EITA25 - Group 19 - Project 1

André Frisk, Malin Åstrand, Sebastian Malmström & Axel Beke

January 27th, 2022

Question A: What does the OpenSSL switch -CAcreateserial do?

It works as a counter for the serial numbers on the keys, so instead of having to set serial numbers yourself that are unique for each key it keeps track and does it for you.

Question B: How can you tell keytool to generate a CSR for an X.509 version 3 client certificate (at step 4), or tell OpenSSL to force generation of a version 3 certificate (at step 5)?

By specifying an extension file. A version 1 do not have an extension file so if you add the flag -extfile and then an extension file the version goes to 3.

Question C: What are extensions and what can they contain?

Extensions are files that give flexibility and control over certificates. They can put constraints on the CA, so that it can only sign certificates of a certain nature or for certain action.

Question D: Is it possible to just make a copy of the client-side truststore, why or why not?

Yes, it is possible. Since the truststores function is the to check the validity of the certification sent in by the connecting part. To do this they use certificates from the CA and as along as its the same CA and the same required verification certificate it is possbile to just copy and rename.

Question E: What is the purpose of each password? That is, what does each password protect?

The keypass value that you are prompted for specifies a password for the private key about to be generated. You will always need this password in order to access the keystore entry containing that key. The entry doesn't have to have its own password. When you are prompted for the key password, you are given the option of letting it be the same as the keystore password.

A -> password for truststore
B -> 1st password for keystore
C -> 2nd password for keystore
A protects the integrity of the truststore. It is there so no one can change the trusted public key without access. The 2 passwords for the keystore is for firstly accessing the private keys in the keystore and then one password to change each individual key.

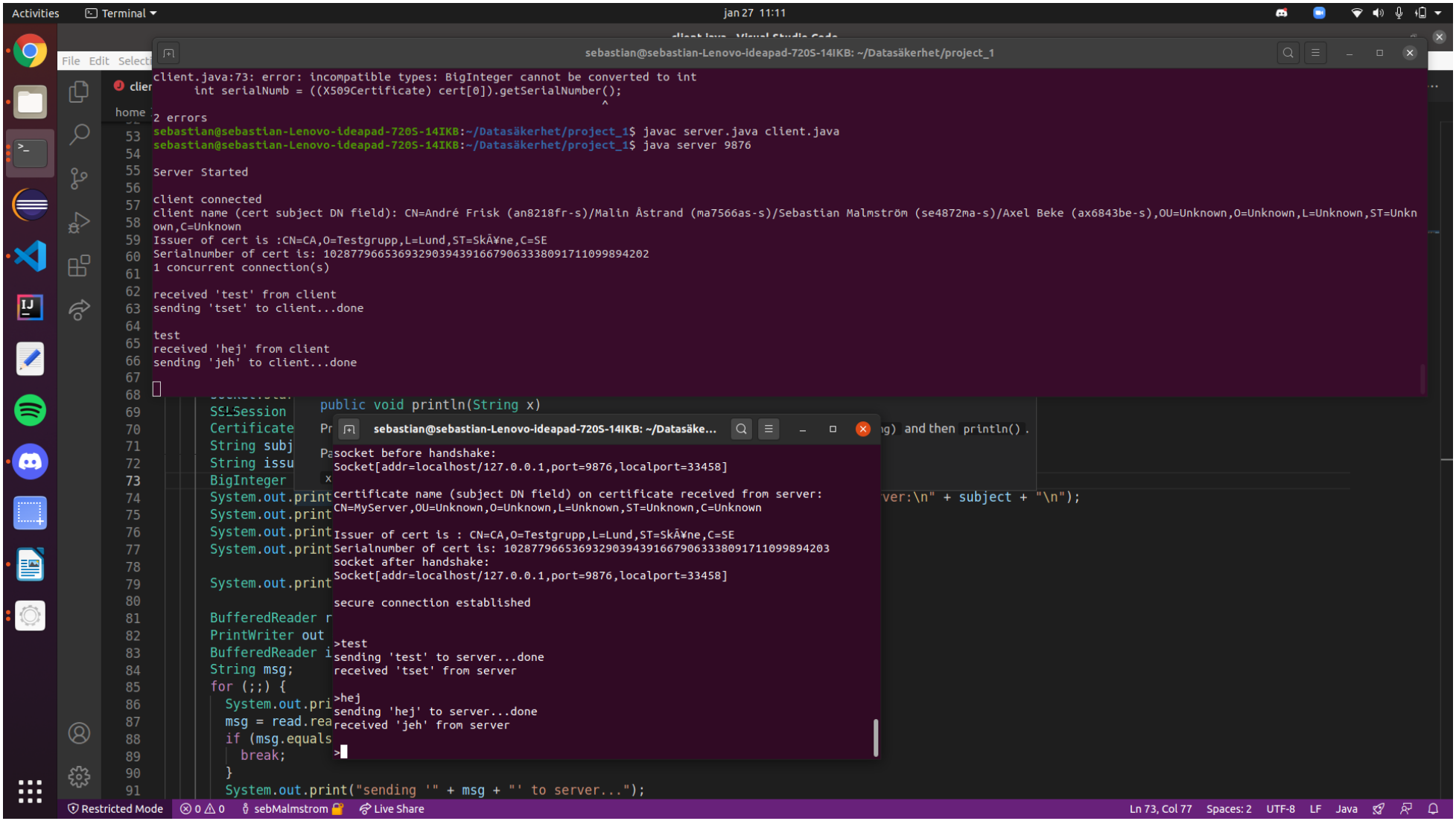
Question F: What does the server answer?

