Hani Mebar, **Assignment 2**
Haaga-Helia University of Applied Sciences
ICT4HM103-3004
10/11/2022

Assignment 2

Questions

x) Read and summarize (with some bullet points)

- € Schneier 2015: Applied Cryptography: Chapter 2 - Protocol Building Blocks: subchapters "2.3 One-way Fuctions" and "2.4 One-Way Hash Functions".
- Karvinen 2022: Cracking Passwords with Hashcat
- Karvinen 2020: Command Line Basics Revisited
- Voluntary bonus article: € Santos et al 2017: Security Penetration Testing - The Art of Hacking Series LiveLessons: Lesson 6: Hacking User Credentials (8 videos, about 30 min)

  a) Install hashcat and test that it works.
  b) Crack this hash: 21232f297a57a5a743894a0e4a801fc3
  c) Crack this Windows NTLM hash: f2477a144dff4f216ab81f2ac3e3207d
  d) Try cracking this hash and comment on your hash rate $2y$18$axMtQ4N8j/NQVItQJed9uORfsUK667RAWfycwFMtDBD6zAo1Se2eu . This subtask d does not require actually cracking the hash, just trying it and commenting on the hash rate.
  e) Voluntary bonus: make hashcat work with your display adapter (GPU).
  f) Voluntary bonus: create some hashes of your own, then crack them with hashcat.
  g) Voluntary bonus hash. John the Ripper aka 'john' might also work here.

```
$ sudo grep elmik9 /etc/passwd /etc/shadow
/etc/passwd:elmik9:x:1003:1003:Elmeri "9" Elmik,,,:/home/elmik9:/bin/bash
/etc/shadow:elmik9:$1$xpRkwrhq$aXdu7HQirUmuTZW2m8OXs.:18401:0:99999:7:::
```

Answers

(X) Schneier Bullet points:

- **One-way functions** in cryptography means it's something that is easy to create but practically impossible to unmake. Like, seeing something is easy but to unsee it is not possible (forgetting doesn't count!).
- Because they can't be reversed, then it's practically impossible to confirm or deny their existence.
- So if one-way functions are a super cryptographic method of security but can't be cracked, how can we use them and what is their point if we can't even undo them when we need to? If there is a trapdoor one-way function, then they are very useful. Eg having 'intruction' manual, or key to the trap door.

Hani Mebar, **Assignment 2**
Haaga-Helia University of Applied Sciences
ICT4HM103-3004
10/11/2022

- One-way hash functions has many names: compression function, contraction function, message digest, fingerprint, cryptographic checksum, message integrity check (MIC), and manipulation detection code (MDC).
- A **hash-function** takes a variable-length input string (called a pre-image) and converts it to a fixed-length (generally smaller) output string (called a hash value).
- It produces a value that indicates whether a candidate fingerprint is likely to be the same as the real fingerprint.
- **A one-way hash function** is a hash function that works in one direction: It is easy to compute a hash value from pre-image, but it is hard to generate a pre-image that hashes to a particular value. So, similar to the one-way function but in terms of digital fingerprints from pre-images.
- **Collision-free** one-way hash functions are single pre-image hash values. Ie there are no duplicates.
- Hash-functions are public but they are computationally unfeasible to find pre-images that has to that value.
- **Message Authentication Code (MAC)** *(are these the same that we see in the back of the modems? MAC address? or in our data setting on mobile phones?)* also known as data authentication codes (DAC) is a one-way hash function with a secret key. The hash value is both function and key, so only someone with the key can verify the hash value.

