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

Contact Information

Company Name	SOUTHTX.TECH, LLC (STX)
Contact Name	Andrew Cheetham
Contact Title	President & CEO, Penetration Tester

Document History

Version	Date	Author(s)	Comments
001	June 5, 2023	Andrew Cheetham	
002	August 18, 2023	Andrew Cheetham	Updated vulnerability screenshots.

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 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 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.

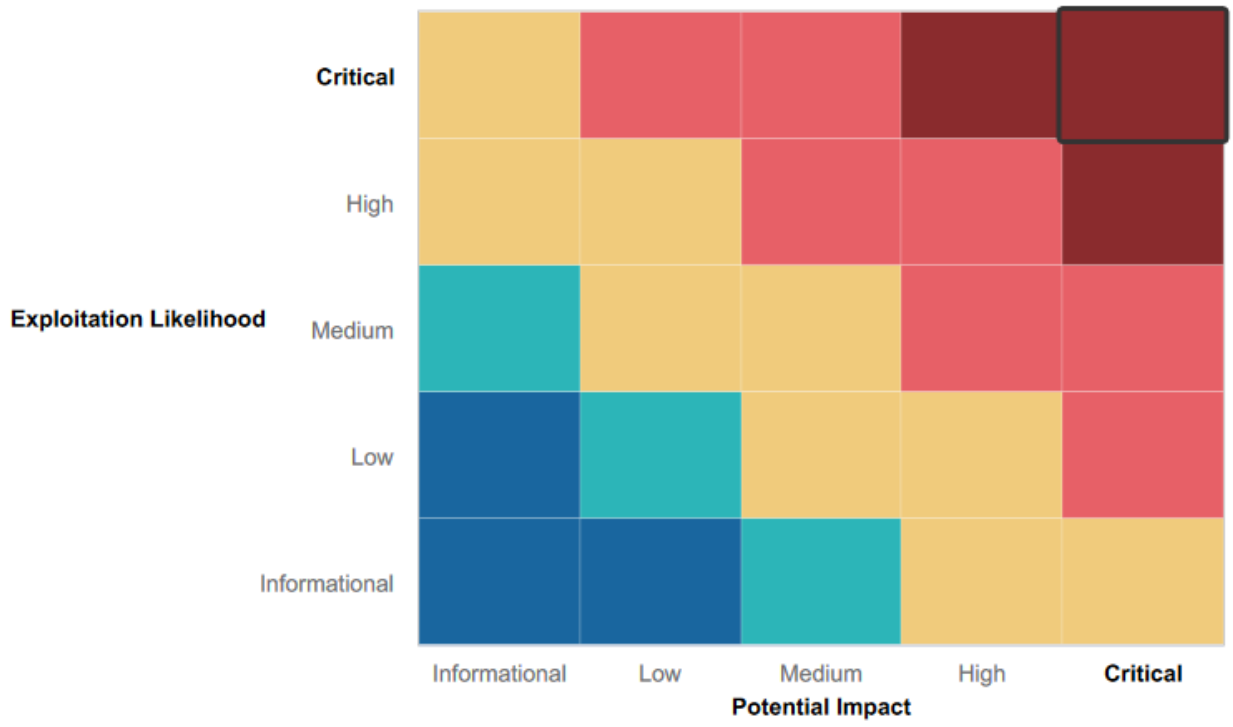
Executive Summary of Findings

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:



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.

1. Rekall's public facing domain has password protected logins for both users and administrators.
2. Some common exploits on open ports are rejected by the server (access denied on vnc exploits on web application server, for example).
3. The Windows network domain controller is isolated from other machines and the only available attack surface can be obtained by compromising another Windows machine on the same local network.

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.

