Computer Science 328 -01

Lab Assignment 7

Due Date: Sunday, March 17, 2024 (before midnight)

Objective: Experimenting basic Ethical Hacking techniques

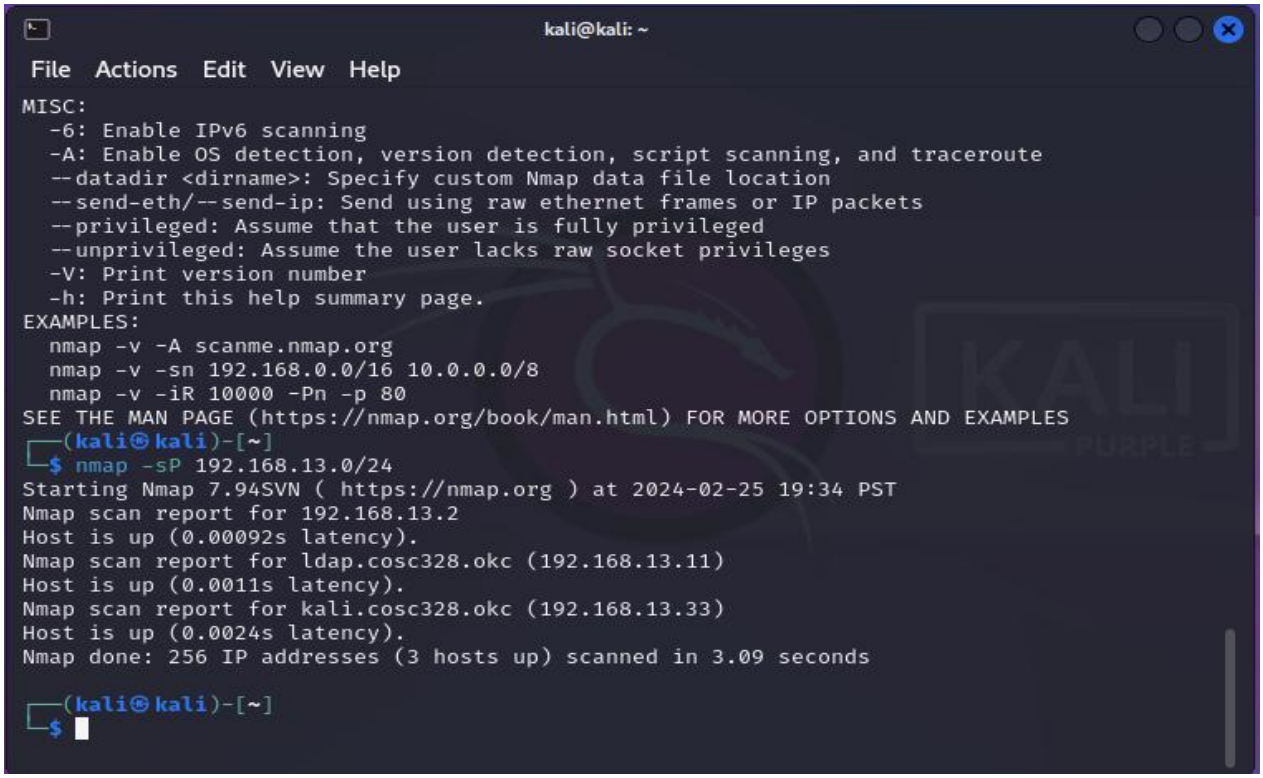
- *Please note that an Ubuntu 20.04 virtual machine is required in order to see the hacking results for a few hacking exercises in this lab. If you don't have one, you can download it from a Google drive here:*

  https://drive.google.com/file/d/1vMXZtiAUYdRgJZYV4C7_-skgStquc40o/view?usp=sharing

1: Running *Nmap* from Kali VM

- **Nmap** is one of most commonly used hacking tool to explore open ports of all computers available in a network very quick and easy. Please refer to the following Youtube video to learn what you can do with **Nmap** tool: **https://www.youtube.com/watch?v=4t4kBkMsDbQ&list=PLIhvC56v63IIJZRa3lzK6IeBQOH_VFjUQ&index=4**

- Then try these ***nmap*** commands with Kali VM. Click on Applications icon from Kali VM => 01-Information Gathering => nmap => a terminal will pop up as shown below:



Enter the command "***nmap –sP 192.168.n.0/24***" at the terminal (where **n** is your NAT subnet number, and 13 as shown in the image above was the NAT subnet on your instructor's desktop) to find out which computers are up and running in your NAT subnet: 192.168.n.0/24 for all your VMs. Take a screen shot and insert your screen shot below:

```
┌──(kali㉿kali)-[~]
└─$ nmap -sP 192.168.58.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-12 14:49 EST
Nmap scan report for 192.168.58.2
Host is up (0.00045s latency).
Nmap scan report for vhost1.cosc.okc (192.168.58.11)
Host is up (0.00063s latency).
Nmap scan report for yhuVM.cosc.okc (192.168.58.33)
Host is up (0.000080s latency).
Nmap done: 256 IP addresses (3 hosts up) scanned in 2.63 seconds
```

- Next try the command "***nmap –sT –p 80,443 192.168.n.0/24***" to find out all computers with port 80 and 443 open in the subnet, take a screen shot and insert your screen shot below:

```
└─$ nmap -sT -p 80,443 192.168.58.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-12 14:50 EST
Nmap scan report for 192.168.58.2
Host is up (0.00044s latency).

PORT    STATE  SERVICE
80/tcp  closed http
443/tcp closed https

Nmap scan report for vhost1.cosc.okc (192.168.58.11)
Host is up (0.00073s latency).

PORT    STATE  SERVICE
80/tcp  open   http
443/tcp closed https

Nmap scan report for yhuVM.cosc.okc (192.168.58.33)
Host is up (0.00033s latency).

PORT    STATE  SERVICE
80/tcp  closed http
443/tcp closed https

Nmap done: 256 IP addresses (3 hosts up) scanned in 2.36 seconds
```

2: Setting up Telnet server to be hacked using Wireshark

- Start Ubuntu VM2 (Ubuntu 20.04), and open a terminal for setting up a Telnet server.
  - Update repository list, then upgrade and install Telnet server:

    ```
    $ sudo apt update
    $ sudo apt upgrade
    $ sudo apt install telnetd
    ```

  - Next verify Telnet status is active as shown below:

    ```
    $ sudo systemctl status inetd
    ```

```
cs213@ldap:~$ sudo systemctl status inetd
[sudo] password for cs213:
● inetd.service - Internet superserver
     Loaded: loaded (/lib/systemd/system/inetd.service; enabled; vendor preset:
     Active: active (running) since Sat 2024-01-27 19:05:21 PST; 2min 46s ago
       Docs: man:inetd(8)
   Main PID: 1191 (inetd)
      Tasks: 1 (limit: 2392)
     Memory: 536.0K
     CGroup: /system.slice/inetd.service
             └─1191 /usr/sbin/inetd

Jan 27 19:05:21 ldap.cosc328.okc systemd[1]: Starting Internet superserver...
Jan 27 19:05:21 ldap.cosc328.okc systemd[1]: Started Internet superserver.
```

- If any firewall is running on your system (try **"sudo ufw status"** command to find out (i.e. active)), then we need to enable the necessary port (23). Otherwise we can ignore the following steps.

```
$ sudo ufw allow 23
$ sudo ufw reload
$ sudo ufw enable
```

- Since Telnet client comes as a pre-installed package, we can switch to Ubuntu VM1 and try to telnet to Ubuntu VM2 with a terminal immediately:

```
$ telnet 192.168.13.22   //with username: cs213 and password: letmein
```

```
cs213@ldap:~$ telnet 192.168.13.22
Trying 192.168.13.22...
Connected to 192.168.13.22.
Escape character is '^]'.
Ubuntu 20.04 LTS
vm2.cosc328.okc login: cs213
Password:
Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.15.0-91-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

303 updates can be installed immediately.
1 of these updates is a security update.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Tue Jan 18 18:35:56 PST 2022 from 192.168.25.132 on pts/1
cs213@vm2:~$
```

- Take a screen shot of your VM1's terminal to show the telnet connect to your VM2, and insert the screen below:

```
herrycooly@cosc328:~$ telnet 192.168.58.11
Trying 192.168.58.11...
Connected to 192.168.58.11.
Escape character is '^]'.
Ubuntu 20.04.6 LTS
cosc328.okc login: herrycooly
Password:
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-97-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

2 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Your Hardware Enablement Stack (HWE) is supported until April 2025.
*** System restart required ***
Last login: Thu Feb 29 23:00:07 EST 2024 from 192.168.58.22 on pts/1
herrycooly@cosc328:~$
```