Cracking Passwords with Hashcat

- Systems store passwords as hashes, eg a series of numbers of letters that look like a cat jumped on your keyboard.
- Hashcat is a software that matches hashes with leaked passwords. The process of cracking hashes is illegal unless you have been granted permission.
- You need a list or dictionary of the leaked passwords which can with be purchased or can be taken from approved lists like the one from rockyou in github.
- Hashes have different algorithm types. MD2, MD4, MD5, SHA-1, SHA-2, NTLM, LANMAN etc...
- After following the steps for hashcat, we should check how it went using the '- -show' command

Command line basics revisited

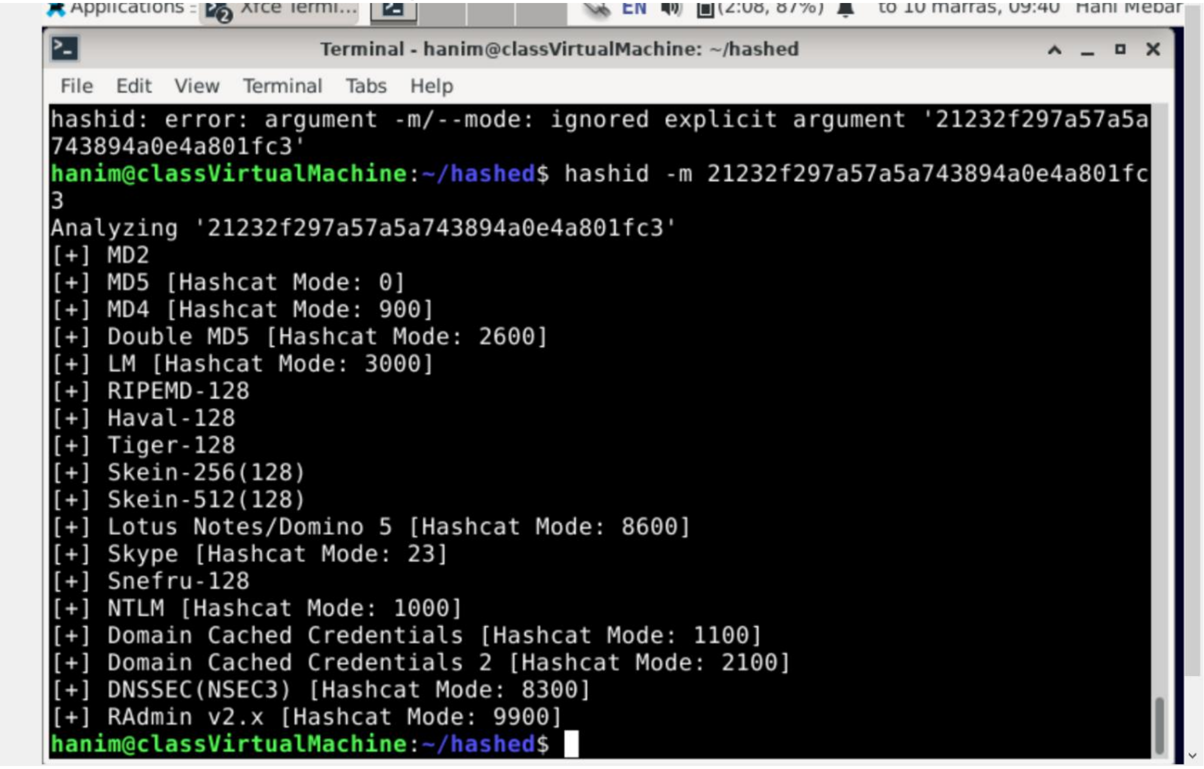
- Command line in Linux and BSD (Berkely Software Distribution) has been around for a long time; since 1977 (so, after Fortran)
- We are always in some directory.
- Played around and did the same as what Tero showed in the assignment.

Bonus on livelessons, the art of hacking.

- Repeated most of what was already shared by Tero, but I also learned about John the Ripper (which is like Hashcat but splits between LM and NT) and takes longer than hashcat.
- Main notes I took were on how to make passwords stronger
- Split between Consumer and Organization passwords. Hashing alg is not enough, you need to also salt, 2-factor auth, randomness, and have strong passwords everywhere. (are the suggested passwords good and safe?)
- Breaches will still happen, but with the above, especially 2-factor auth, it can still prevent attackers from getting your data.

