Threshold Cryptosystem – Advanced Cryptography and Computer Security 01/2017 - 05/2017

Group Members:

- Aditya Nalge
- Arti Gupta

Developed a Threshold Cryptosystem which is a (n,t) encryption system for a group of 'n' trusted users allowing any member to encrypt a message and which requires at least a threshold 't' number of members to work together to decrypt it.

- Elgamal Encryption was used to create the Public Key and Encrypt the plain data using the obtained Public Key
- Shamir's Secret Sharing Algorithm was used to create Secret Shares for each user.
- Unlike traditional decryption, there is no secret key for decryption.
- Users create their own decryption shares using their own secret share and the first part of cipher text.
- Decryption of cipher text is done only when a threshold number of users enter their decryption shares in the system.

Digital Signature Algorithm (DSA) was used to implement digital signatures that provided Non repudiation and Authorization. The encrypted data can be decrypted even if multiple users lose their decryption shares as long as a threshold number of decryption shares are available. NO user has the knowledge of the original secret being used for encryption or decryption.

Results:

Performance Analysis:

- We tested the project with different test cases and were able to retrieve the original message every time.
- Initially, we hardcoded calculation of power values in our code. This was time consuming. Using the in-built pow function allowed us to improve time efficiency and consequently code is execution is faster.
- We intended to use cryptographic libraries for various phases of our project, but, since it involved several concepts (decryption shares 5 times for 5 users etc), we are manually computing the shares and using cryptographic libraries only for implementing digital signatures and random number generator.

Security Analysis:

- Instead of using any random function, we are generating random numbers using cryptography libraries to make it strongly secure as it is a strong random function.
- Our project provides security in many aspects. We have implemented digital signatures to avoid non-repudiation and provide authorization.

Technology Used: Python ▪ Cryptographic Libraries

NOTE: Exhaustive Project Documents and Source Code are Available on https://github.com/adityanalge