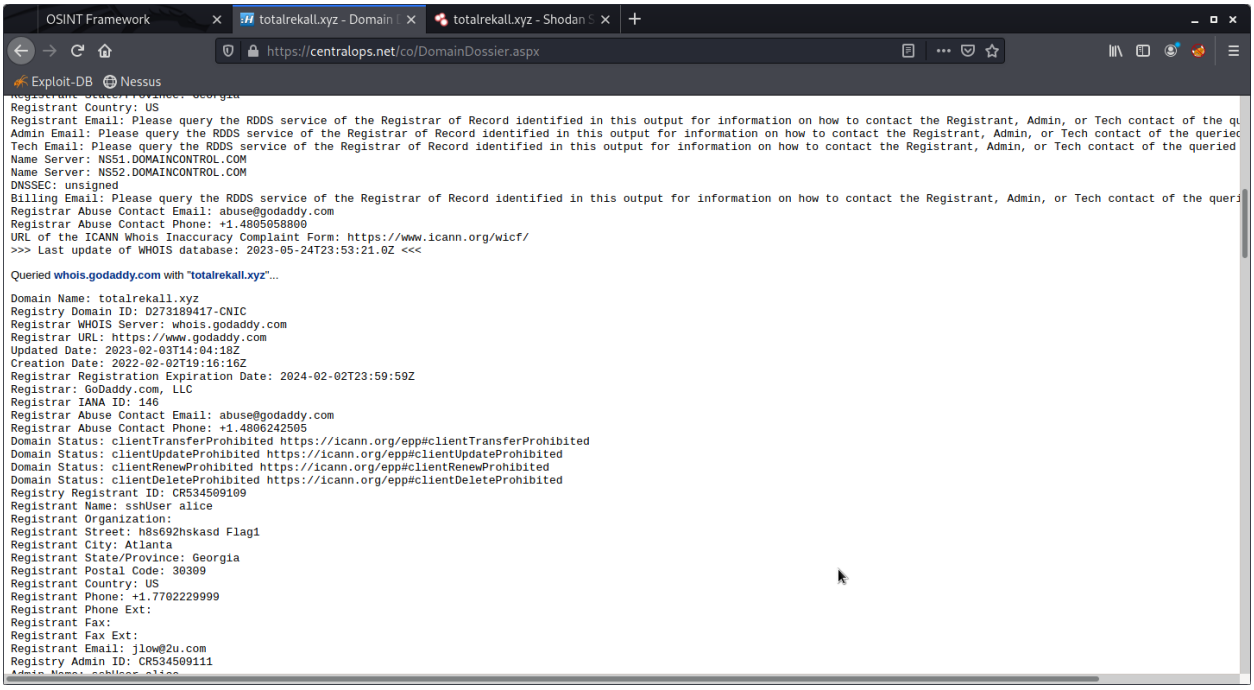
1. Some users and administrators are using weak passwords, including *Tanya4life* and *Changeme!*
2. Server info on Rekall's web application is publicly accessible, including OS and kernel/service version numbers. In addition, at least one hidden file is viewable to the public at the web application IP address.
3. Domain IP network on the two hosts used to deliver Rekall's web app shows open ports, indicating possible vulnerabilities using services such as HTTP, VNC, X11 and mySQL. Critically, the Rekall web app is vulnerable to XSS attacks in multiple login fields.
4. Network with 5 Linux machines with various open ports and vulnerabilities was found at IP range 192.168.13.0/24.
5. Rekall's Linux server network contains 5 machines, all of which have open ports and vulnerabilities with various levels of exploitability. One Linux server has a critical vulnerability that could be corrected by updating both OS and application software.
6. Rekall's Windows network is vulnerable to attacks from Metasploit, including access gained due to unpatched software, credential dumping techniques, and lateral movement.

Executive Summary

In the reconnaissance phase of our penetration test of the Rekall network, STX was able to quickly determine Rekall's primary IP address, which is not unusual. More unusual is that we were able to use the Nmap software to locate an additional IP address on the network which had two open ports and thus several vulnerabilities. Additional scans of both hosts revealed pertinent information that an attacker would likely leverage while attempting to gain control of one or both machines, including their operating systems and kernel versions, and service names and versions.

Though Rekall's login page is password protected, every field on the site (including the login page) is vulnerable to attack which allows an attacker access to a Linux shell on the server. This type of exposure is easily corrected through input validation techniques and sanitization, discussed later. Using common search operands, STX was able to exploit information learned to locate a file with sensitive company information that assisted in further attacks.

Further reconnaissance uncovered a Rekall network of five Linux machines in the IP range of 192.168.13.0/24, as shown in the screenshot below.



All of these machines contained open ports and outdated OS and service software, any of which could be successfully exploited by attackers to gain sensitive information directly and through a remote administrator shell. For example, at 192.168.13.12, STX was able to gain a shell using a common Metasploit module entitled *exploit/multi/http/tomcat_jsp_upload_bypass.* Once we achieved access to this machine on the network, we were able to use common exploit techniques such as lateral movement and privilege escalation to achieve shells on other machines in the network.

It's important to note that an experienced attacker may already know what vulnerabilities and software versions are likely to provide easiest access to a machine, but even an inexperienced hacker can gain access using freely available tools such as Nessus to determine the severity of exploits on a given machine.

STX also used simple search techniques to uncover the Github repository for Rekall's developer(s). An open directory there contained user credentials, as shown below.

