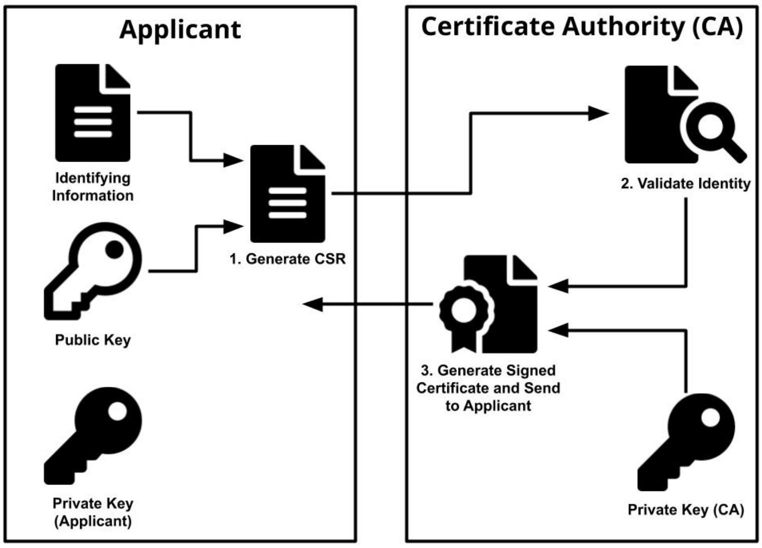
Assignment 3

Certification Authority

A certificate authority (CA), also sometimes referred to as a certification authority, is a company or organization that acts to validate the identities of entities (such as websites, email addresses, companies, or individual persons) and bind them to cryptographic keys through the issuance of electronic documents known as digital certificates.

A digital certificate provides:

- **Authentication**, by serving as a credential to validate the identity of the entity that it is issued to.
- **Encryption**, for secure communication over insecure networks such as the Internet.
- **Integrity** of documents signed with the certificate so that they cannot be altered by a third party in transit.



TLS certificates usually contain the following information:

- The subject domain name
- The subject organization
- The name of the issuing CA, etc

When a user tries to connect to a server, the server sends them its TLS certificate.

The user then verifies the server’s certificate using CA certificates that are present on the user’s device to establish a secure connection. This verification process uses public key cryptography, such as RSA or ECC, to prove the CA signed the certificate. As long as you trust the CA, this demonstrates you are communicating with the server certificate’s subject (e.g., Google.com).

Program Architecture

Program has been divided into 3 parts : One server (CA) and two clients.

Server (CA) :

```
class Certificate:
    def __init__(self,id,PUC,T,DUR,info):
        self.id = id
        self.PUC = PUC
        self.T = T
        self.DUR = DUR
        self.info = info
```

This is blueprint for certification authority.

```
#request is from A(1) or B(2)
def receive_request(self,personAorB):
if personAorB == 1:
print("request received from A.")
elif personAorB == 2:
print("request received from B.")

def return_certificate(self,x):
if x == 1:
return Certificate(1,7,0,60,1234567890)
if x == 2:
return Certificate(1,5,0,60,1234567890)
```

There is 2 main component of certification authority.
1) Receive Request : It is used to recieve request or verify request from clients.
2) Return Certificate : It is used to return certificate for whom clients want to communicate.

Client :

```
def get_PUA(self):
return(self.PUA)

def get_PRA(self):
return(self.PRA)
```

There are 3 main functions in client.
1) Get_PUA : Used to get public key of client1 i.e. A

2) Get_PRA : Used to get private key of client1.

3) Send message : Main driver function of the client used to send message from other clients.

Working

Suppose client A wants to send message to client B so, he needs to get verified public key of client B. And we all know verified public key of every client is stored at CA, which acts as the center of trust. CA provides the public key of client B. Now client A encrypts the message using that public key and sends it to client B.
Client B receives and decrypts the message with its private key.

Conclusion

We have developed a minimal working model of CA. The real architecture of CA and the process of verification of domains is so robust that its very hard to break. In real life, client doesn’t request to CA every time it tries to connect a new domain, instead of that verified certificates of CA are already placed inside the browser.