

Assignment 1

Setting up the connection:

The assignment is about establishing a secure connection between a client and a server using the predefined port 7007.

Our implementation makes use of the provider defined in the JSSE (Java Secure Socket Extension).

We chose this provider because it supports both SSL and TLS protocols. This way, they add a level of security when we establish a connection between the two entities. Among others, it includes functionalities related to:

- Data encryption
- Server authentication
- Client authentication
- Message integrity

After a secure connection is established, TCP protocol is used to transmit messages back and forth between the client and the server.

The steps needed to create the connection are as follows:

- Firstly we generate a certificate for both entities, as they are needed for the Handshake part of the connection. In this part, both entities agree on the parameters required to make the connection such as:

- Session ID
- Compression method
- Cipher Suite etc.

```
C:\Users\USER\Documents\Uni Materials\ITU\Semester 1\Security 1\Mini-Project>key
tool -genkey -keyalg RSA -keysize 2048 -validity 360 -alias mykey -keystore myKey
Store.jks
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]: Mr Hacker
What is the name of your organizational unit?
[Unknown]: Hackrama
What is the name of your organization?
[Unknown]: Hackers United
What is the name of your City or Locality?
[Unknown]: Cph
What is the name of your State or Province?
[Unknown]: Denmark
What is the two-letter country code for this unit?
[Unknown]: dk
Is CN=Mr Hacker, OU=Hackrama, O=Hackers United, L=Cph, ST=Denmark, C=dk correct?

[no]: yes

Enter key password for <mykey>
(RETURN if same as keystore password):
```

```
C:\Users\USER\Documents\Uni Materials\ITU\Semester 1\Security 1\Mini-Project>key
tool -export -alias mykey -keystore myKeyStore.jks -file mykey.cert
Enter keystore password:
Certificate stored in file <mykey.cert>
```

```
C:\Users\USER\Documents\Uni Materials\ITU\Semester 1\Security 1\Mini-Project>key
tool -import -file mykey.cert -alias mykey -keystore myTrustStore.jts
Enter keystore password:
Re-enter new password:
Owner: CN=Mr Hacker, OU=Hackrama, O=Hackers United, L=Cph, ST=Denmark, C=dk
Issuer: CN=Mr Hacker, OU=Hackrama, O=Hackers United, L=Cph, ST=Denmark, C=dk
Serial number: 9c00364
Valid from: Tue Oct 09 18:02:01 CEST 2018 until: Fri Oct 04 18:02:01 CEST 2019
Certificate fingerprints:
    MD5: F5:30:A5:4F:72:B4:48:45:2A:21:0A:7A:A4:0F:94:EC
    SHA1: 3F:9D:7A:DB:09:25:E6:76:98:78:5F:EB:37:11:79:05:3B:76:5D:57
    SHA256: 05:BF:48:EB:6A:F0:D3:ED:31:BA:4D:C9:E9:F8:61:E4:6E:D5:74:A3:D5:
85:0B:57:28:2E:89:29:9E:46:20:7D
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 2048-bit RSA key
Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 74 E8 0B 62 2D 1A EB 0B 9A 41 51 73 86 E9 6E F1 t..b-....AQs..n.
0010: 30 07 3B 26 0.;&
]
]

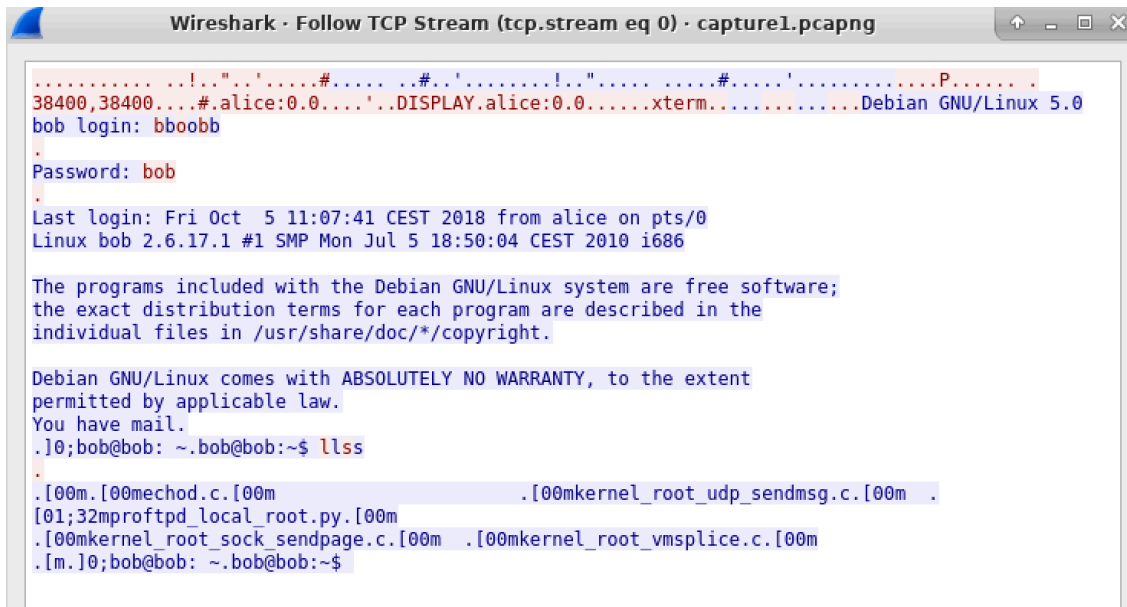
Trust this certificate? [no]: yes
Certificate was added to keystore
```