totalrekall / site Public

Code Issues Pull requests 1 Actions Projects Security Insights

main site / xampp.users

totalrekall Added site backup files

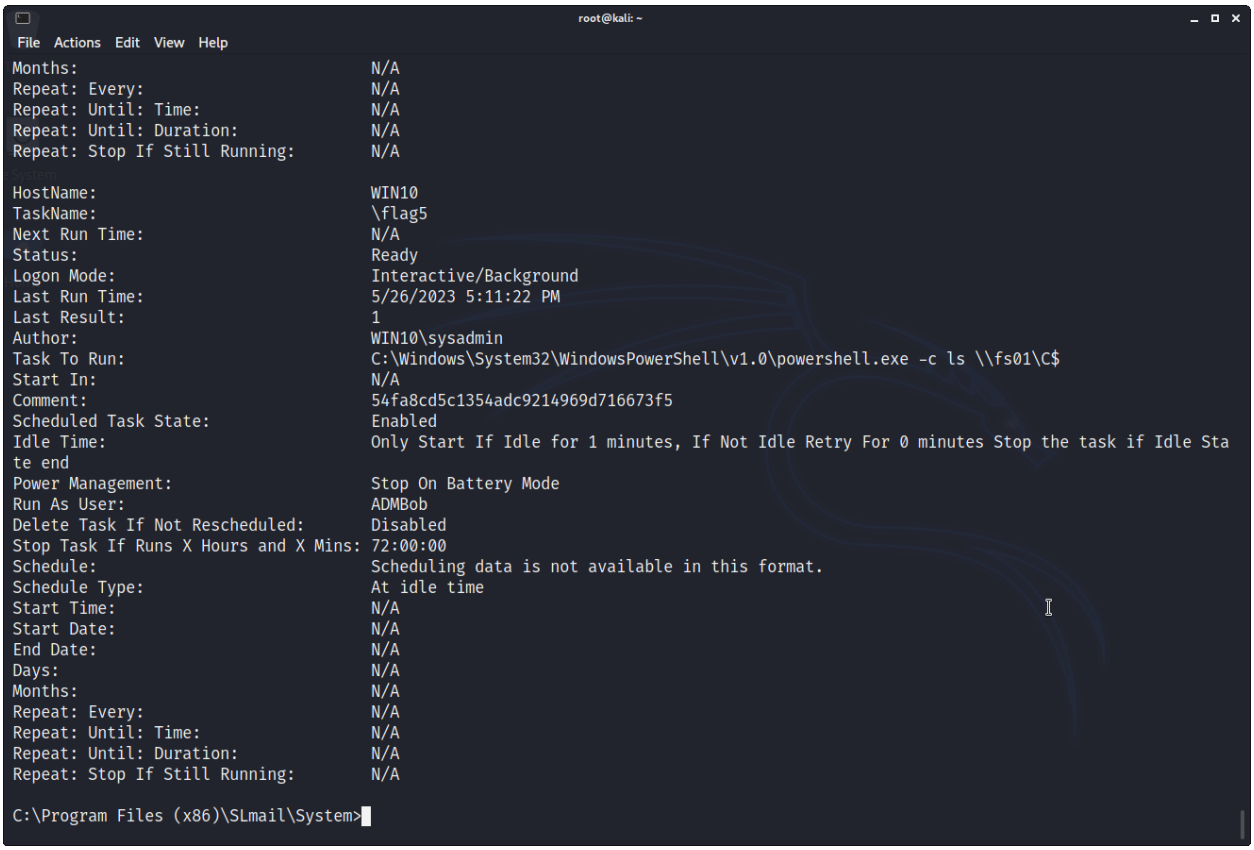
1 contributor

1 lines (1 sloc) 46 Bytes

```
1  trivera:$apr1$A0vSKwao$GV3sgGAj53j.c3GkS4oUC0
```

Though these credentials included a password hash rather than a plain-text password, the hash was easily cracked and applied to obtain access to the root directory of another server.

Using the credentials obtained, STX was able to gain a command (SYSTEM-level) at the host Windows 10 using a common Metasploit module. This exploit can be repeated each time an attacker wants to access this machine, and we successfully used the same attack against the Win10 machine numerous times to gain access. As shown in the screenshot below, we were able to create an obscured task within Windows 10's Task Scheduler system. An attacker might use a method such as this to create an automatic call-back to their system to keep open a malicious connection.



Using the same Metasploit module to gain access to the Win10 host, we were able to use a different software tool to print (to the screen) additional *username:password hash* credentials. This was

achieved by using an exploit named Mimikatz in which a non-compromised machine can be successfully attacked from a compromised machine on the same local network. From this attack, we collected our second set of administrator credentials, *ADMBob:Changeme!*

In the last phase of our penetration testing, we determined that the Rekall's domain controller machine, WinDC01, is vulnerable to various lateral movement exploits, allowing an attacker to gain SYSTEM-level access from another compromised machine, in this case, Win10. This is perhaps the most critical vulnerability we discovered; it would allow an attacker to gain user and administrator credentials for any other machines on the Rekall's Windows network.

In conclusion, we suggest that Rekall follow the remediation suggestions outlined below for each of the eleven vulnerabilities that we identified, then follow that up with another full penetration test.