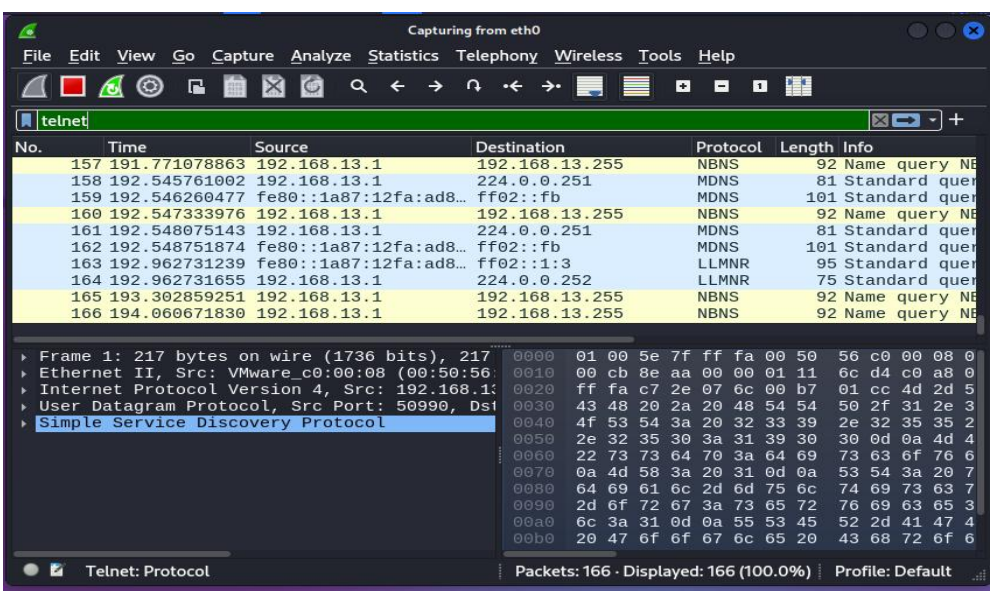
PS:

```
herrycooly@cosc328:~$ telnet 192.168.58.22
Trying 192.168.58.22...
telnet: Unable to connect to remote host: No route to host
herrycooly@cosc328:~$
```

VM2 can telnet to VM1 but not the other way round for some reason(even after followed the lab instruction for opening ports).

- Now switch to Kali VM, and open the Wireshark application.

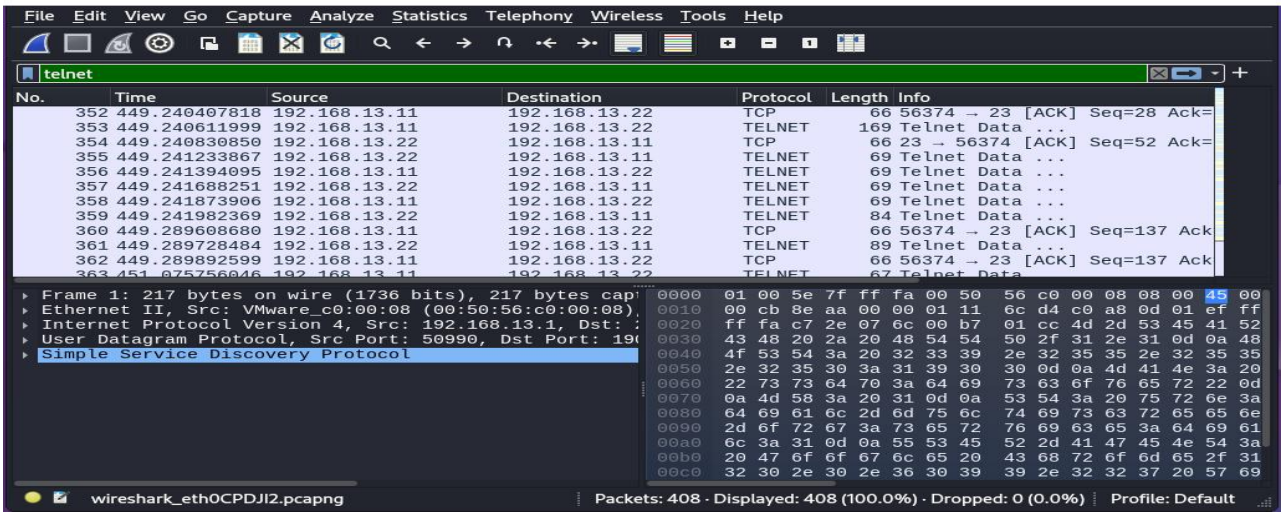
  - First start capturing the interface **eth0** (connecting to NAT subnet) with Wireshark, specify telnet as filter as shown below:



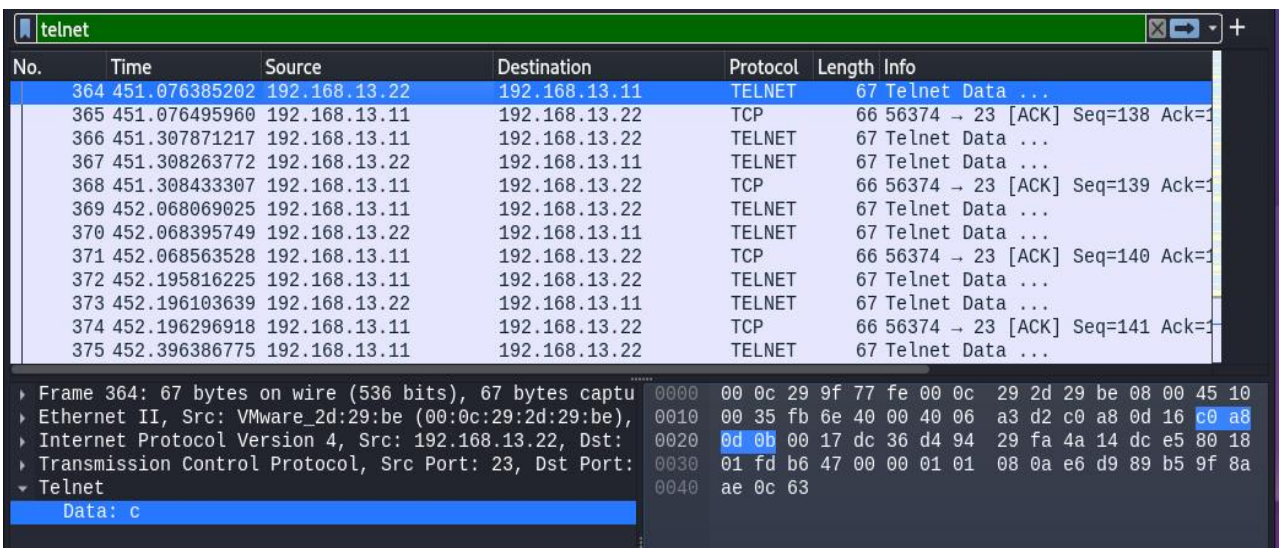
- Next make a telnet connection from Ubuntu VM1 (192.168.13.11) to VM2 (192.168.13.22) machine using the command:

  $ telnet 192.168.13.22 //with username: **cs213** and password: **letmein**

- Then switch back to your Kali VM machine, and stop capturing packets with Wireshark. Take a look at the Wireshark capturing screen:



Let's look at the Telnet Data (starting from frame #364 in this example) for the username:



telnet

No.	Time	Source	Destination	Protocol	Length	Info
364	451.076385202	192.168.13.22	192.168.13.11	TELNET	67	Telnet Data ...
365	451.076495960	192.168.13.11	192.168.13.22	TCP	66	56374 → 23 [ACK] Seq=138 Ack=1
366	451.307871217	192.168.13.11	192.168.13.22	TELNET	67	Telnet Data ...
367	451.308263772	192.168.13.22	192.168.13.11	TELNET	67	Telnet Data ...
368	451.308433307	192.168.13.11	192.168.13.22	TCP	66	56374 → 23 [ACK] Seq=139 Ack=1
369	452.068069025	192.168.13.11	192.168.13.22	TELNET	67	Telnet Data ...
370	452.068395749	192.168.13.22	192.168.13.11	TELNET	67	Telnet Data ...
371	452.068563528	192.168.13.11	192.168.13.22	TCP	66	56374 → 23 [ACK] Seq=140 Ack=1
372	452.195816225	192.168.13.11	192.168.13.22	TELNET	67	Telnet Data ...
373	452.196103639	192.168.13.22	192.168.13.11	TELNET	67	Telnet Data ...
374	452.196296918	192.168.13.11	192.168.13.22	TCP	66	56374 → 23 [ACK] Seq=141 Ack=1
375	452.396386775	192.168.13.11	192.168.13.22	TELNET	67	Telnet Data ...

```
▸ Frame 366: 67 bytes on wire (536 bits), 67 bytes captu
▸ Ethernet II, Src: VMware_9f:77:fe (00:0c:29:9f:77:fe),
▸ Internet Protocol Version 4, Src: 192.168.13.11, Dst:
▸ Transmission Control Protocol, Src Port: 56374, Dst Po
▾ Telnet
    Data: s

0000  00 0c 29 2d 29 be 00 0c  29 9f 77 fe 08 00 45 10
0010  00 35 80 d3 40 00 40 06  1e 6e c0 a8 0d 0b c0 a8
0020  0d 16 dc 36 00 17 4a 14  dc e5 d4 94 29 fb 80 18
0030  01 f6 a5 65 00 00 01 01  08 0a 9f 8a ae f4 e6 d9
0040  89 b5 73
```