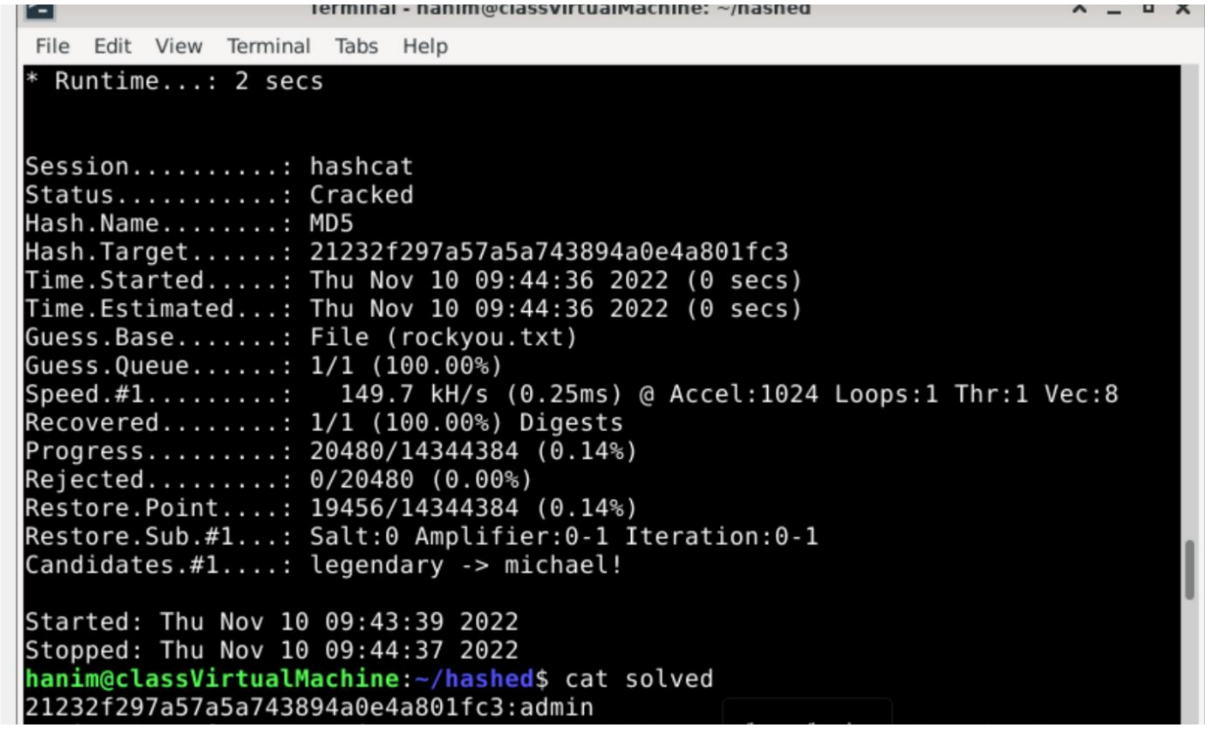
Main part of assignment

a) Installed hashid and used the example from the lesson to test it and it worked,
b) Crack this hash: 21232f297a57a5a743894a0e4a801fc3.
    1. Checked what algorithm is used, tried with MD5 first.



```
hashid: error: argument -m/--mode: ignored explicit argument '21232f297a57a5a
743894a0e4a801fc3'
hanim@classVirtualMachine:~/hashed$ hashid -m 21232f297a57a5a743894a0e4a801fc
3
Analyzing '21232f297a57a5a743894a0e4a801fc3'
[+] MD2
[+] MD5 [Hashcat Mode: 0]
[+] MD4 [Hashcat Mode: 900]
[+] Double MD5 [Hashcat Mode: 2600]
[+] LM [Hashcat Mode: 3000]
[+] RIPEMD-128
[+] Haval-128
[+] Tiger-128
[+] Skein-256(128)
[+] Skein-512(128)
[+] Lotus Notes/Domino 5 [Hashcat Mode: 8600]
[+] Skype [Hashcat Mode: 23]
[+] Snefru-128
[+] NTLM [Hashcat Mode: 1000]
[+] Domain Cached Credentials [Hashcat Mode: 1100]
[+] Domain Cached Credentials 2 [Hashcat Mode: 2100]
[+] DNSSEC(NSEC3) [Hashcat Mode: 8300]
[+] RAdmin v2.x [Hashcat Mode: 9900]
hanim@classVirtualMachine:~/hashed$
```

