2 Certificate Authority and PKI (RC)

2.1 PKI Infrastructure (20 Marks)

In task two you will create a PKI infrastructure for a new start up company called TechnoWizard. TechnoWizard wants to get a public key certificate from our CA. You are responsible to get that certificate and verify if it works well.

For simplicity, you create digital certificates without going to pay any commercial CA. You should become a root CA yourself, and then use this CA to issue certificate for anyone (including TechnoWizard servers). You are also allowed to register the certificate in a combination including your own name. Therefore, the name used in the TechnoWizard server certificate must contain TechnoWizard, your last name and the current year. The registered URL will be "www. technowizard.com".

Name the CA's public-key certificate and private key as "ca.crt" and "ca.key". Also, the server's public-key certificate and private key should be named as "server.crt" and "server.key".

For this task you will need to submit evidence of your certificate as well as your public keys. Do not submit your private keys.

2.2 Man in the middle (20 Marks)

After you have generated your own certificate authority and the certificates for server, you will be implementing a secure channel between server and client (in presence of a powerful Man-in-the-middle). You should use system A as client, and system B as server. Store the CA's certificate in the "./client-certs"folder on the client device (A) and use it for your handshake requests. Use the python packages"socket" and "ssl" for your implementation (other packages are not allowed to be used).

For this task you should submit all code created and evidence of your successful man in the middle attack.

3 File Integrity (RC)

3.1 File integrity code (10 Marks)

In your assessment folder on blackboard you will find two files given by TechnoWizard. One of these files has been stored on a secure folder and has not been tampered with. The other was stored on an unsecure server and has been tampered. Identify without opening said files, which file has been tampered with.

As with task two you should use python for your code.

Submission requirements are the generated code files.

3.2 File integrity report (10 Marks)

Write a one page report detailing which file was tampered with, how you identified the file and why you created the identification as you did.

Submit files as a PDF.