telnet

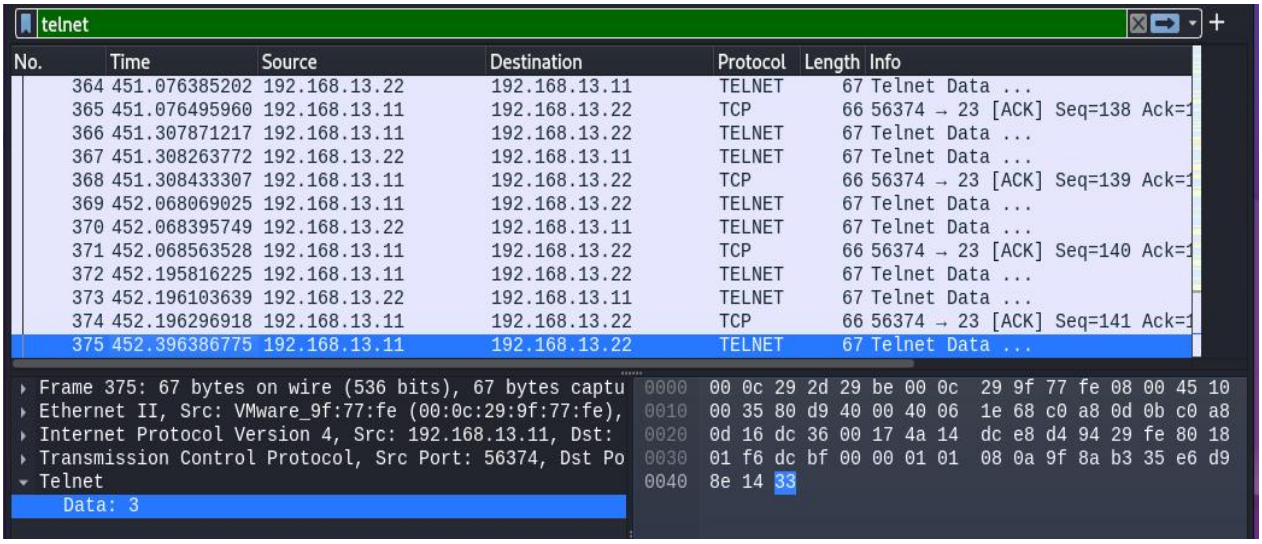
No.	Time	Source	Destination	Protocol	Length	Info
364	451.076385202	192.168.13.22	192.168.13.11	TELNET	67	Telnet Data ...
365	451.076495960	192.168.13.11	192.168.13.22	TCP	66	56374 → 23 [ACK] Seq=138 Ack=1
366	451.307871217	192.168.13.11	192.168.13.22	TELNET	67	Telnet Data ...
367	451.308263772	192.168.13.22	192.168.13.11	TELNET	67	Telnet Data ...
368	451.308433307	192.168.13.11	192.168.13.22	TCP	66	56374 → 23 [ACK] Seq=139 Ack=1
369	452.068069025	192.168.13.11	192.168.13.22	TELNET	67	Telnet Data ...
370	452.068395749	192.168.13.22	192.168.13.11	TELNET	67	Telnet Data ...
371	452.068563528	192.168.13.11	192.168.13.22	TCP	66	56374 → 23 [ACK] Seq=140 Ack=1
372	452.195816225	192.168.13.11	192.168.13.22	TELNET	67	Telnet Data ...
373	452.196103639	192.168.13.22	192.168.13.11	TELNET	67	Telnet Data ...
374	452.196296918	192.168.13.11	192.168.13.22	TCP	66	56374 → 23 [ACK] Seq=141 Ack=1
375	452.396386775	192.168.13.11	192.168.13.22	TELNET	67	Telnet Data ...

```
▸ Frame 369: 67 bytes on wire (536 bits), 67 bytes captu
▸ Ethernet II, Src: VMware_9f:77:fe (00:0c:29:9f:77:fe),
▸ Internet Protocol Version 4, Src: 192.168.13.11, Dst:
▸ Transmission Control Protocol, Src Port: 56374, Dst Po
▾ Telnet
    Data: 2

0000  00 0c 29 2d 29 be 00 0c  29 9f 77 fe 08 00 45 10
0010  00 35 80 d5 40 00 40 06  1e 6c c0 a8 0d 0b c0 a8
0020  0d 16 dc 36 00 17 4a 14  dc e6 d4 94 29 fc 80 18
0030  01 f6 e2 84 00 00 01 01  08 0a 9f 8a b1 ec e6 d9
0040  8a 9c 32
```