    2. Applied the command for hashcat to work, and it ended up cracking it.

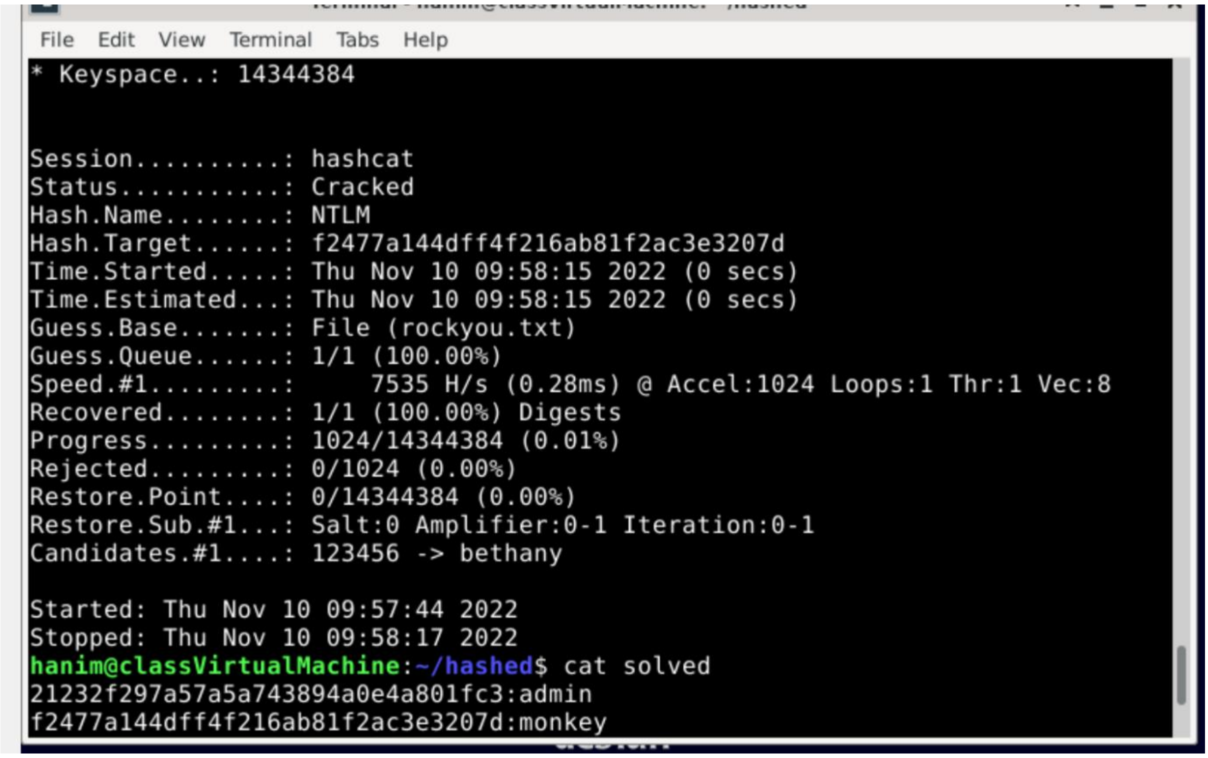

Terminal - hanim@classVirtualMachine: ~/hashed
File Edit View Terminal Tabs Help

```
* Runtime...: 2 secs


Session..........: hashcat
Status...........: Cracked
Hash.Name........: MD5
Hash.Target......: 21232f297a57a5a743894a0e4a801fc3
Time.Started.....: Thu Nov 10 09:44:36 2022 (0 secs)
Time.Estimated...: Thu Nov 10 09:44:36 2022 (0 secs)
Guess.Base.......: File (rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:   149.7 kH/s (0.25ms) @ Accel:1024 Loops:1 Thr:1 Vec:8
Recovered........: 1/1 (100.00%) Digests
Progress.........: 20480/14344384 (0.14%)
Rejected.........: 0/20480 (0.00%)
Restore.Point....: 19456/14344384 (0.14%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidates.#1....: legendary -> michael!

Started: Thu Nov 10 09:43:39 2022
Stopped: Thu Nov 10 09:44:37 2022
hanim@classVirtualMachine:~/hashed$ cat solved
21232f297a57a5a743894a0e4a801fc3:admin
```

3. Applied cat solved command and password revealed was **<u>‘admin’</u>**

