images** |  |
| **Affected Hosts** | Totalrekall.xyz |
| **Remediation** | Limit DNS Public exposure, also remove sensitive subdomains to prevent exposure of internal network structure, |

|  |  |
| --- | --- |
| **Vulnerability 12** | **Findings** |
| **Title** | Certificates |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Medium |
| **Description** | We discovered that Rekall's website stored its security certificate files on the web server in an accessible location. This vulnerability exposes the security certificate to potential unauthorized access, compromising the integrity and confidentiality of the data transmitted between the server and its users. |
| **Images** |  |
| **Affected Hosts** | Totalrekall.xyz |
| **Remediation** | Hardening Rekall’s security by using certificate storage, such as a cloud key management system such as Azure Key Vault, ensures strong private key protection and limits access to those who have the key. |

|  |  |
| --- | --- |
| **Vulnerability 13** | **Findings** |
| **Title** | NMAP Scan |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | An NMAP scan conducted on the IP range 192.168.13.0/24 revealed several active hosts with open network ports. This scan provided detailed information about each host's network services and potential vulnerabilities, indicating the presence of exploitable entry points within the network. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.0/24 |
| **Remediation** | Strengthen a firewall to restrict access to vulnerable ports, close all unnecessary ports, and ensure only trusted IP addresses can access information. |

|  |  |
| --- | --- |
| **Vulnerability 14** | **Findings** |
| **Title** | Aggressive NMAP Scan |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | After conducting an aggressive NMAP scan, we identified a host running on Drupal. The scan revealed detailed information about the host's open ports, services, and possible vulnerabilities specific to the Drupal content management system. This information indicates potential security weaknesses that could be exploited, highlighting the need for further investigation and remediation to secure the Drupal installation. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.13 |
| **Remediation** | Strengthen a firewall to restrict access to vulnerable ports, close all unnecessary ports, and ensure only trusted IP addresses can access information. |

|  |  |
| --- | --- |
| **Vulnerability 15** | **Findings** |
| **Title** | Nessus Scan |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Medium |
| **Description** | By running a Nessus scan, we identified several open-source vulnerabilities related to Rekall's server, specifically linked to the current version of Apache it is using. These vulnerabilities include known security flaws that could be exploited by attackers to gain unauthorized access, execute arbitrary code, or disrupt server operations, indicating that the Apache server version in use is outdated and lacks critical security patches. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.0/24 |
| **Remediation** | Ensure the server is updated to the newest version to minimize Rekall’s attack surface. |

|  |  |
| --- | --- |
| **Vulnerability 16** | **Findings** |
| **Title** | CVE 2017-5638 – Apache Struts Vulnerability RCE |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | By exploiting the open-source vulnerability identified as CVE-2017-5638, we were able to execute arbitrary commands on the Rekall server. This security flaw, associated with a known vulnerability in Apache Struts, provided unauthorized access to sensitive data stored on the server. The exploitation of this vulnerability highlights the critical need for timely patching and updating of software components to mitigate such risks. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.10 |
| **Remediation** | Ensuring all servers are updated and patched to protect against open-source vulnerabilities. |

|  |  |
| --- | --- |
| **Vulnerability 17** | **Findings** |
| **Title** | CVE-2014-6271 – Apache Mod\_cgi Bash Environment Variable Code Injection (Shell Shock) |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | Another exploit utilized in Metasploit was the multi/HTTP/apache\_mod-cgi\_bash-env\_exec exploit, commonly known as the Shellshock vulnerability, which targeted port 80. By exploiting this vulnerability, we gained unauthorized access to the server and were able to read and modify the /etc/sudoers file, thereby escalating our privileges to root. With root access, we subsequently accessed the /etc/passwd file, revealing a list of all available users on the system. This exploitation highlights critical security flaws in the server’s configuration and underscores the importance of patching known vulnerabilities. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.11 |
| **Remediation** | Restricting access to the sudoers file. Updating the server to the most recent version. |

|  |  |
| --- | --- |
| **Vulnerability 18** | **Findings** |
| **Title** | CVE-2019-6340 Drupal RESTful Web Services unsterilized RCE |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | By exploiting the vulnerability using the Metasploit module exploit/unix/webapp/drupal\_restws\_unserialized, we were able to gain unauthorized access through the Meterpreter session. This allowed us to retrieve the server's username, a critical piece of information that could be leveraged for further attacks such as brute-force attempts or password spraying. This vulnerability highlights significant security weaknesses in the Drupal RESTful Web Services module, emphasizing the need for immediate remediation to prevent unauthorized access and potential data breaches. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.13 |
| **Remediation** | To mitigate this vulnerability, it’s important to have strong MFA and authentication mechanisms and monitor suspicious activity to limit username exposure. |