The server answer the input string you wrote but backwards. For example **Hej** will be sent out as **jeH** to the client.

Question G: What is the purpose of setNeedClientAuth(true) in the server?

It controls whether accepted server-mode SSLSockets will be initially configured to *require* client authentication.

Question H: Show printouts, written directly in your submission pdf, from both server and client (we know that this is not really a question, but you get the point).


Activities
Terminal
jan 27 11:11
sebastian@sebastian-Lenovo-ideapad-720S-14IKB: ~/Datasäkerhet/project_1
client.java:73: error: incompatible types: BigInteger cannot be converted to int
int serialNumb = ((X509Certificate) cert[0]).getSerialNumber();
^
2 errors
sebastian@sebastian-Lenovo-ideapad-720S-14IKB:~/Datasäkerhet/project_1$ javac server.java client.java
sebastian@sebastian-Lenovo-ideapad-720S-14IKB:~/Datasäkerhet/project_1$ java server 9876
Server Started
client connected
client name (cert subject DN field): CN=André Frisk (an8218fr-s)/Malin Åstrand (ma7566as-s)/Sebastian Malmström (se4872ma-s)/Axel Beke (ax6843be-s),OU=Unknown,O=Unknown,L=Unknown,ST=Unknown,C=Unknown
Issuer of cert is :CN=CA,O=Testgrupp,L=Lund,ST=SkÃ¥ne,C=SE
Serialnumber of cert is: 102877966536932903943916679063338091711099894202
1 concurrent connection(s)
received 'test' from client
sending 'tset' to client...done
test
received 'hej' from client
sending 'jeh' to client...done
public void println(String x)
sebastian@sebastian-Lenovo-ideapad-720S-14IKB: ~/Datasäke...
socket before handshake:
Socket[addr=localhost/127.0.0.1,port=9876,localport=33458]
certificate name (subject DN field) on certificate received from server:
CN=MyServer,OU=Unknown,O=Unknown,L=Unknown,ST=Unknown,C=Unknown
Issuer of cert is : CN=CA,O=Testgrupp,L=Lund,ST=SkÃ¥ne,C=SE
Serialnumber of cert is: 102877966536932903943916679063338091711099894203
socket after handshake:
Socket[addr=localhost/127.0.0.1,port=9876,localport=33458]
secure connection established
>test
sending 'test' to server...done
received 'tset' from server
>hej
sending 'hej' to server...done
received 'jeh' from server
>
Restricted Mode
sebMalmstrom
Live Share
Ln 73, Col 77
Spaces: 2
UTF-8
LF
Java