c) Crack this Windows NTLM hash: f2477a144dff4f216ab81f2ac3e3207d

1. Since this is NTLM then the hashcat mode is number 1000
2. So I enter the command hashcat -m <u>1000</u> ‘####hascode’ rockyou.txt -o solved.

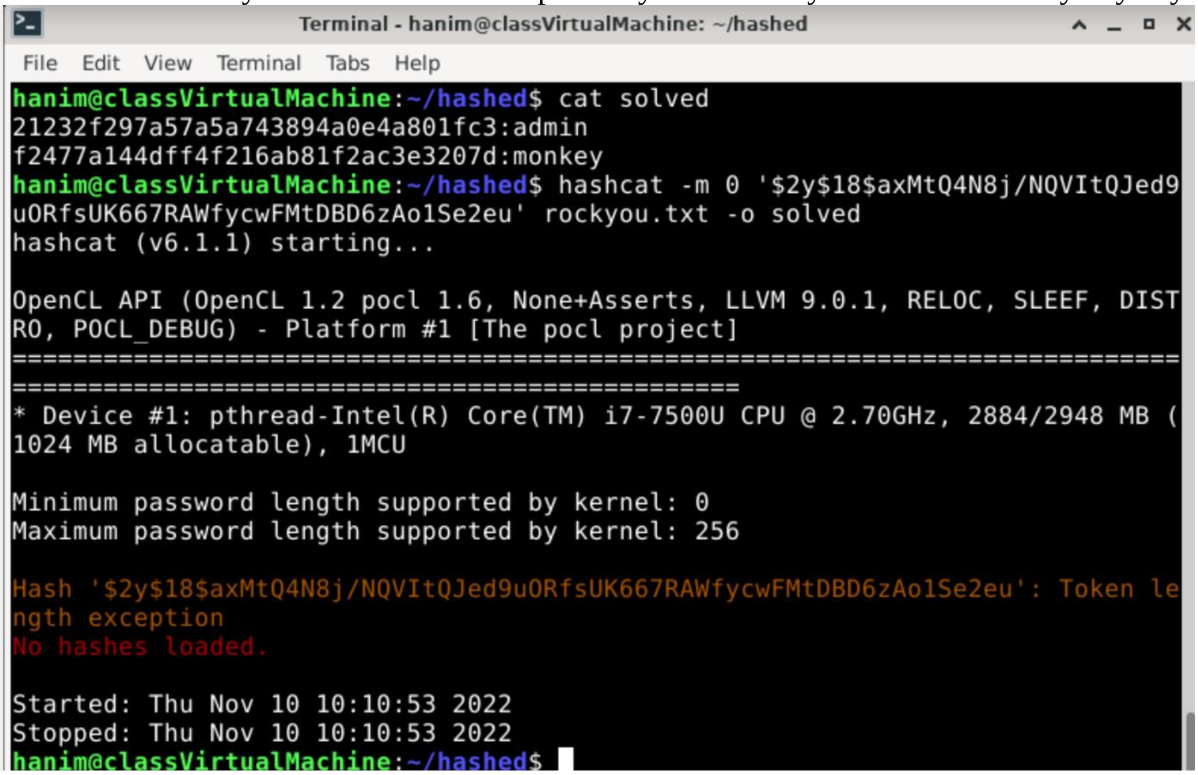


```
* Keyspace..: 14344384


Session..........: hashcat
Status...........: Cracked
Hash.Name........: NTLM
Hash.Target......: f2477a144dff4f216ab81f2ac3e3207d
Time.Started.....: Thu Nov 10 09:58:15 2022 (0 secs)
Time.Estimated...: Thu Nov 10 09:58:15 2022 (0 secs)
Guess.Base.......: File (rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:     7535 H/s (0.28ms) @ Accel:1024 Loops:1 Thr:1 Vec:8
Recovered........: 1/1 (100.00%) Digests
Progress.........: 1024/14344384 (0.01%)
Rejected.........: 0/1024 (0.00%)
Restore.Point....: 0/14344384 (0.00%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidates.#1....: 123456 -> bethany

Started: Thu Nov 10 09:57:44 2022
Stopped: Thu Nov 10 09:58:17 2022
hanim@classVirtualMachine:~/hashed$ cat solved
21232f297a57a5a743894a0e4a801fc3:admin
f2477a144dff4f216ab81f2ac3e3207d:monkey
```

