Group Size: 2

Part A: Secure file transfer between Alice (student A) and Bob (student B)
1. Create RSA (2048)
key pairs for Alice and Bob and exchange public keys over email. Password protect your respective private keys

2. Alice creates a text file named SA.key with this info <symmetric encryption algo, its parameters and passphrase>. Bob also does same thing (SB.key). These serve like keys for decrypting files exchanged in each way.

3. Alice has to securely send SA.key to Bob. Devise a mechanism in such a way that only Bob can see that message and verify it indeed came from Alice without any tampering. Similarly, Bob has to securely send his SB.key to Alice and prove its authenticity and integrity.

4. Alice encrypts a large file (some PDF/Photo) with SA.key and sends it along with a signature to Bob so that he could decrypt it with the same SA.key and verify it indeed came from Alice without tampering. Similarly, Bob should send some large file securely to Alice without any tampering.

Part B: Alice (Browser), Bob (web server) and Charlie (Root CA)

1. Charlie (one of the TAs of this course) generates a self-signed certificate named charlie-ca.pem or charlie-ca.crt as s/he is the root CA.

2. Bob generates CSR named bob-browser.csr and emails it to Charlie for getting the end-user cert named bob-browser.crt. Note that CSR should contain all relevant details of Bob (Student B) like Country, Organization, CN/SAN/etc. Bob verifies

bob-browser.crt is valid and signed by the root CA, Charlie.

3. Alice (Student A) gets charlie-ca.crt over email from Charlie and bob-browser.crt over email from Bob and verify that Bob's certificate is valid and signed by the root CA, Charlie. Comment on whether Bob's cert is of type X.509 V3, what is the serial no assigned, and what are the key usages/constraints associated with the cert.

Deliverables: Submit a report (Google Doc) explaining how you did you complete these two PARTs. Suffix Alice and Bob with last two digits of your RollNos. Include text contents of
.pem, .crt/.cer, .csr files generated for Alice, Bob and Charlie as part of this assignment in your report using commands like openssl x509 -in <cert-name> -text, openssl req -in <csr-name> -text, openssl pkey -in <public-key-name> -text -pubin, cat <encrypted-private-key > cat SA.key cat SB.key