- After the environment is set up and handshake phase is completed, the two entities can exchange messages using the Secure Socket Layer as long as the client doesn't wish to terminate the connection.
- During the communication, the client writes messages that the server receives and sends back to the client. This is done to verify that the message remains unchanged from the moment it sent by the client until the server receives it, thus we have ensured data integrity.

Assignment 2

Part A

When inspecting the packets in wireshark, we find a packet which has Telnet: Data: Password:. The next packet would then consist of the password, and through wireshark we analyze the packet and follows the TCP stream. From this we get the login credentials.

A screenshot of a Wireshark window titled "Wireshark - Follow TCP Stream (tcp.stream eq 0) - capture1.pcapng". The window displays a terminal session. The text in the terminal is as follows:

```
.....!.."'.#.....#..'!.."#.....P.....  
38400,38400...#.alice:0.0...'.DISPLAY.alice:0.0.....xterm.....Debian GNU/Linux 5.0  
bob login: bboobb  
.  
Password: bob  
.  
Last login: Fri Oct 5 11:07:41 CEST 2018 from alice on pts/0  
Linux bob 2.6.17.1 #1 SMP Mon Jul 5 18:50:04 CEST 2010 i686  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
You have mail.  
.]0;bob@bob: ~/.bob@bob:~$ llss  
.  
.[00m.[00mechod.c.[00m .[00mkernel_root_udp_sendmsg.c.[00m .  
[01;32mproftpd_local_root.py.[00m  
.[00mkernel_root_sock_sendpage.c.[00m .[00mkernel_root_vmsplice.c.[00m  
.[m.]0;bob@bob: ~/.bob@bob:~$
```

Part B

Since TLS uses synchronized encryption, with a private key obtained through a key sharing algorithm, for instance Diffie-Helman, it's not possible to decrypt the traffic.

Part C

When inspecting the results from the keylog, we find a HTTP-POST-request, which contains Alice's log-in credentials. Since these are not encrypted on the client side, Mallory can read these as plain-text.

```
mallory [Running]
Applications *eth0 Terminal - mallory... Terminal - mallory... 13:16

Terminal - mallory@mallory: ~
File Edit View Terminal Tabs Help

Flow Details
2018-10-08 13:15:22 POST https://bob/index.php?option=com_user&task=login
← 301 Moved Permanently text/html 20b 185ms

Request Response Detail
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 115
Connection: keep-alive
Referer: https://bob/index.php?option=com_content&view=frontpage
Cookie: ca835771a6589ed0ff94355cd5a5841f=db9fad7c6cc4460c19374658b4c
edd5e
Content-Type: application/x-www-form-urlencoded
Content-Length: 172
URLEncoded form [m:auto]
username: alice
passwd: alice123
remember: yes
Login: Login
op2: login
return:
aW5kZXgucGhwP29wdGlvbj1jb21fY29udGVudCZ2aWV3PWZyb250cGFnZQ==
881bebd3ef599139f4ca9afc8a04506e:1
[43/44] [*:8080]
```