3. Took longer to solve but it was cracked. The password was **‘monkey’**

d) Try cracking the hash $2y$18$axMtQ4N8j/NQVItQJed9uORfsUK667RAWfycwFMtDBD6zAo1Se2eu

1. I am not sure what type of hash this is so I looked around.
2. What I got is that if a hash has a '$' then this is a delimiter between the salt and hash. This means it falls under BCRYPT or MD5-Crypt.
3. Since hashcat doesn't have either of these algorithmic keys then it will likely not solve it and will probably take infinity to solve. I will try anyway



```
hanim@classVirtualMachine:~/hashed$ cat solved
21232f297a57a5a743894a0e4a801fc3:admin
f2477a144dff4f216ab81f2ac3e3207d:monkey
hanim@classVirtualMachine:~/hashed$ hashcat -m 0 '$2y$18$axMtQ4N8j/NQVItQJed9
uORfsUK667RAWfycwFMtDBD6zAo1Se2eu' rockyou.txt -o solved
hashcat (v6.1.1) starting...

OpenCL API (OpenCL 1.2 pocl 1.6, None+Asserts, LLVM 9.0.1, RELOC, SLEEF, DIST
RO, POCL_DEBUG) - Platform #1 [The pocl project]
==============================================================================
=================================================
* Device #1: pthread-Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz, 2884/2948 MB (
1024 MB allocatable), 1MCU

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

Hash '$2y$18$axMtQ4N8j/NQVItQJed9uORfsUK667RAWfycwFMtDBD6zAo1Se2eu': Token le
ngth exception
No hashes loaded.

Started: Thu Nov 10 10:10:53 2022
Stopped: Thu Nov 10 10:10:53 2022
hanim@classVirtualMachine:~/hashed$
```

4. So as expected it didn't work but it also didn't take long; using MD5 hashcat immediated informed that the length of the hash was an exception.

e) I did not try the GPU bonus.

f) I tried this one but I decided I want to find out my own hask key frp gmail or LinkedIn or Yahoo Mail etc.. but I could not figure out how to get my hash number for any of those services (I suppose that is a good thing).

g) Didn't try john the ripper YET.