Highly Dependable Systems

Project – Stage 1

Highly Dependable Location Tracker

Group 14:

Francisco Seixas – 83462

Pedro Campos – 83951

Maria Margarida Lopes - 86477

1 - Introduction

As proposed and recommended by the teachers, we decided to go with a simple Client-Server implementation through the use of sockets.

There are three types of users: common users, byzantines and a super user. Each one of them have specification, in order to keep the implemented ambient organised. Common users are allowed to submit Report Locations and serve as witness to other users; however they are not able to move freely in the board. Byzantines however can move during an epoch, since they "free will" movement. The super user has special access to other user's locations.

Since user's can serve as witnesses and provers, we implemented the communication between them with recourse to threads and tcp channels. This way, we can simulate a "Bluetooth" transmission when a user asks another to prove their location at a certain epoch.

2 – Integrity

The Client-Server communication is performed over a secure channel that implements a Hybrid Cryptosystem Model. Assuming a proper key Distribution, each User posses an RSA Generated Asymmetric Key Pair (User Public Key + User Private Key) and the Server's Public Key; The Server knows all User's Public Keys and has its own Asymmetric Key Pair.

For any intention to communicate, the User randomly generates (using AES) a one-time Symmetric Key.

Basically, all data which is going to be transferred, is encrypted with a that symmetric key (Assuring Confidentiality) and the symmetric key itself is encrypted using asymmetric encryption. Both encrypted piece of data and encrypted key are delivered to the recipient.

Essential feature about hybrid encryption is that symmetric key is created over and over for every new session. The power of random numbers is that its consequence cannot be guessed, repeated or predicted by a hacker. This approach ensures forward security of communication is case of leaks of symmetric keys from previous sessions (Replay Attacks).

Authentication, Data Integrity and Non-repudiation are also assured. Each communication party creates a digital signature by encrypting the hash value of the message, using it's Private Key. At the receiving side this digital signature is used for Authentication, Data Integrity and Non-repudiation checking.

3– Dependability

In order to use threads as means of communication and to maintain the system internal state, the use of external files such as epochIMap.txt(0< I <N), which describes all the locations of the users prior to submitting report, i.e. the server does not have access to it. We also use epochI.txt, which saves the internal state of the Server, and even if the Server goes down for some reason (maintenance or power cut), the data is saved in the file, so it can be used by another instance of the server.

4– Conclusion

Although we implemented protection against data manipulation by some attacker disguised as user, we did not implement a test which shows the mechanism working. However in order to save data in the server, the proof given by a witness is 100% checked by the Server, which will answer accordingly when the usedID from the witness does correspond to the signature sent.