Summary Vulnerability Overview

Vulnerability	Severity
1. Server info vulnerable to Nmap scans	**Medium**
2. Network scan shows hosts with open ports	**Medium**
3. Subdomain search shows hidden .txt file	**High**
4. Web application vulnerable to XSS attacks	**High**
5. Network with 5 Linux machines with open ports and vulnerabilities found	**Medium**
6. Nessus scan of machine in the Linux network returned critical vulnerability	**Critical**
7. Admin shell achieved at .12 machine with Metasploit module.	**Critical**
8. User credentials *user:hash* posted in unprotected Github repository	**Critical**
9. Win10 machine vulnerable to repeatable pop3 exploit	**Critical**
10. Win10 machine vulnerable to Mimikatz/kiwi credential snooping	**High**
11. WinDC01 machine vulnerable to lateral movement exploits	**Critical**

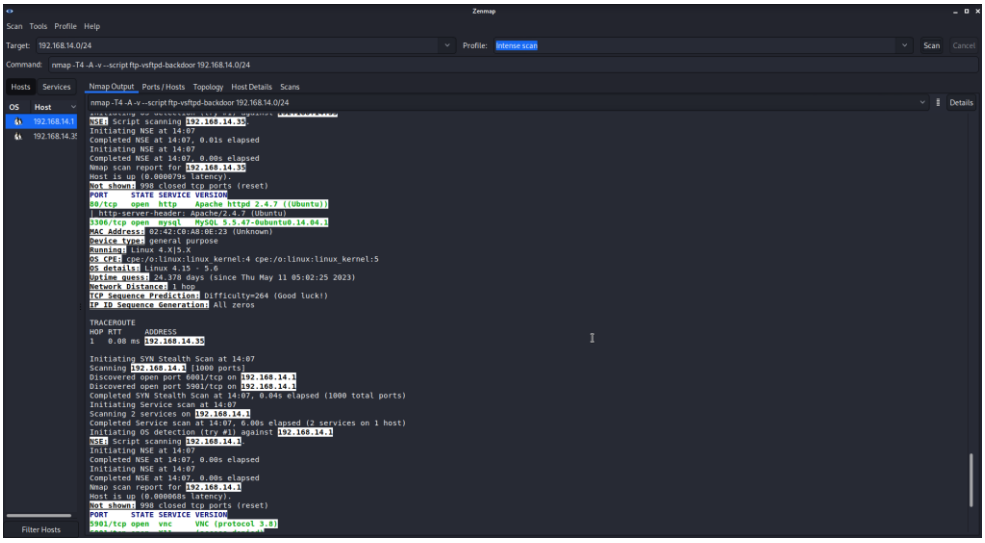
The following summary tables represent an overview of the assessment findings for this penetration test:

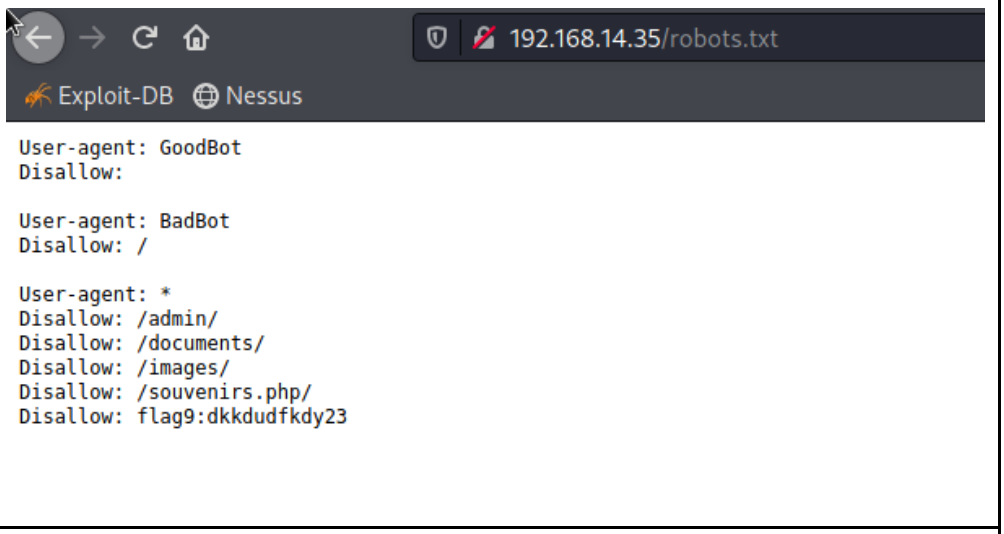
Scan Type	Total
Hosts	• 192.168.14.35 and .1 • 192.168.13.1 and .10, .12, .13, .14 • 172.22.117.10 and .20
Ports	• 80, 3306, 5901, 6001 • 5901, 6001, 22, 80, 8080, 8009 • 21, 25, 53, 79, 80, 88, 106, 110, 135, 139, 389, 443, 445, 464, 593, 636, 3268, 3269

