Report

Assignment 4
Malmö University

MALMÖ HÖGSKOLA

2016-11-03

Simon Gullstrand
Jonas Wahlfrid
Björn Hansson

System architecture

Client

The client is implemented in Java and the GUI is built with the widget toolkit Swing. With GUI elements for setting the camera IP, port, password, resolution and FPS.

Server

The server is implemented as a ACAP application using C. It is uploaded and running on the camera.

Communication Interface

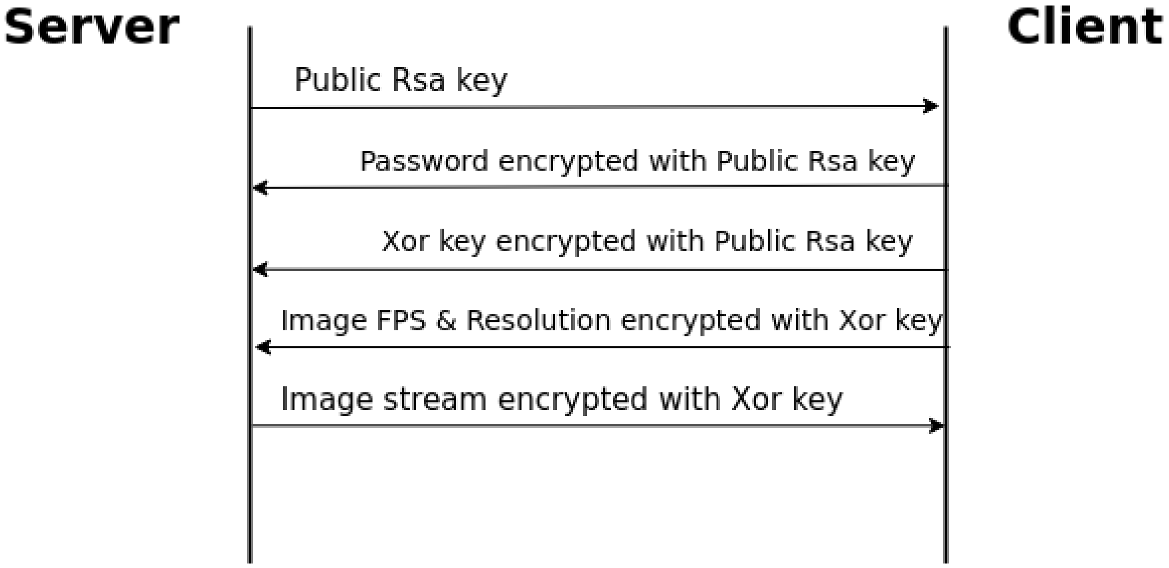
Server
Client
Public Rsa key
Password encrypted with Public Rsa key
Xor key encrypted with Public Rsa key
Image FPS & Resolution encrypted with Xor key
Image stream encrypted with Xor key

Figure 1: The communication interface between the Server and the Client.

Security implementation

1. The server is using RSA [1] and sends its public key to the client. The client encrypts the password to access the camera, using the server's public RSA key. Then the server decrypts the password, using its private RSA key. If the password is wrong, the client gets disconnected.
2. Next the client generates a unique XOR [2] key and encrypts it with the server's public RSA key. The server decrypts the XOR key, using its private RSA key. Now communication with XOR encryption is possible from both sides.
3. The client sends the user specified FPS and resolution encrypted with XOR. The server decrypts it and sets the desired settings.

4. The server then sends the image stream encrypted using XOR. The client decrypts it and projects the “video” to the user through the GUI.

This solution is only using RSA for sending the password and XOR key to the server. XOR is used for the rest of the communication. The reason for this is because XOR is faster and a more lightweight alternative then to implementing RSA on both sides. RSA is used for the critical information only. This way, we only needed to implemented most of the RSA functionality on the server. The client only needs to be able to encrypt with RSA. The obvious vulnerability is that XOR can basically be decrypted by guessing. An alternative to our solution which would increase the security of the communication is to use multiple encryptions. For example AES [3] algorithm with SHA256 hashing.

Security mechanisms

In order to achieve the perfect security every aspect of the system must be secure. Which means that not just the communication between the client and the server must be encrypted but also the process of encrypting and decrypting must be well thought out. Since a “hacker” could try to duplicate the client software to act as the real client to gain access to the systems communication. This is prevented with the use of firstly RSA cryptography on the server where only the public key will be visible to the client, which also can not be used to decrypt the messages. Secondly the client has to send a four digit (number based) password encrypted with the public key and if the password is wrong the communication is interrupted. If the password is ok then the client can send its uniquely generated XOR key encrypted with the RSA public key to the server to enable further communication [4].

Thoughts about the assignment

The assignment was amusing and challenging. Cryptography together with C language and Axis camera can however be very difficult, but it also gives opportunities to acquire a lot of new knowledge. As in earlier assignments, the client implementation was a lot simpler to develop.

References

[1] https://en.wikipedia.org/wiki/RSA_(cryptosystem)
[2] https://en.wikipedia.org/wiki/XOR_cipher
[3] https://en.wikipedia.org/wiki/Advanced_Encryption_Standard
[4] https://en.wikipedia.org/wiki/Information_security#Access_control