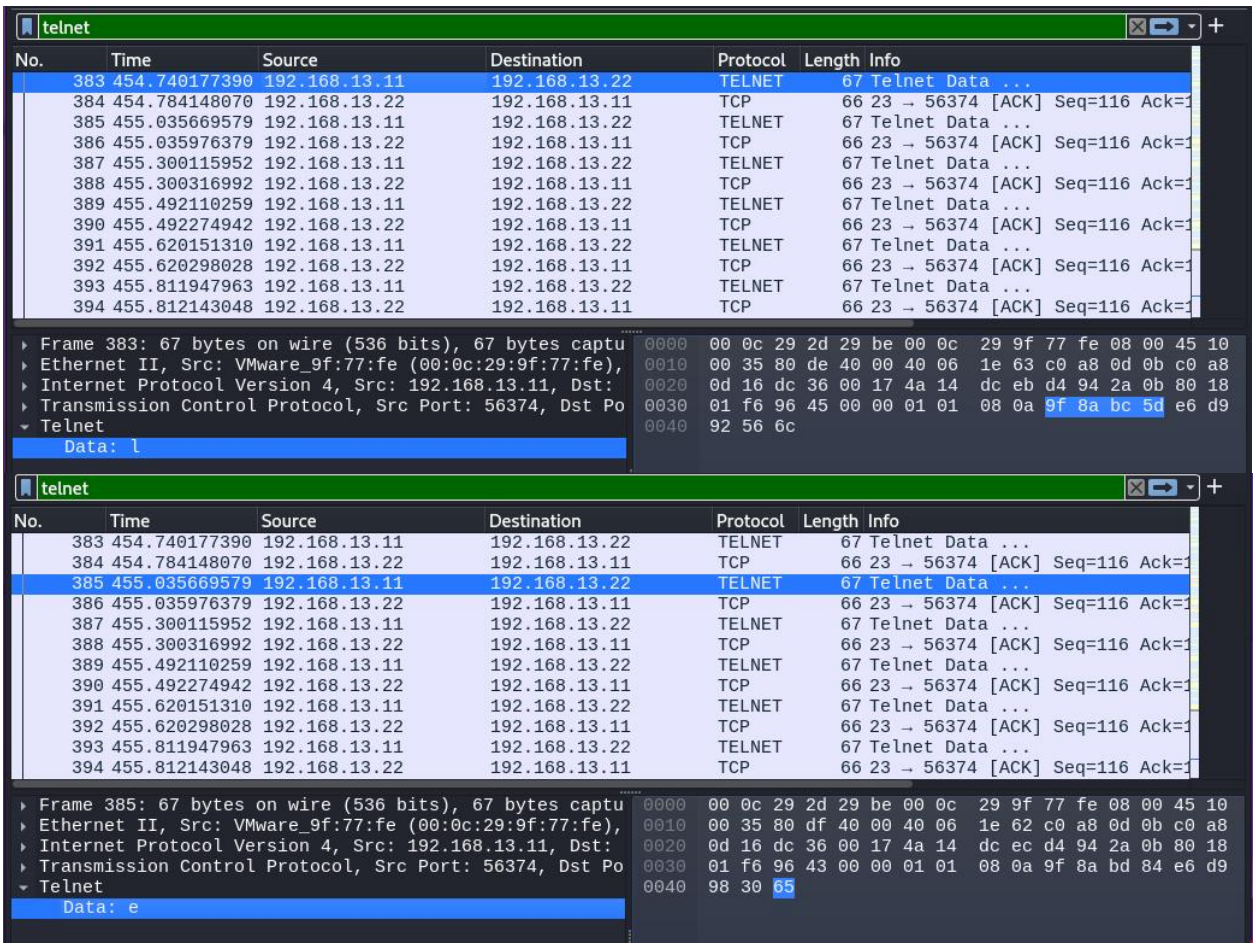
telnet

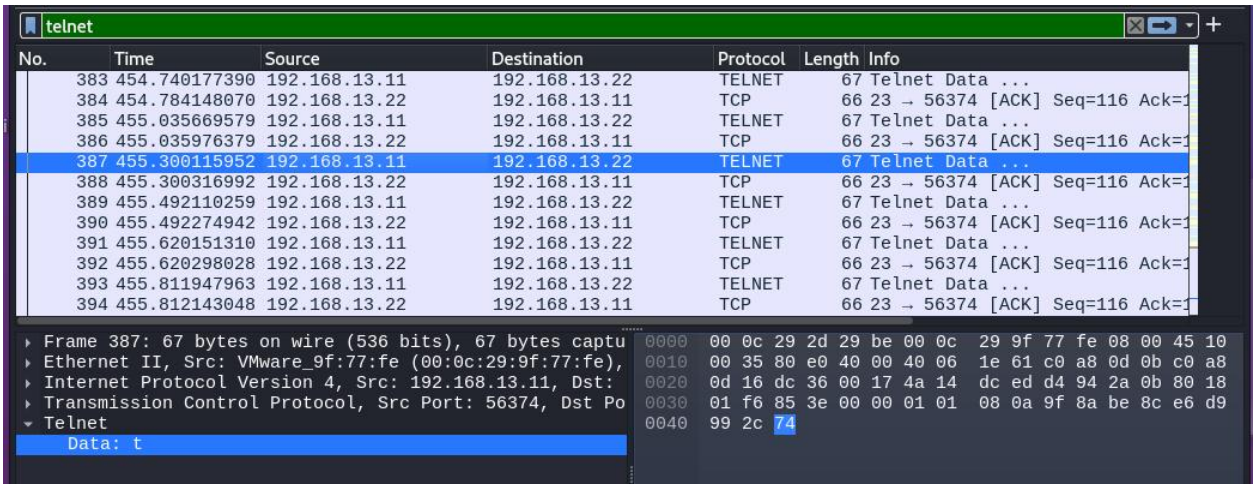
No.	Time	Source	Destination	Protocol	Length	Info
364	451.076385202	192.168.13.22	192.168.13.11	TELNET	67	Telnet Data ...
365	451.076495960	192.168.13.11	192.168.13.22	TCP	66	56374 → 23 [ACK] Seq=138 Ack=1
366	451.307871217	192.168.13.11	192.168.13.22	TELNET	67	Telnet Data ...
367	451.308263772	192.168.13.22	192.168.13.11	TELNET	67	Telnet Data ...
368	451.308433307	192.168.13.11	192.168.13.22	TCP	66	56374 → 23 [ACK] Seq=139 Ack=1
369	452.068069025	192.168.13.11	192.168.13.22	TELNET	67	Telnet Data ...
370	452.068395749	192.168.13.22	192.168.13.11	TELNET	67	Telnet Data ...
371	452.068563528	192.168.13.11	192.168.13.22	TCP	66	56374 → 23 [ACK] Seq=140 Ack=1
372	452.195816225	192.168.13.11	192.168.13.22	TELNET	67	Telnet Data ...
373	452.196103639	192.168.13.22	192.168.13.11	TELNET	67	Telnet Data ...
374	452.196296918	192.168.13.11	192.168.13.22	TCP	66	56374 → 23 [ACK] Seq=141 Ack=1
375	452.396386775	192.168.13.11	192.168.13.22	TELNET	67	Telnet Data ...

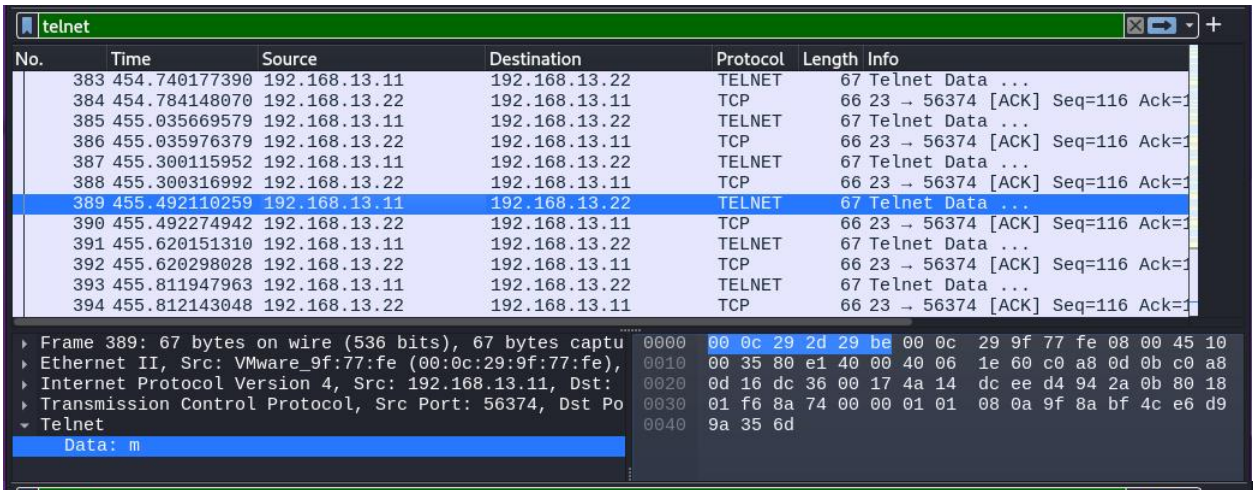
```
▸ Frame 372: 67 bytes on wire (536 bits), 67 bytes captu
▸ Ethernet II, Src: VMware_9f:77:fe (00:0c:29:9f:77:fe),
▸ Internet Protocol Version 4, Src: 192.168.13.11, Dst:
▸ Transmission Control Protocol, Src Port: 56374, Dst Po
▾ Telnet
    Data: 1

0000  00 0c 29 2d 29 be 00 0c  29 9f 77 fe 08 00 45 10
0010  00 35 80 d7 40 00 40 06  1e 6a c0 a8 0d 0b c0 a8
0020  0d 16 dc 36 00 17 4a 14  dc e7 d4 94 29 fd 80 18
0030  01 f6 e0 09 00 00 01 01  08 0a 9f 8a b2 6c e6 d9
0040  8d 95 31
```

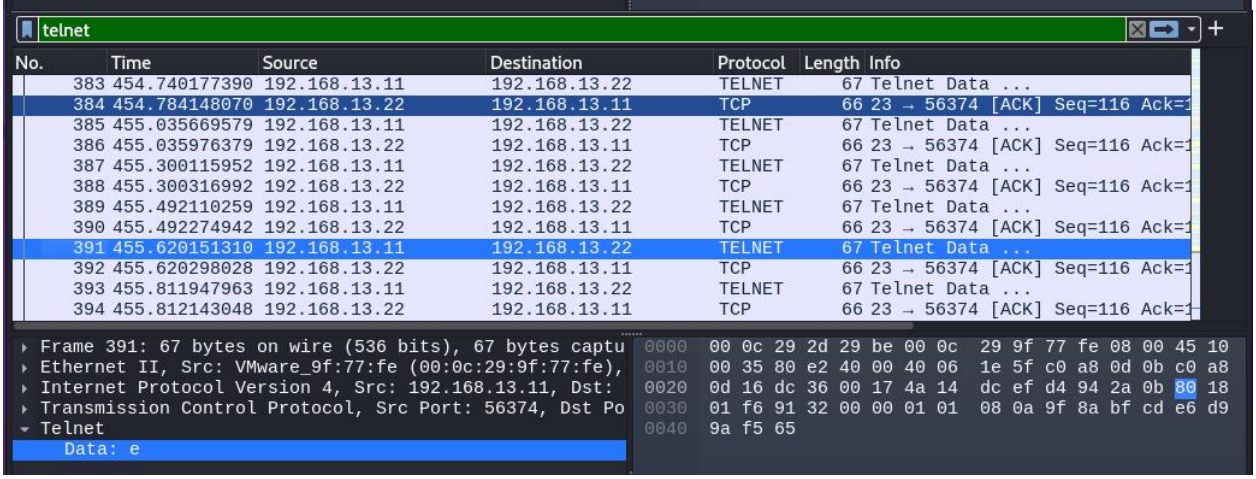


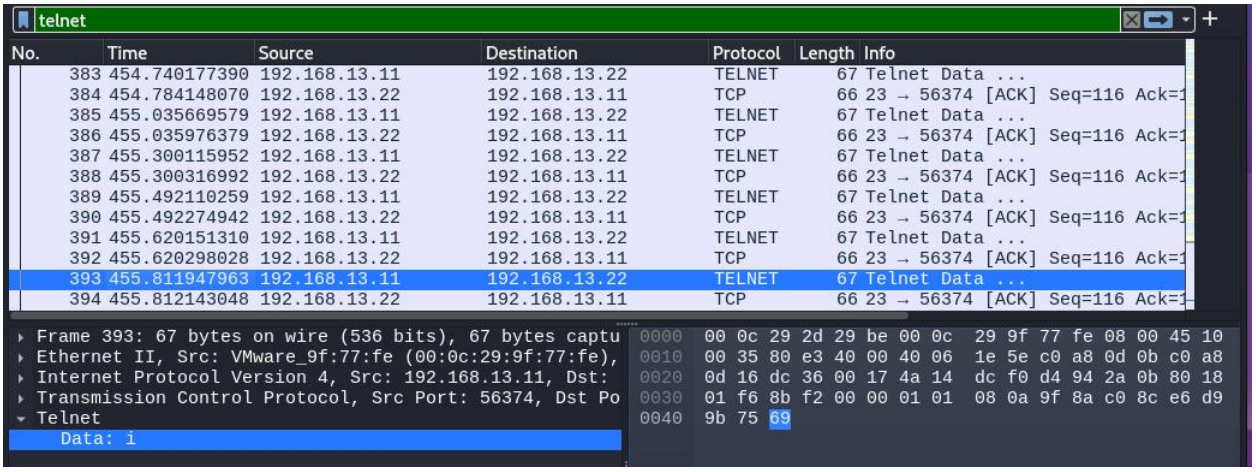
And now let's look at the Telnet Data (starting from frame #383 in this example) for the password:

