Assignment 3

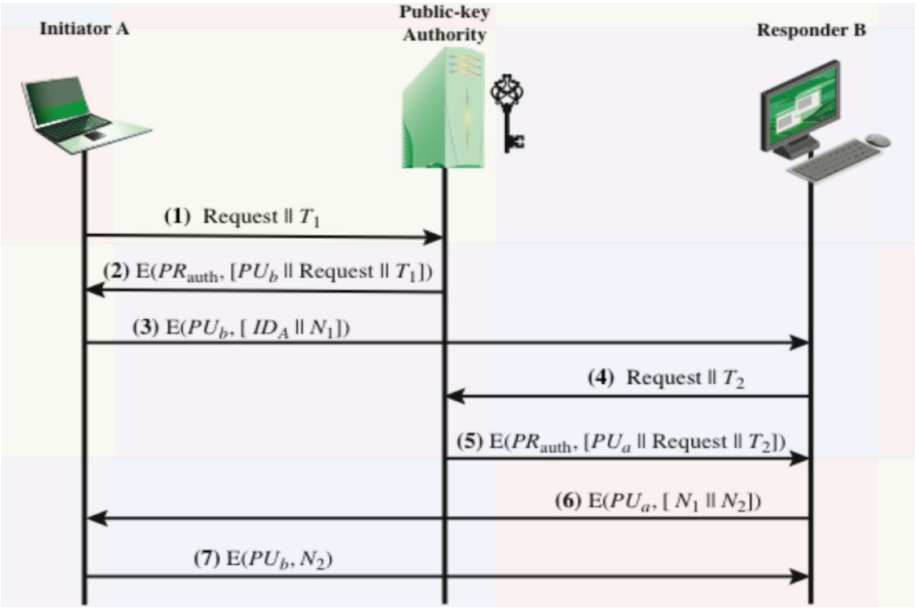
Public Key Distribution Implementation

By: Dheeraj - 2017044, Sehaj Singh - 2017099

Details of Implementation

- To implement the key distribution, we have used socket programming in Python Language.
- In our submission, there are 4 files namely: ClientA.py, ClientB.py, PKDA.py and RSA.py .
- Requests sent to PKDA are not encrypted but all the other messages/replies are encrypted through RSA encryption.
- Steps to run the files:
    - Open 3 separate terminals/powershells/command prompts
    - Execute PKDA.py in one of them and a socket will be opened by the authority to listen to requests from clients A and B.
    - Now execute ClientB.py and finally, ClientA.py
    - All the outputs will be generated in the three respective terminals.

Diagram



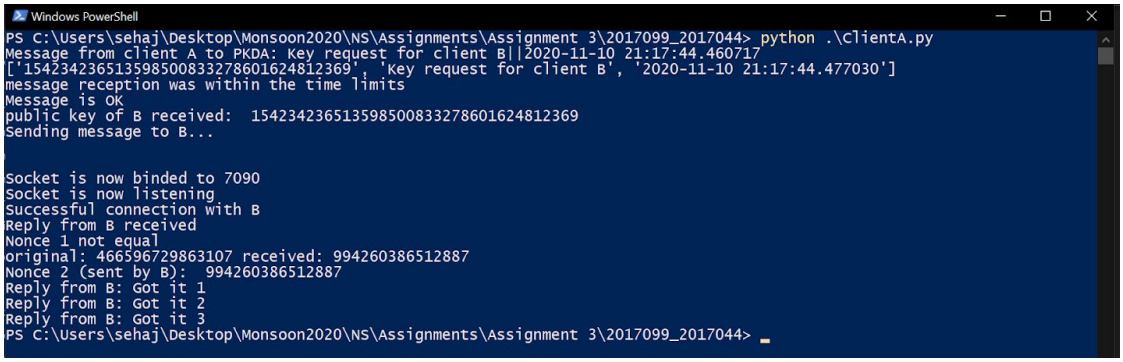
Final Results

- The results obtained were as expected. Please find the screenshots of a sample run at the end of this report.

Some Assumptions

● As mentioned in the problem statement that the clients in the system (somehow) know the distributor (PKDA)'s public key.
● Private keys are kept with the clients themselves.
● Public keys of all the clients are available at PKDA.
● According to the diagram that is shown in the above, the exchange of public keys takes place.
● Client B requires to terminate its connection with Client A so that it can connect with PKDA and fetch A's public key; it takes place after the step(3) in the diagram. This connection will be re-set up after step(5).
● To protect the system from replay attacks, use timestamps in the messages. Messages will be treated as invalid if there is timeout.
● Using a random number, nonces are created with a very large range of numbers, to ensure that previous messages can't be reused for the replay attacks; they will be used for authentication protocol.
● The received message buffer size is 1024.

Sample Run



```
Windows PowerShell
PS C:\Users\sehaj\Desktop\Monsoon2020\NS\Assignments\Assignment 3\2017099_2017044> python .\ClientB.py
Socket is now listening
Successful connection with A
Request from A is received
ID of A: 1  Nonce received: 466596729863107


Message from client B to PKDA: Key request for client A||2020-11-10 21:17:44.489915


['182106040449176377254583258871575511', 'Key request for client A', '2020-11-10 21:17:44.489915']


message is received within time
Message received from PKDA: ['182106040449176377254583258871575511', 'Key request for client A', '2020-11-10 21:17:44.48
9915']
Public key of A received from PKDA:  182106040449176377254583258871575511
Sending confirmation/reply to A...

 Received confirmation of A ['994260386512887']
Nonce sent is equal to nonce received
Message from A: Hi 1
Message from A: Hi 2
Message from A: Hi 3
PS C:\Users\sehaj\Desktop\Monsoon2020\NS\Assignments\Assignment 3\2017099_2017044>
```

```
Windows PowerShell
PS C:\Users\sehaj\Desktop\Monsoon2020\NS\Assignments\Assignment 3\2017099_2017044> python .\PKDA.py
Socket has been created
Socket is now binded to 9060
socket is now listening
Client A is now connected....
Message is received within time
Sending public key of B
Message being sent:  154234236513598500833278601624812369||Key request for client B||2020-11-10 21:17:44.477030
Successfully connected to client B
Public key of client A is being sent
PS C:\Users\sehaj\Desktop\Monsoon2020\NS\Assignments\Assignment 3\2017099_2017044>
```