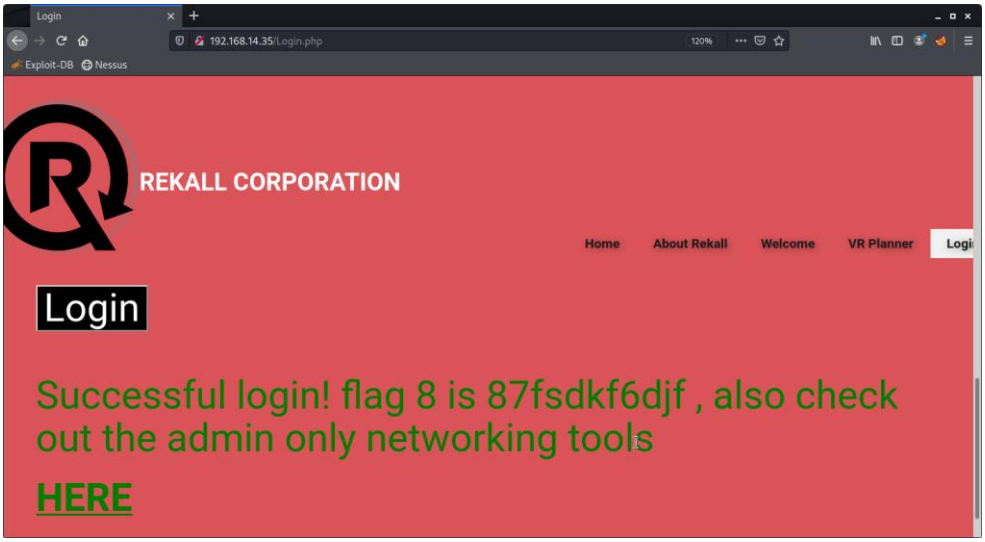
Exploitation Risk	Total
Critical	5
High	3
Medium	3

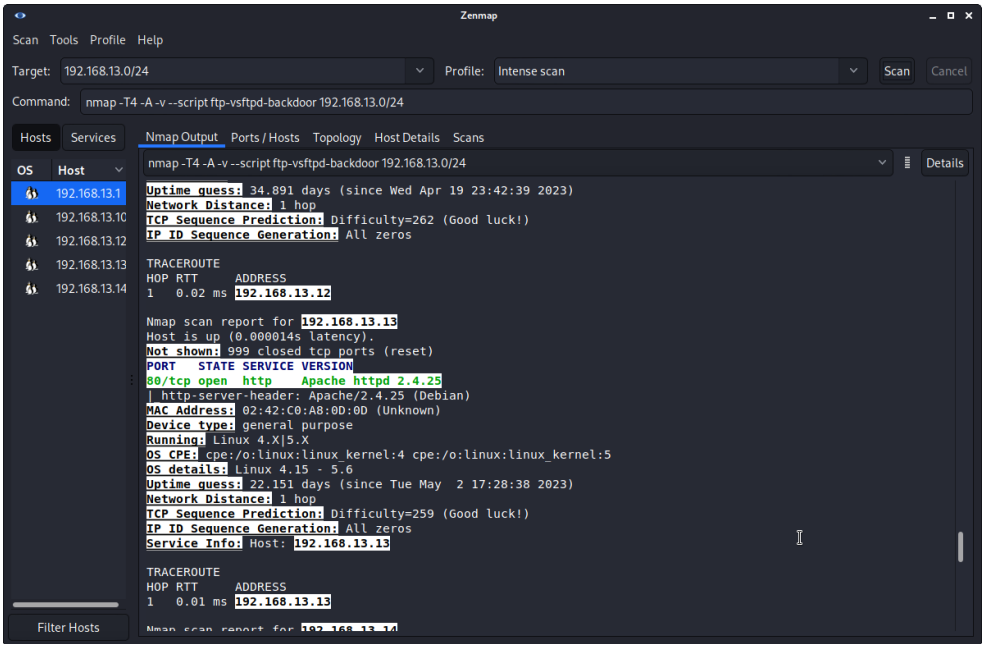
Vulnerability Findings

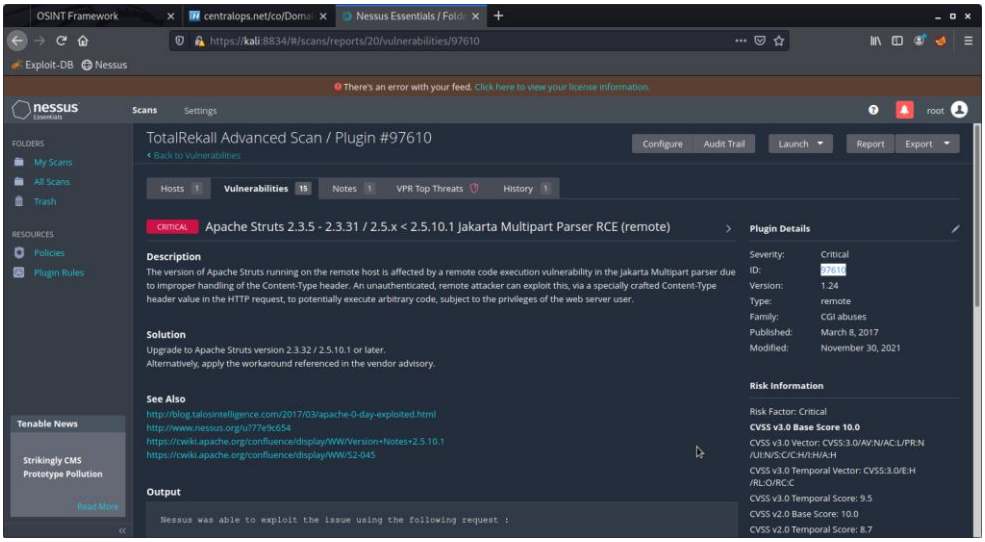
Vulnerability 1	Findings
Title	Server info vulnerable to Nmap scans
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Medium
Description	Using Nmap/Zenmap, STX was able to find OS and service information on two hosts, both the web app host at 192.168.14.35 and at 192.168.14.1.
Images	N/A
Affected Hosts	192.168.14.35
Remediation	• Protect public-facing pages from revealing server information. • Remove or password-protect asset pages and other sensitive locations on active servers.