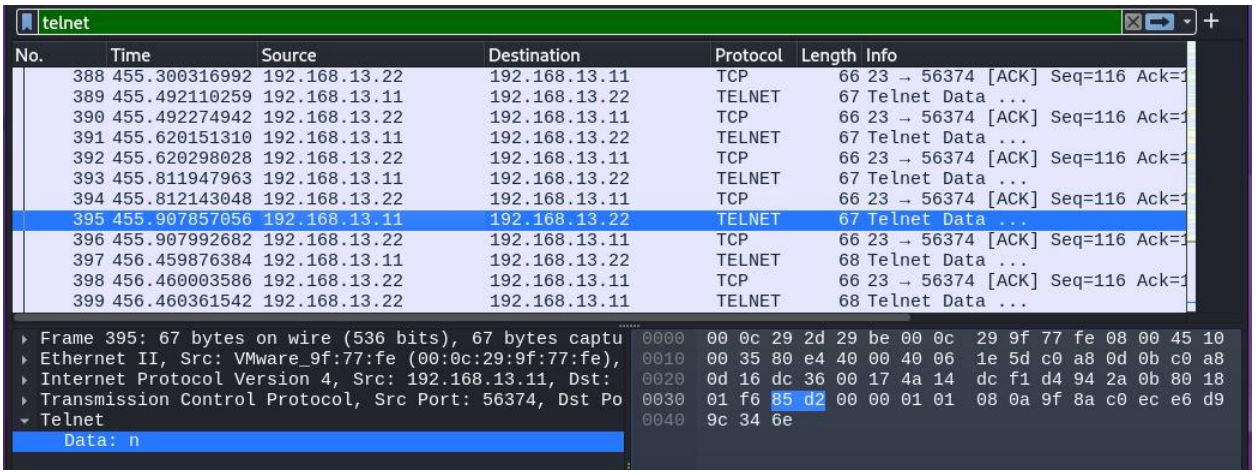



telnet
No. Time Source Destination Protocol Length Info
383 454.740177390 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
384 454.784148070 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
385 455.035669579 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
386 455.035976379 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
387 455.300115952 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
388 455.300316992 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
389 455.492110259 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
390 455.492274942 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
391 455.620151310 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
392 455.620298028 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
393 455.811947963 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
394 455.812143048 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
Frame 387: 67 bytes on wire (536 bits), 67 bytes captu
Ethernet II, Src: VMware_9f:77:fe (00:0c:29:9f:77:fe),
Internet Protocol Version 4, Src: 192.168.13.11, Dst:
Transmission Control Protocol, Src Port: 56374, Dst Po
Telnet
Data: t
0000 00 0c 29 2d 29 be 00 0c 29 9f 77 fe 08 00 45 10
0010 00 35 80 e0 40 00 40 06 1e 61 c0 a8 0d 0b c0 a8
0020 0d 16 dc 36 00 17 4a 14 dc ed d4 94 2a 0b 80 18
0030 01 f6 85 3e 00 00 01 01 08 0a 9f 8a be 8c e6 d9
0040 99 2c 74

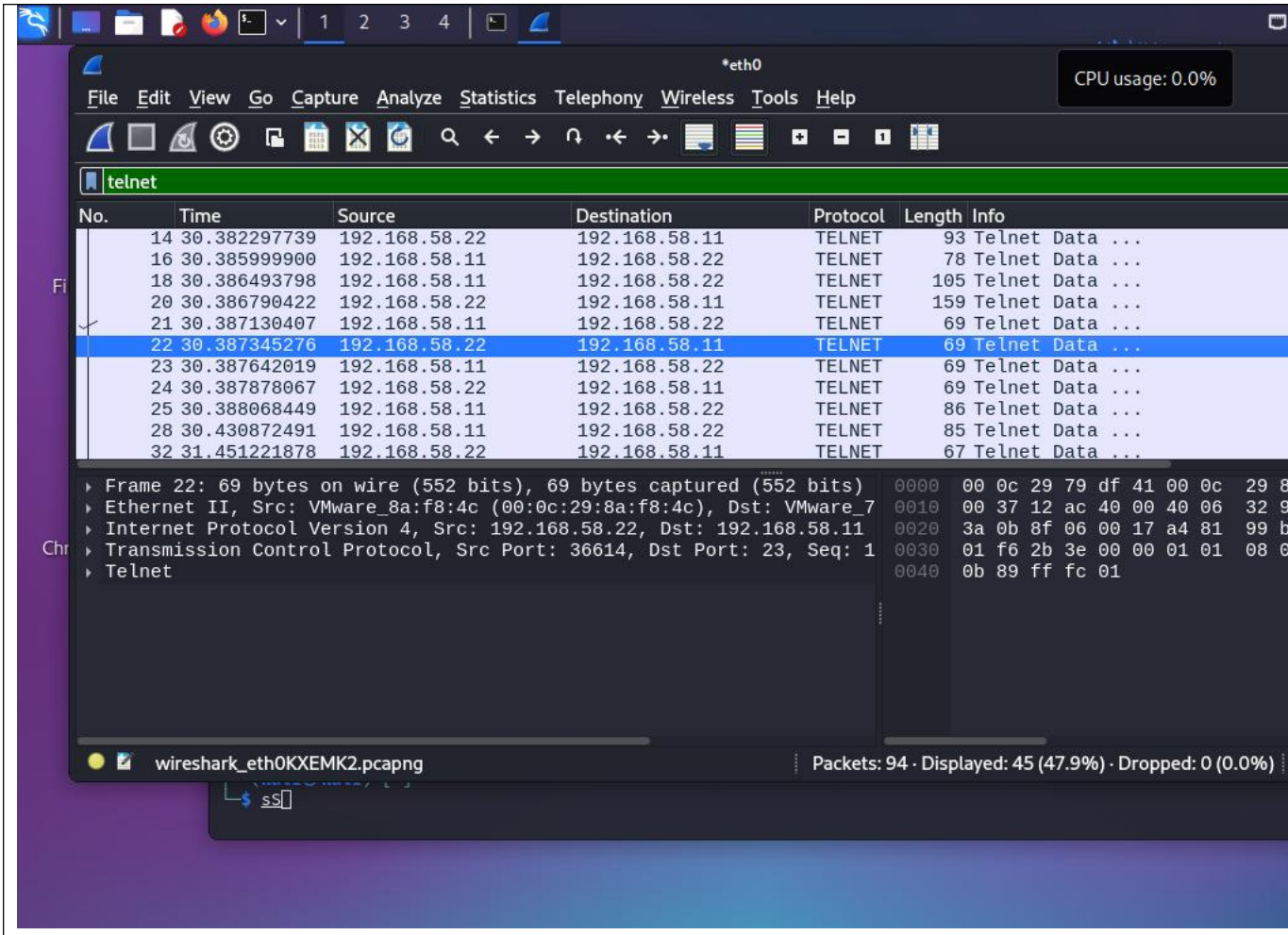

telnet
No. Time Source Destination Protocol Length Info
383 454.740177390 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
384 454.784148070 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
385 455.035669579 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
386 455.035976379 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
387 455.300115952 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
388 455.300316992 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
389 455.492110259 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
390 455.492274942 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
391 455.620151310 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
392 455.620298028 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
393 455.811947963 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
394 455.812143048 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
Frame 389: 67 bytes on wire (536 bits), 67 bytes captu
Ethernet II, Src: VMware_9f:77:fe (00:0c:29:9f:77:fe),
Internet Protocol Version 4, Src: 192.168.13.11, Dst:
Transmission Control Protocol, Src Port: 56374, Dst Po
Telnet
Data: m
0000 00 0c 29 2d 29 be 00 0c 29 9f 77 fe 08 00 45 10
0010 00 35 80 e1 40 00 40 06 1e 60 c0 a8 0d 0b c0 a8
0020 0d 16 dc 36 00 17 4a 14 dc ee d4 94 2a 0b 80 18
0030 01 f6 8a 74 00 00 01 01 08 0a 9f 8a bf 4c e6 d9
0040 9a 35 6d


telnet
No. Time Source Destination Protocol Length Info
383 454.740177390 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
384 454.784148070 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
385 455.035669579 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
386 455.035976379 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
387 455.300115952 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
388 455.300316992 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
389 455.492110259 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
390 455.492274942 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
391 455.620151310 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
392 455.620298028 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
393 455.811947963 192.168.13.11 192.168.13.22 TELNET 67 Telnet Data ...
394 455.812143048 192.168.13.22 192.168.13.11 TCP 66 23 → 56374 [ACK] Seq=116 Ack=1
Frame 391: 67 bytes on wire (536 bits), 67 bytes captu
Ethernet II, Src: VMware_9f:77:fe (00:0c:29:9f:77:fe),
Internet Protocol Version 4, Src: 192.168.13.11, Dst:
Transmission Control Protocol, Src Port: 56374, Dst Po
Telnet
Data: e
0000 00 0c 29 2d 29 be 00 0c 29 9f 77 fe 08 00 45 10
0010 00 35 80 e2 40 00 40 06 1e 5f c0 a8 0d 0b c0 a8
0020 0d 16 dc 36 00 17 4a 14 dc ef d4 94 2a 0b 80 18
0030 01 f6 91 32 00 00 01 01 08 0a 9f 8a bf cd e6 d9
0040 9a f5 65



