Exercise 1

a) I can't create a tcpdump filter that captures only SYN flood packets. SYN flood attacks can occur in different ways. Such as a direct attack in which the attacker uses a single source device with a real IP address to implement the attack, and a spoofed attack in which an attacker can spoof the IP address on each packet he/ she sends. What's worse, in a distributed attack an attacker may have each distributed device spoof addresses from which it sends packets.

   Even if we know whether the IP address is real or not, we can't identify an attack only according to this. Therefore, I think it's not possible that we can tell all the SYN flood packets concisely from all the SYN packets. Although usually the SYN flood packets are not followed by SYN-ACK packets and we can use this feature to distinguish them from other SYN packets, it's possible that sometimes some normal SYN packets could be not followed by SYN-ACK packets for some reasons such as SYN-ACK packet gets lost. In this case, I think I can't create a tcpdump filter to only captures SYN flood packets.

b) We can open two terminal and use them to capture incoming SYN packets and incoming ACK packets separately during a period of time. In the first terminal, we type the following command to capture SYN packets:

   ```
   tcpdump -i <interface> "tcp[13] == 2"
   ```

   In the second terminal, we type the following command to capture ACK packets:

   ```
   tcpdump -i <interface> "tcp[13] == 16"
   ```

   Because attackers usually choose spoofed IP adresses as its source addresses of the attacking packets, the server will never receive the final ACK from the client(attacker). In this way, we can compare the capture size of SYN packets and ACK packets. If the former is comparatively larger than the latter, then it indicates there is an SYN flooding attack on this server.

Exercise 2

a) The log files of the Chrome web browser describe user's operations on Google Chrome. They are hidden files so we can access them using the terminal. The log files are located at following location on my laptop:

`/home/xucheng/.config/google-chrome/Default`

b) Following are all kinds of log files that are located under `/var/log` on my Linux VM.

```
xucheng@ubuntu:/var/log$ ls
alternatives.log  dpkg.log         journal              vmware
apt               faillog          kern.log             vmware-network.1.log
auth.log          fontconfig.log   lastlog              vmware-network.2.log
bootstrap.log     gdm3             speech-dispatcher    vmware-network.log
btmp              gpu-manager.log  syslog               vmware-vmsvc.1.log
cups              hp               tallylog             vmware-vmsvc.log
dist-upgrade      installer        unattended-upgrades  wtmp
```

The path of the log in which failed logins are recorded is: `/var/log/auth.log`
To prove that, I tried to type incorrect passwords twice on purpose and then I typed the following command in the terminal: `sudo less /var/log/auth.log`
The following info shows two authentication failures in the auth.log

```
Feb 23 12:42:31 ubuntu gdm-password]: pam_unix(gdm-password:auth): authenticatio
n failure; logname= uid=0 euid=0 tty=/dev/tty1 ruser= rhost=  user=xucheng
Feb 23 12:42:37 ubuntu gdm-password]: pam_unix(gdm-password:auth): authenticatio
n failure; logname= uid=0 euid=0 tty=/dev/tty1 ruser= rhost=  user=xucheng
Feb 23 12:42:40 ubuntu gdm-password]: pam_unix(gdm-password:session): session op
ened for user xucheng by (uid=0)
```

c) The following is the data record in the format of <X, A,Y> that I will use :
`<Timestamp,Information,Class>`
The timestamp is the numeric attribute, which records the time when the recorded event happens. Information is a nominal attribute which describes all the information on the subject and event. The Class is the nominal target variable representing the class or type of this data record based on the X and A attributes.
To be more specific, the timestamp is corresponding to the time in the audit records in (b). The Information contains all other content in the audit record or some selected content like “user=xucheng” and “authentication failure”. For Class variable, we add a value like 0 or 1 to represent a category value of this event. For example, failure is corresponding to 0, and success is corresponding to 1. So according to the class, we can use a login counter module like pam_tally in Linux to count the number of times

of 0s, which represents login failure, in one hour. If the number exceeds 3, an alarm should be raised. If not, reset the counter in the next hour, and keep monitoring.