Project_3_Report(2022577 and 2022583)

Programming_Language: Python

Porblem statement

As per the algorithm outlined in the assignment, we have been assigned Project Zero, which is a RSA-based Public-key Certification Authority (CA) and we followed all the assumptions as given in this assignment. For the security purpose , every client will generate its private and public key using RSA Algorithm.

RSA key pair generation Algorithm

First of all, we assumed four large prime number two for each client which are about 1024 bit long digits and computed n,z, e , d where n=p*q ,

z=(p-1)*(q-1)

We select e from a set of prime numbers such that it is relatively prime to z and satisfies e < n, and compute d as the modular inverse of e using the gmpy2 library. A Certification Authority (CA) is established to issue certificates, allowing clients to request their own public key certificate or that of another client.

Certificate Authority's Implementation

Every client knows about the private and public key of CA such that it can get the information from the certificate which is provided by CA .To obtain a certificate, a client must first register with the CA by providing its ID and public key. The CA then issues a certificate containing PR-CA (the private key of the CA), ID (the client's ID), IDCA (the CA's ID), PU (the client's public key), TA (the issuance time), and DURA (the validity duration, assumed to be 3600 seconds). The certificate data is encrypted using the SHA-256 algorithm, and the final certificate is a concatenation of the original data and its encrypted version. Once registered, each client is stored in the CA's database, and when a client requests a certificate, the CA retrieves and provides the corresponding certificate.

Assumed, Client A requests the certificate from CA, then CA will provide the certificate by checking this , and when a client receives a certificate, it verifies the digital signature using the CA's public key and if verification succeeds, the certificate is considered valid.

Verification of Certificate

From this client A can extract the public key of another client and send the message with encryption of using public key of another client annd another client can get that messages using his/her private key. Now, both Client A and Client B successfully register with the CA and CA issues signed certificates for both clients and then clients verify the authenticity of certificates using the CA's public key so,this verification process ensures integrity and prevents tampering .

Encryption and Decryption of messages

Assume that client A sends three messages ("Hello1", "Hello2", "Hello3") to client B where each message is encrypted with client B's public key and decrypted using client B's private key and then client B responds with ACK1, ACK2, and ACK3 upon successful decryption and this process is repeated when Client B sends messages to Client A.

Output for clientA like how this mechanism works.

Certificate A:

```
 {
   "CERT": [
     [
       "ID1",
       [
         9815509181410530771610496987 56…….,
         3
       ],
       1743334864,
       3600,
       "CA_ID25608"
```

```
        ],
        "35bb044be5b91567baa058282fdd43033ec78b657a92b……….
    ]
}
Certificate verification successful!
ID1 requested the certificate of ID2 from CA.
Extracted Public Key of Client B (from CA): (22673861473944013639628…..)
Client A sending messages to Client B
b'\x00*O\xd9\xe1q\xcf\x19)\x1e\x06\xf6\x04\xcc\x85g\x…….'
b'Hello1'
Client B decrypted message: Hello1
ACK 1
```