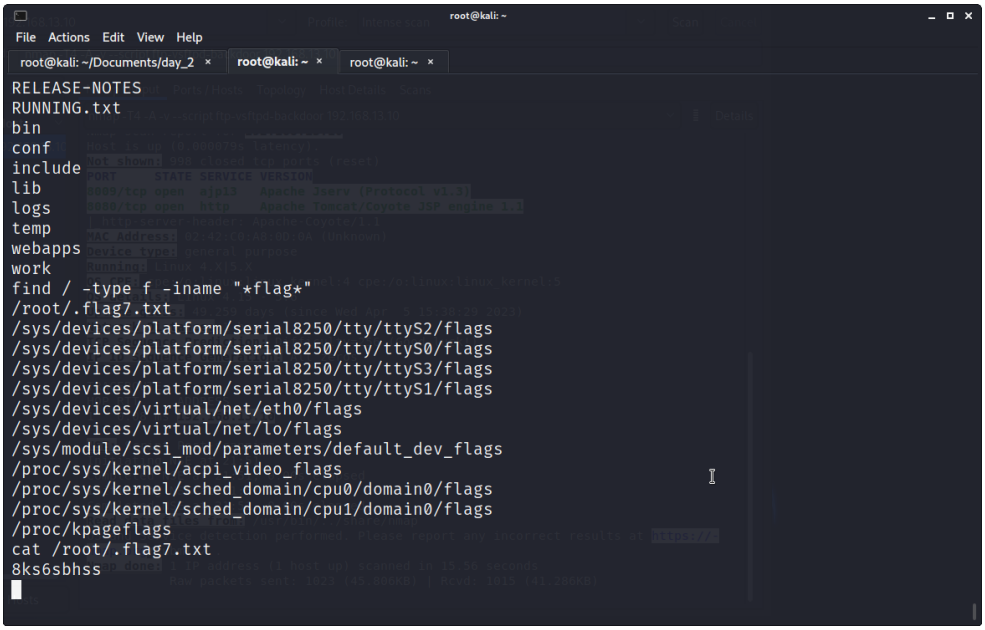
Vulnerability 2	Findings
Title	Network scan shows hosts with open ports
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Medium
Description	Using “intense” settings with Zenmap, STX was able to detect open, unfiltered ports on both network machines. The .35 machine, which hosts Rekall’s web application, showed HTTP open at P80 and MySQL open at P3306. The .1 machine showed vnc open at P5901 and X11 open at P6001, though access was denied on this last port. Attackers would use this port and service information, not to mention the fact that these machines are simply reporting these ports as open rather than filtered, to try known exploits until they are able to find one that returns a shell on the machine.
Images	
Affected Hosts	192.168.14.35; 192.168.14.1
Remediation	● Use ufw or firewalld on web app service to filter scans and prevent open ports and services from appearing. ● Ensure that open ports are open only to known machines on the local network.