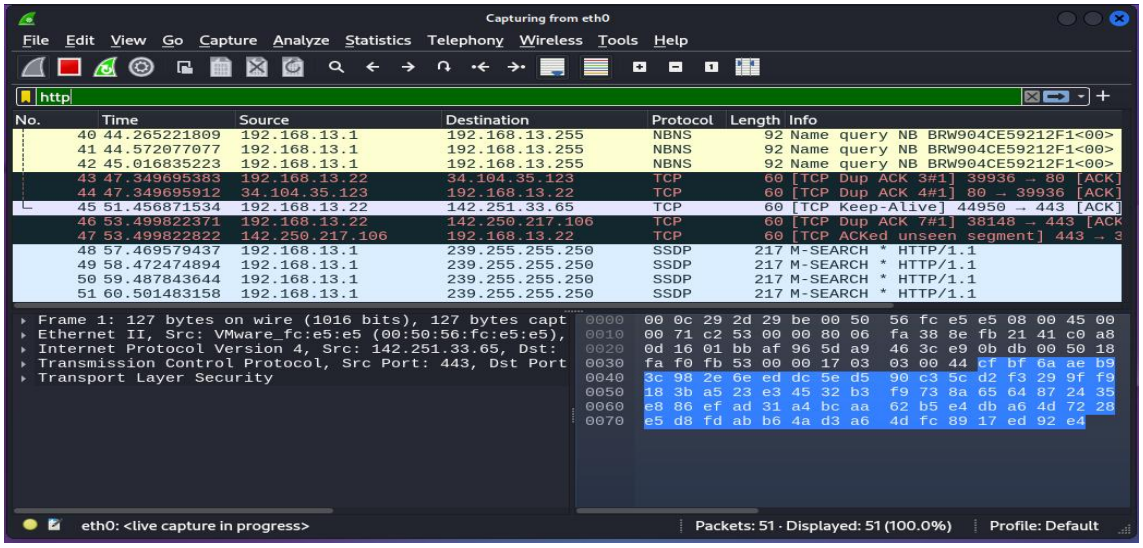


- Now it's your turn to do the same thing and take your similar **screen shots for the password part only** from your Kali VM and insert them below:



3: Sniffing HTTP data packets using Wireshark

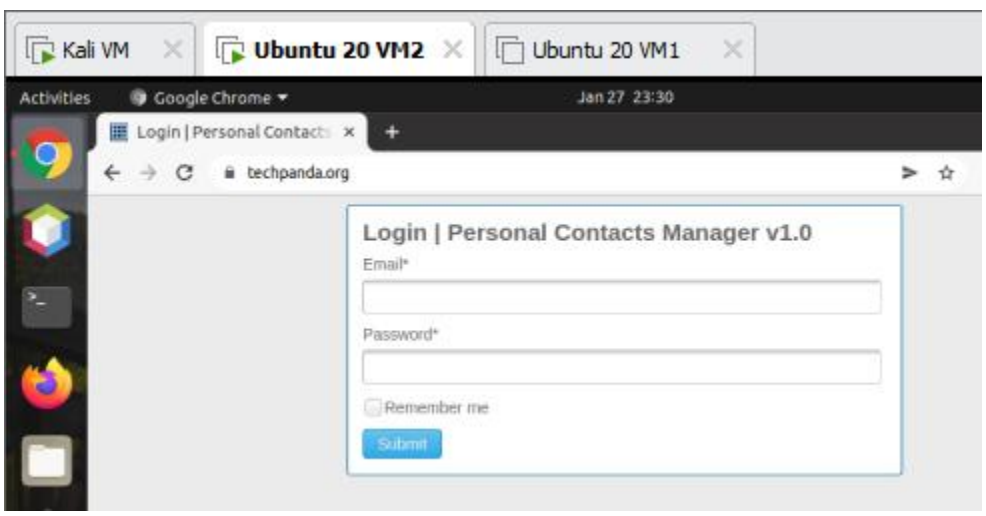
Open Wireshark in your Kali VM and capture the interface eth0 (NAT subnet) with http as the new filter.



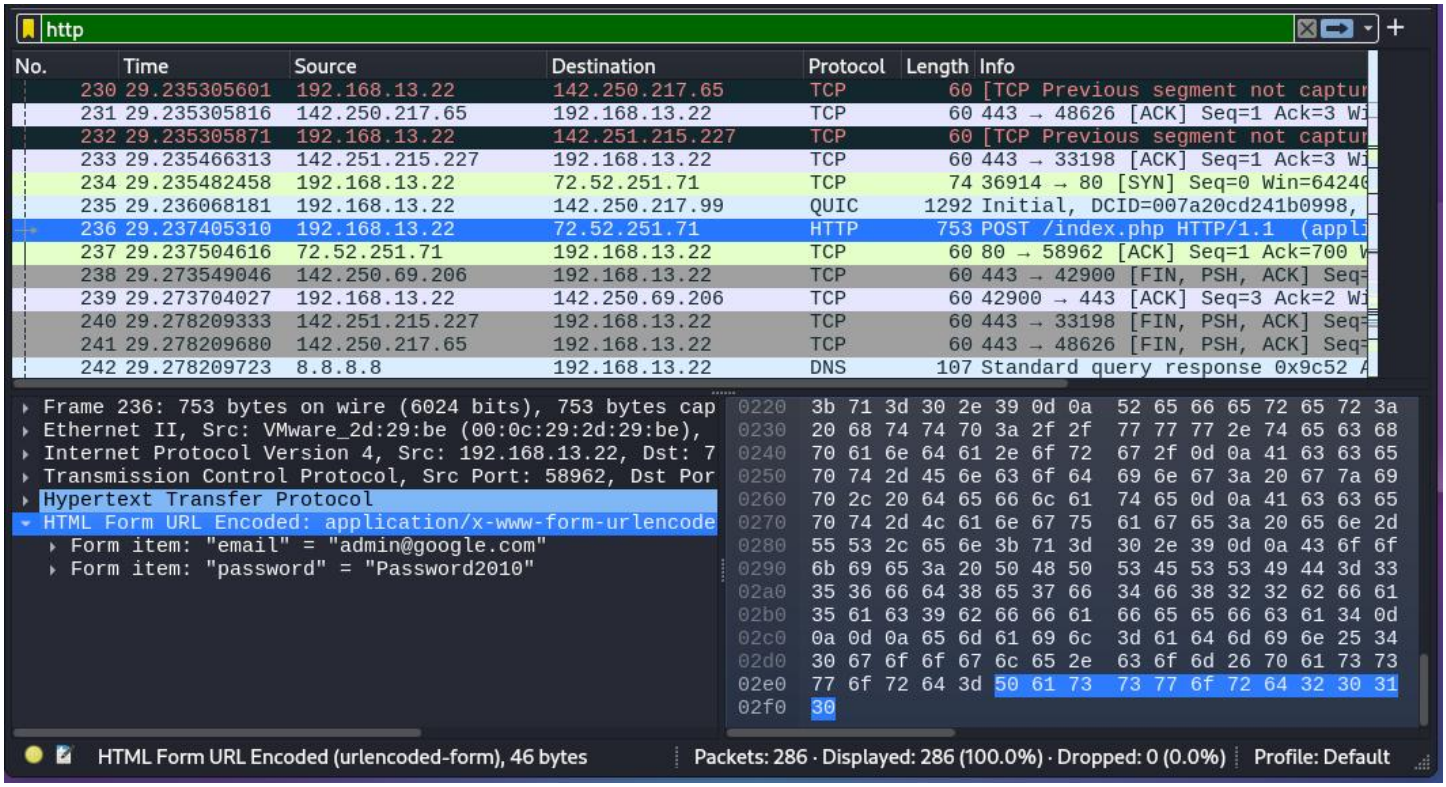
Next refer to the site below:

https://www.guru99.com/wireshark-passwords-sniffer.html

and follow the steps for sniffing a test site (http://www.techpanda.org/) with a browser running in an Ubuntu 20.04 VM. The username and password to be used are **admin@google.com** and **Password2010** respectively for logging into the test site**.**

