Network and System Security

Assignment-4 Report

Name: **Kaustav Vats**
RollNo: **2016048**

Designed a multi-threaded server to which the users now connect remotely. The users are to be authenticated via a Needham Schrodher (NS) authentication scheme. Server listens on two ports, 5555 for KDC and 6666 for Chat App. All communication between users are encrypted with shared keys. I have also implemented Diffie Hellman algorithm. For any size of the group.

Needham-Schroeder protocol allows to prove the identity of the end users communicating, and also presents a middle man from evesdropping.
We will be using some terms in this document which needs to be understood first.
Nonce: Nonce is a randomly generated string which is only valid for some period of time, This is used in encryption protocols to prevent replay attack. For example if somebody captures a packet during the communication between me and a shopping website, he can resend the packet without decrypting it, and the server can accept the packet and do operations on it. To prevent this, nonce(the random value generated) is added to the data, so as the server can check if that nonce is valid, or expired.

The Diffie-Hellman algorithm is being used to establish a shared secret that can be used for secret
communications while exchanging data over a public network using the elliptic curve to generate points and get the secret key using the parameters.

- For the sake of simplicity and practical implementation of the algorithm, we will consider only 4 variables one prime P and G (a primitive root of P) and two private values a and b.
- P and G are both publicly available numbers. Users (say Alice and Bob) pick private values a and b and they generate a key and exchange it publicly, the opposite person received the key and from that generates a secret key after which they have the same secret key to encrypt.

Functions implemented

- **/who:** Prints who all are logged in to the chat server, along with a user IDs.
- **/write_all:** Write message which gets broadcasted to all users.
- **/create_group:** Creates a group and return group id.
- **/group_invite:** Sends an invitation to user with group id.
- **/group_invite_accept:** User enters the group id to accept the group invite.
- **/request_public_key:** Send request for public key to a specific users.
- **/send_public_key:** Sends public key of the user to entered user id.
- **/init group dhxchg:** This process initiates a DH exchange rst with any two users and then adds more users to the set.
- **/write_group:** Writes a message to the group members created using the shared key.
- **/list_user_files:** Takes user id of the user and prints the files of that particular user.
- **/request file:** Request a le and also specify the port number to connect to and send. Sender waits for that file at that port.

Protection from the vulnerabilities:

- User can only be logged in from his own shell. No other user can login using his/her credentials
- User password protection- while authenticating itself with the KDC, password of the user is visible using *.
- Buffer overflow protection.
- Extension of above password protection functionality- Password will be entered like any other linux system. **On terminal password won't be visible and length of the password will also be unknown.**