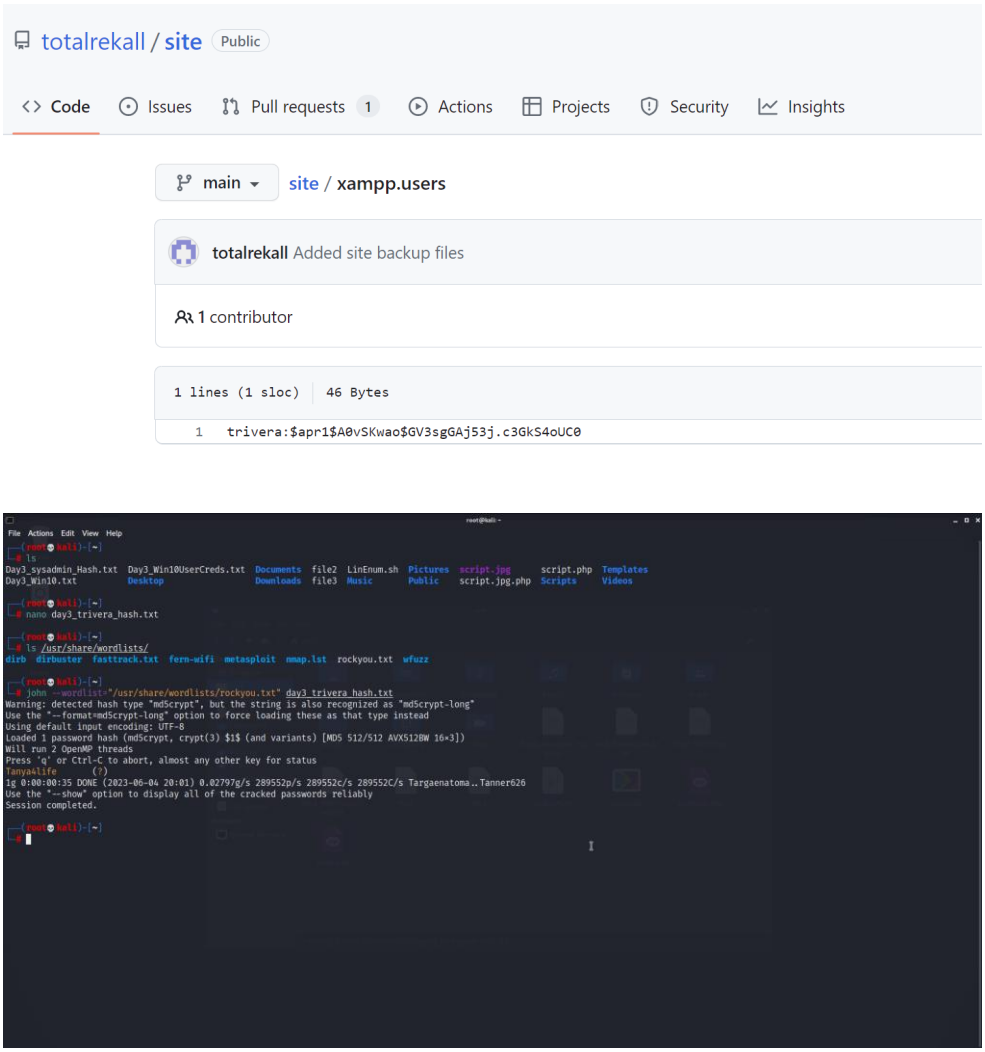
Vulnerability 3	Findings
Title	Subdomain search shows hidden .txt file
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	High
Description	Using Google search common operands such as *site:* and *filetype:,* STX was able to locate a hidden document on Rekall's server machine, *robots.txt.* This file contains sensitive info that an attacker would use to gain additional access to company machines.
Images	
Affected Hosts	192.168.14.35
Remediation	• Remove or password-protect any non-HTML file types that can be located with "Google dorking" search methods.

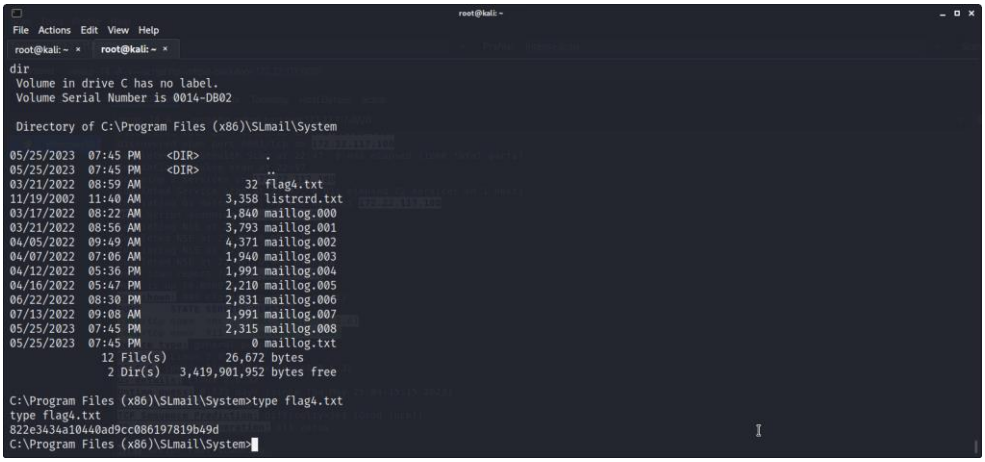
Vulnerability 4	Findings
Title	Web application vulnerable to XSS attacks
Type (Web app / Linux OS / WIndows OS)	Web app
Risk Rating	High
Description	The Rekall web application has several text fields for user interaction, including login fields for both users and administrators. None of the text entry fields use input validation methods and so STX was able to return information about the sever environment and eventually obtain a Linux shell by using a variety of XSS attacks that employed simple JavaScript commands.
Images	
Affected Hosts	192.168.14.35
Remediation	• Use input validation methods to prevent scripts/commands from being entered in text or login fields. • Sanitize input in text and login fields before providing a response.

Vulnerability 5	Findings
Title	Network with 5 Linux machines with open ports and vulnerabilities found
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Critical
Description	A scan of another IP address range belonging to Rekall uncovered a network of 5 Linux machines, all with a few open ports and vulnerabilities. Using another “intense” Zenmap scan, STX found exploitable weaknesses in each machine and was able to use common techniques (Metasploit modules) to gain access to sensitive information and a Linux shell.
Images	
Affected Hosts	192.168.13.0/24 (5 hosts up)
Remediation	• As mentioned above, Rekall can use both network and client firewalls to protect these ports, marking them as filtered rather than open or closed. • Ensure that traffic in/out from open ports is being monitored by an IDS/IPS for suspicious content and activity.