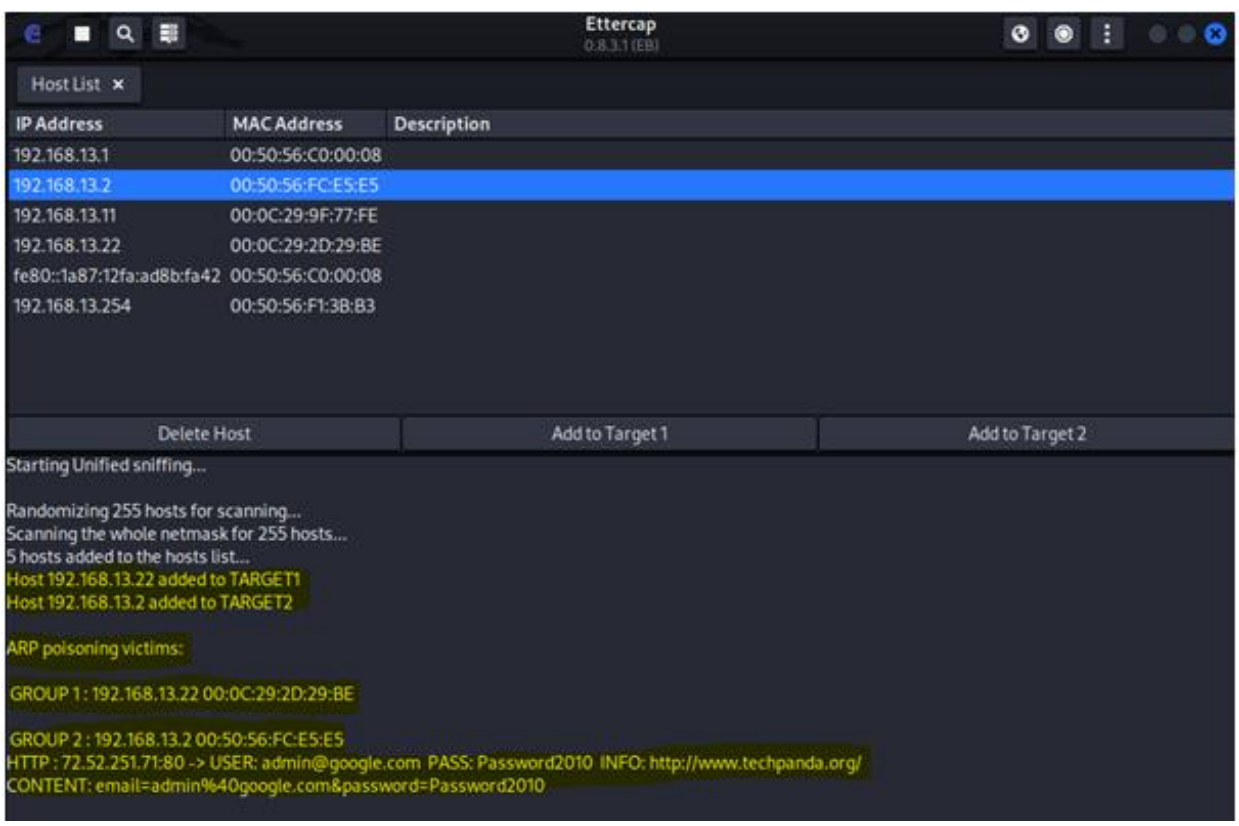


When you logged into the site successfully in your Ubuntu VM, you switch back to Kali VM and stop the packets capturing in WireShark. Look for a captured HTTP packet using POST method in Wireshare:



4: MITM attack with ARP poisoning

- Need an Ubuntu 20.04 VM to be used as a targeted victim in order to see the results.
- Please follow the instructions from " MITM attack with ARP Poisoning using Ettercap.pdf" , and see if you are able to come up with similar results like this:



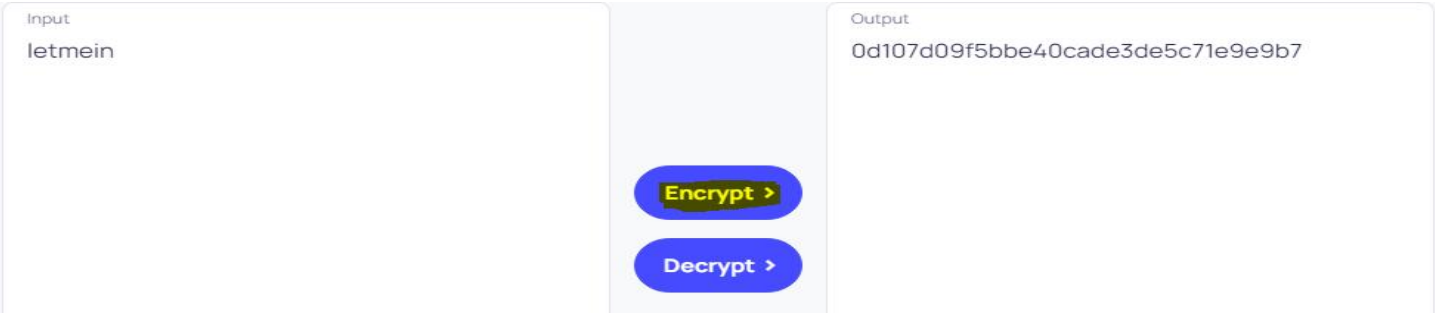
Screen shot of your results here:

```
Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
5 hosts added to the hosts list...
Host 192.168.58.22 added to TARGET1
Host 192.168.58.2 added to TARGET2

ARP poisoning victims:

 GROUP 1 : 192.168.58.22 00:0C:29:8A:F8:4C

 GROUP 2 : 192.168.58.2 00:50:56:E2:22:C8
HTTP : 72.52.251.71:80 -> USER: admin@google.com  PASS: Passwor2010  INFO: http://www.techpanda.org/
CONTENT: email=admin%40google.com&password=Passwor2010

HTTP : 72.52.251.71:80 -> USER: admin@google.com  PASS: Passwor2010  INFO: http://www.techpanda.org/
CONTENT: email=admin%40google.com&password=Passwor2010
```

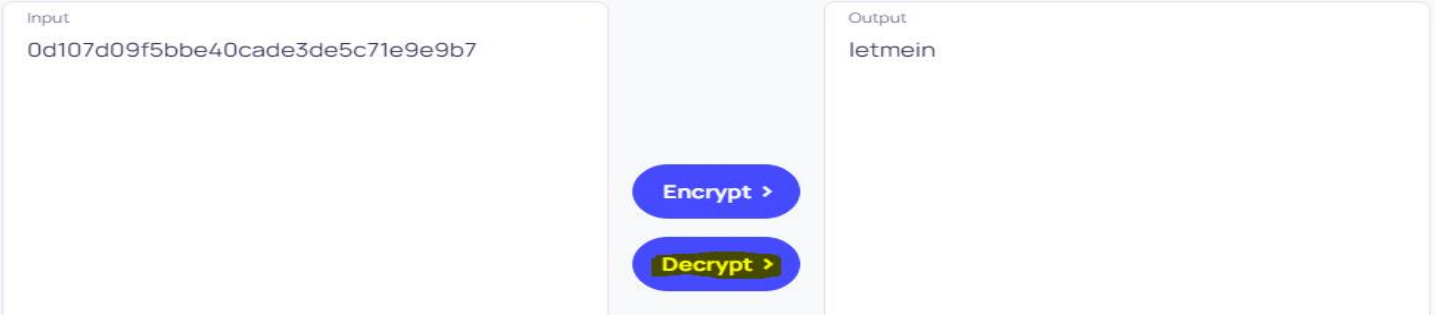
5: Password cracking for hashed passwords

Please refer to the site regarding password hashing (Encrypt) and password cracking (Decrypt): https://10015.io/tools/md5-encrypt-decrypt

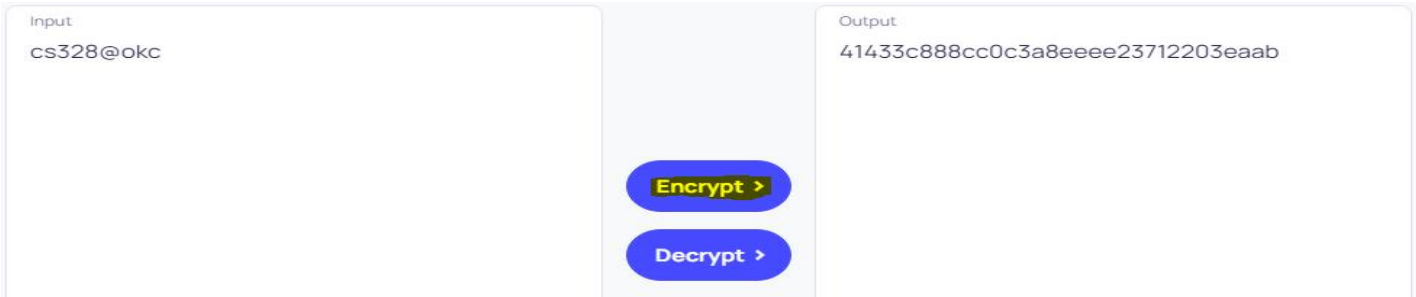
Example1: with a simple password: **letmein**



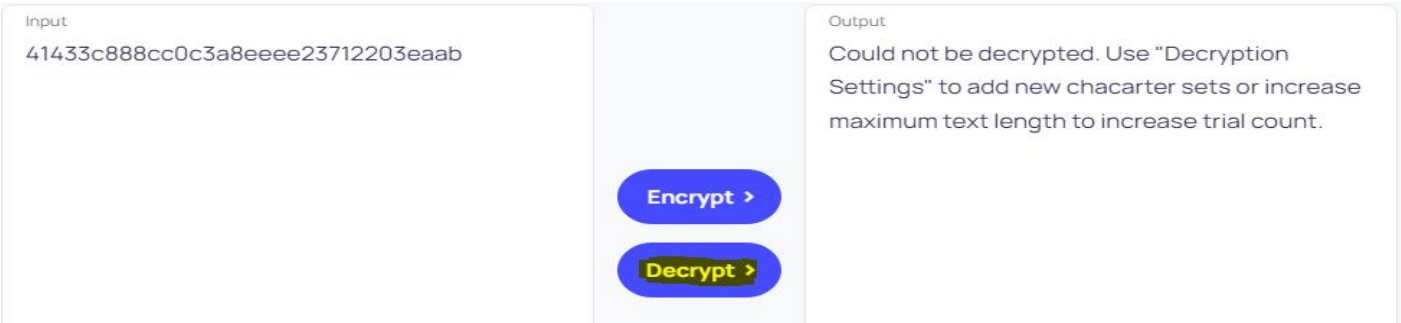
To crack (decrypt) the hashcode "0d107d09f5bbe40cade3de5c71e9e9b7 ", it took a second or less to generate the output "letmein"