|  |  |
| --- | --- |
| **Vulnerability 19** | **Findings** |
| **Title** | Remote SSH |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | This vulnerability demonstrates how we successfully gained unauthorized access to Alice's account via SSH. Once inside her account, we exploited additional security weaknesses to escalate her privileges, allowing us to perform actions and access data beyond her initial permissions. This highlights significant flaws in the account security and privilege management systems, which need to be addressed to prevent potential exploitation by malicious actors. |
| **Images** |  |
| **Affected Hosts** | 192.168.13.14 |
| **Remediation** | Use stronger credentials. MFA is encouraged. |

|  |  |
| --- | --- |
| **Vulnerability 20** | **Findings** |
| **Title** | Password Hash |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Critical |
| **Description** | In this vulnerability assessment, we discovered that Trivera’s credentials were inadvertently exposed on GitHub. By locating the credentials, we were able to retrieve a hashed password. Utilizing password-cracking techniques, we successfully decrypted the hash, revealing the actual username and password. This breach illustrates the critical risk associated with improper handling of sensitive information on public repositories, emphasizing the necessity for secure credential management practices. |
| **Images** |  |
| **Affected Hosts** | Rekall Web Server |
| **Remediation** | Remove credentials from Github. |

|  |  |
| --- | --- |
| **Vulnerability 21** | **Findings** |
| **Title** | Directory Traversal |
| **Type (Web app / Linux OS / WIndows OS)** | Web Application |
| **Risk Rating** | Critical |
| **Description** | In this vulnerability, we successfully accessed the directory listing of the web server at IP address 172.22.117.20 via HTTP. By navigating to the URL, we were able to view the "Index of /" page, which exposed a list of files and directories stored on the server. This exposure highlights a misconfiguration issue, as directory indexing should be disabled to prevent unauthorized users from accessing potentially sensitive files and information. |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Input validation. Restrict user inputs to avoid directory traversal. Use absolute paths or proper permissions so only root users can access these files. |

|  |  |
| --- | --- |
| **Vulnerability 22** | **Findings** |
| **Title** | FTP Protocol Vulnerabilities |
| **Type (Web app / Linux OS / WIndows OS)** | Linux OS |
| **Risk Rating** | Critical |
| **Description** | This vulnerability allowed us to access the File Transfer Protocol (FTP) service, which exposed login credentials being transmitted in plain text. This security flaw means that malicious attackers could intercept these unencrypted credentials during transmission, potentially gaining unauthorized access to the FTP server and its associated resources. This highlights the critical need for secure transmission protocols, such as FTP over SSL/TLS (FTPS), to protect sensitive information from interception and misuse. |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Use FTPS or SFTP to help encrypt the communication channel and protect data and credentials. |

|  |  |
| --- | --- |
| **Vulnerability 23** | **Findings** |
| **Title** | CVE-2003-0264 |
| **Type (Web app / Linux OS / WIndows OS)** | Linux |
| **Risk Rating** | Critical |
| **Description** | In this vulnerability, we identified an exploit targeting the Windows Mail Server. This exploit allows unauthorized access to the server, potentially exposing sensitive email communications and user credentials. Attackers can leverage the vulnerability to gain control over the mail server, manipulate email data, and compromise the overall security of the organization's email infrastructure. This discovery underscores the importance of applying security patches and updates to protect against such exploits. |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Updating to the latest version |

|  |  |
| --- | --- |
| **Vulnerability 24** | **Findings** |
| **Title** | Windows Task Scheduler |
| **Type (Web app / Linux OS / WIndows OS)** | Windows OS |
| **Risk Rating** | Medium |
| **Description** | In this vulnerability, we successfully accessed the Task Scheduler on the target system. This allowed us to view detailed information about scheduled tasks, including their triggers, actions, and configurations. By gaining insight into these scheduled tasks, we identified potential security weaknesses, such as improperly configured tasks or those running with elevated privileges, which could be exploited to execute malicious actions or escalate privileges on the system. |
| **Images** |  |
| **Affected Hosts** | 172.22.117.20 |
| **Remediation** | Edit the permission on who can access these accounts to only valid users. |