Vulnerability 6	Findings
Title	Nessus scan of machine in the Linux network returned critical vulnerability
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Critical
Description	STX used Nessus software to scan each Linux machine in the 192.168.13.0/24 IP range and found a documented critical vulnerability on the .12 server, in this case a CGI abuse. The vulnerable port is 8080 and we found the Metasploit module *Apache Struts Jakarta Multiport Parser OGNL Injection* successfully provided a shell with sudo privileges. It is important to correct this vulnerability immediately since an attacker would use the same information we've found during our pentest to exploit the machine.
Images	
Affected Hosts	192.168.13.12
Remediation	• Update Apache Struts software, which corrects the exploit

Vulnerability 7	Findings
Title	Admin shell achieved at .12 machine with Metasploit module
Type (Web app / Linux OS / WIndows OS)	Linux OS
Risk Rating	Critical
Description	Using information from the scans noted above, STX was able to gain a root shell on one of the Linux machines in this network. We tried many Linux HTTP exploits without success, including *cpi_tararchive_upload* and *vmware_view_planner_4_6_uploading_rce,* but we were able to gain access with *tomcat_jsp_upload_bypass.* This exploit returned a root shell to the machine, effectively placing it completely under our control to search for information, establish persistence, and exploit other network machines.
Images	
Affected Hosts	192.168.13.12
Remediation	• This exploit was patched in Apache_Tomcat_7.0.81. Upgrade immediately.

Vulnerability 8	Findings
Title	User credentials *user:hash* posted in unprotected Github repository
Type (Web app / Linux OS / WIndows OS)	Windows 10
Risk Rating	Critical
Description	STX was able to find sensitive admin information by searching the Rekall Github repository. There are a number of unprotected sensitive files held here, critically *xampp.users* which contains the username and password hash of administrator *trivera.* We were then able to easily crack the password hash using FOSS software (John the Ripper) and so obtained the admin login credentials *trivera:Tanya4life.*
Images	
Affected Hosts	172.22.117.110
Remediation	• Remove *xampp.users* and other sensitive info *(old-site* directory, *assets* directory, *robots.txt)* from the Github repository. • Require user trivera to migrate account to a new username and strong password.

Vulnerability 9	Findings
Title	9. Win10 machine vulnerable to repeatable pop3 exploit
Type (Web app / Linux OS / WIndows OS)	Windows 10
Risk Rating	Critical
Description	Using credentials obtained in the method described above, STX was able to gain a Meterpreter shell on the Win 10 machine at 172.22.117.10. The Metasploit module used was */windows/pop3/seattlelab_pars.* This exploit numerous times to bring up Meterpreter. From the Meterpreter prompt, we were able to determine our user privileges, migrate to different processes, as well as drop into a SYSTEM shell in Windows as needed. This exploit effectively gave us complete and persistent control of the Win10 machine.
Images	
Affected Hosts	172.22.117.10
Remediation	• This SLMail buffer overflow exploit is effective against all versions of Windows but is only successful with current user credentials. • Use client-side antivirus software, which will detect evidence of the exploit.

Vulnerability 10	Findings
Title	Win10 machine vulnerable to Mimikatz/kiwi credential snooping
Type (Web app / Linux OS / WIndows OS)	Windows 10
Risk Rating	Critical
Description	Obtaining access once again to Win10, STX used the Linux implementation of Mimikatz to gain another set of credentials. From a Meterpreter shell on Win10, we loaded the kiwi and used the commands *kiwi_cmd lsadump::sam* and *kiwi_cmd lsadump:cache* to return additional usernames and password hashes. ADMBob's password hash was broken with John in the same method described above. We suspected that this new set of credentials would lead us to gain access to the WinDC01, or Rekall's domain controller.
Images	
Affected Hosts	172.22.117.20
Remediation	• Disable debugging in the Win10 registry unless it is specifically necessary. • Avoid using stored passwords and disable credential caching.

Vulnerability 11	Findings
Title	WinDC01 machine vulnerable to lateral movement exploits
Type (Web app / Linux OS / WIndows OS)	Windows 10
Risk Rating	Critical
Description	STX was able to use a lateral movement exploit through a Meterpreter shell on the Win10 machine to gain SYSTEM-level access to Rekall's domain controller, WinDC01. Once we established high-level privileges on WinDC01, we were able to employ similar techniques to those used previously on Win10 to find *username:password hash* information about all the machines in Rekall's Windows network.
Images	N/A
Affected Hosts	172.22.117.20; 172.22.117.10
Remediation	• Ensure all Windows network computers are using updated antivirus software, which will detect and stop lateral movement attacks. • Refer to Microsoft's document detailing first steps for securing Active Directory, which would prevent common lateral movement attacks on a domain controller.