Example2: with a relatively strong password: **cs328@okc**

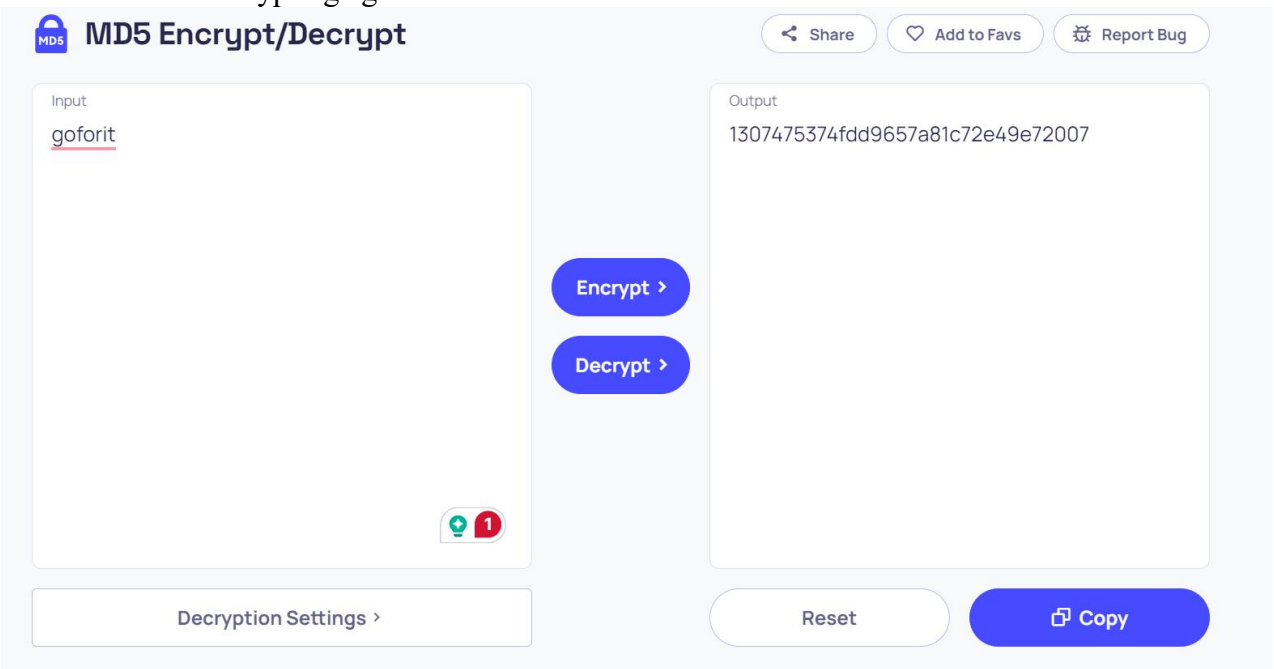


To crack (decrypt) the hashcode “41433c888cc0c3a8eeee23712203eaab”, it couldn’t crack the password successfully within a reasonable time frame and required more character sets and more trial count.

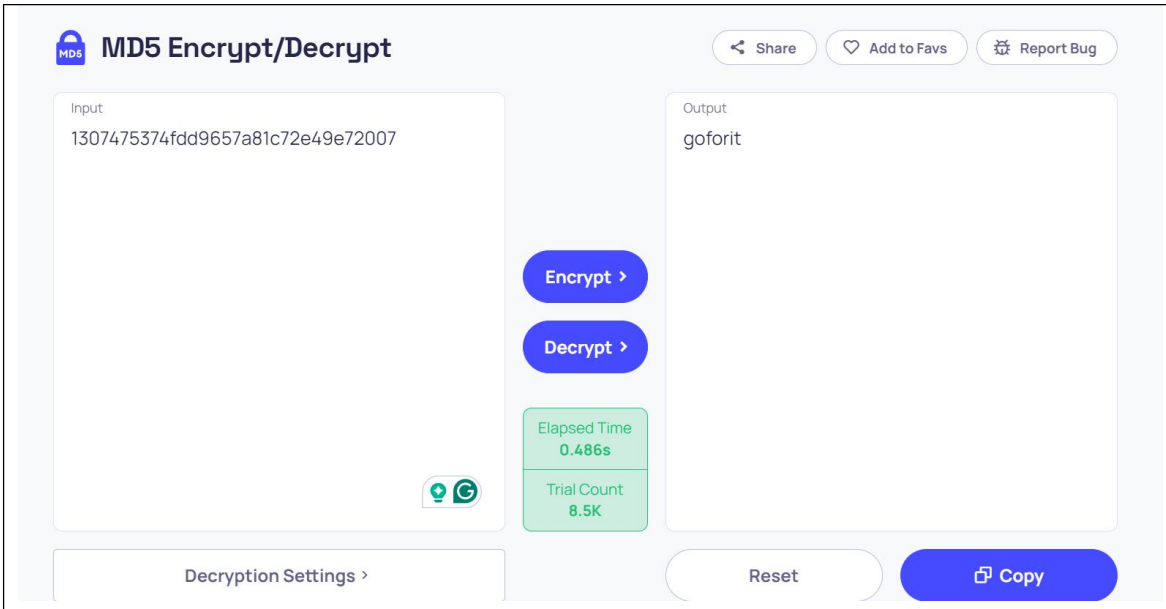


Now it’s your turn to try a simple password “goforit”, take a screen shot for encrypt and decrypt operation respectively and insert them below:

Screen shot for encrypting “goforit”:

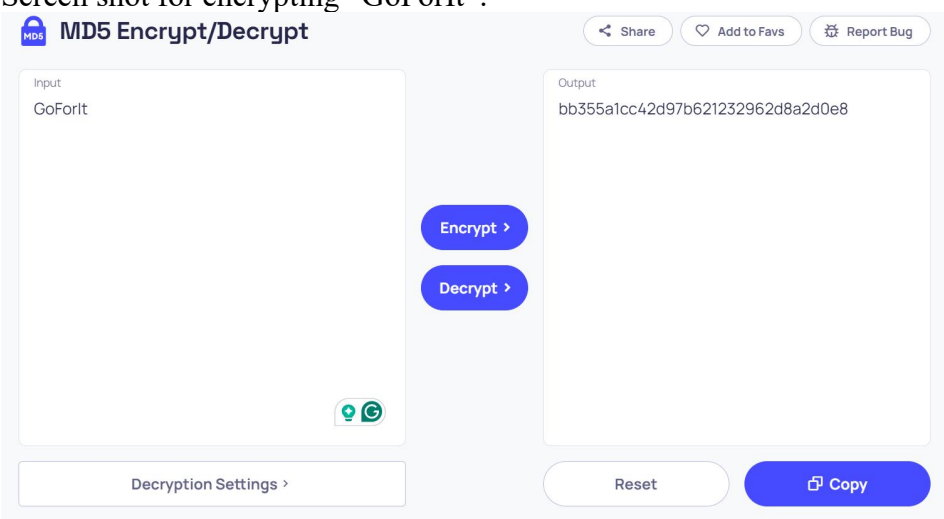


Screen shot for decrypting the hashed code, trying to get the output “goforit”:

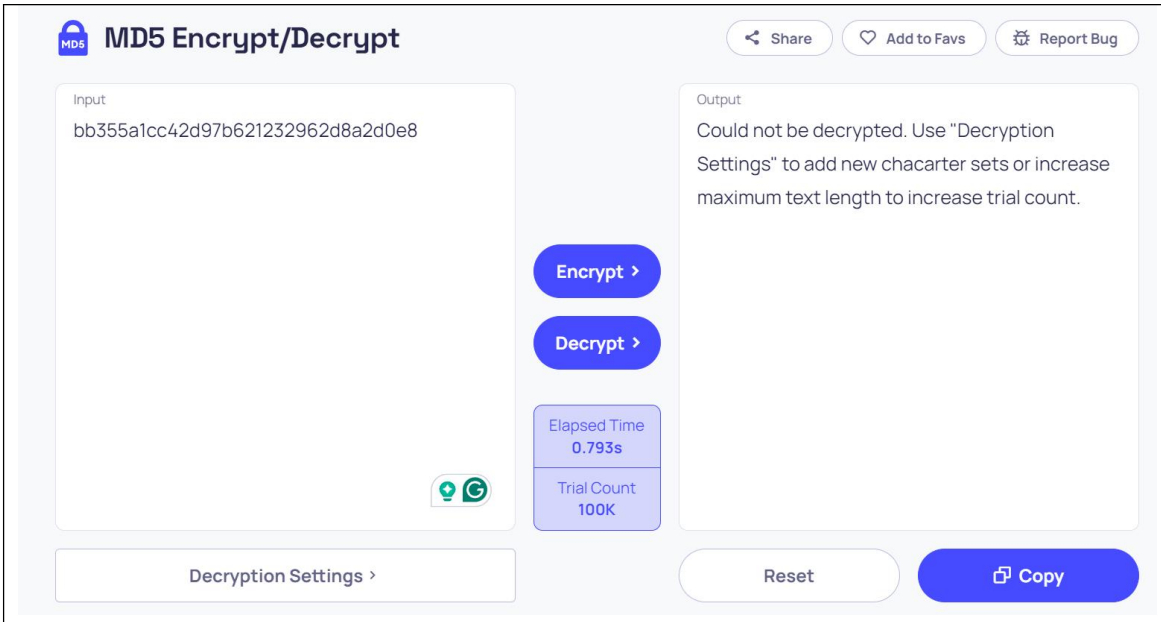


Now try a slightly stronger password “GoForIt”, take screen shots and insert them below:

Screen shot for encrypting “GoForIt”:



Screen shot for decrypting the hashed code, trying to get the output “GoForIt”:



Submitting your work:

Export this Word document with all your answers and screen shots as PDF format, and then submit your PDF file via Lab 7 assignment tab on Moodle by *Sunday, March 17, 2024 (midnight).*