EDA385 Embedded Systems Design. Advanced Course

LUNDS UNIVERSITET

Encryption for Embedded Systems

Supervised by

Flavius Gruian

Submitted by

Ahmed Mohammed Youssef (aso10ayo)

Mohammed Shaaban Ibraheem Ali (aso10mib)

Orges Balla (aso10oba)

Introduction

Nowadays many electronic devices (mobile phones, PDAs, Security Cameras…) need to communicate with other devices via unsecured communication links, using such unsecured communication link is dangerous in case of communicating sensitive data, which may cause economic and/or security damages by illegally knowing it.

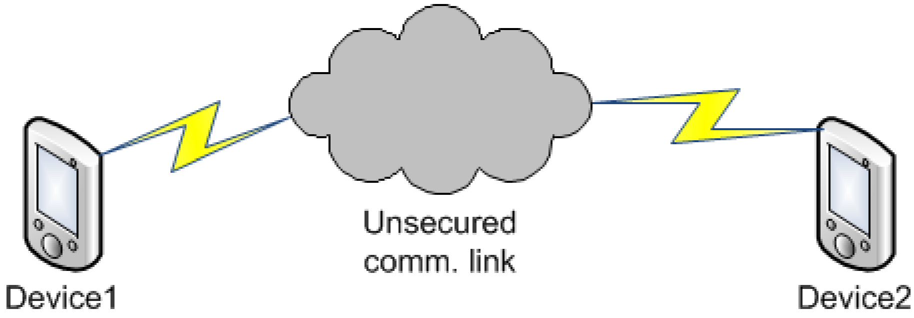


Figure 1 : overall view

Even if the network itself is claimed to be secured, this cannot be taken for granted for communicating sensitive data, so the requirement of securing the data locally on the device level arises.

Project Proposal

In this project we provide a security system that can be integrated with any electronic device that requires encryption/decryption capabilities for the data it holds.

We adopt a hybrid cryptosystem, where we have two cryptosystems included, namely the public-key and the symmetric-key cryptosystem.

We use the symmetric-key cryptosystem for encrypting the data to be transmitted using session keys (one key per session), while the public-key cryptosystem is used to encrypt the session keys used in the symmetric-key cryptosystem.

For public key cryptosystem each communicating entity should have a key pair:

1. The public key which can be known by all the devices in the network, and
2. The private key which is only known by its owner.

By encrypting data with a device public key, only the private key holder can decrypt this message back.

This project deals with only providing an encryption/decryption system for embedded systems. The entire necessary infrastructure that makes RSA works, like a public server where someone can find the public key of the receiving party is assumed to exist. It is also assumed that for the purpose of the project, the receiving party is properly authenticated. Also the inner workings of the cryptography system will not be explained in details, for the project we're working is not about how the cryptography works but how we can make use of it. The algorithms that will be used in this project are accepted standards in the cryptography community.

Message Life Cycle

A typical message life cycle would be as follows:

At the sender's side:-

1. Message is divided into blocks (According to the used block cipher-Symmetric-key cryptosystem-), where the data is called the Plaintext.
2. A session key is generated for each message transmission, session keys are normally random numbers generated in a secure way.

3. Session key is used then to encrypt the data blocks using the block cipher, to produce the Ciphertext.

4. Session key is encrypted using both the sender's private key (for authentication) and the recipient's public key (for confidentiality).

5. Both encrypted session key and the Cipher text are sent over the unsecured communication link to the other device.

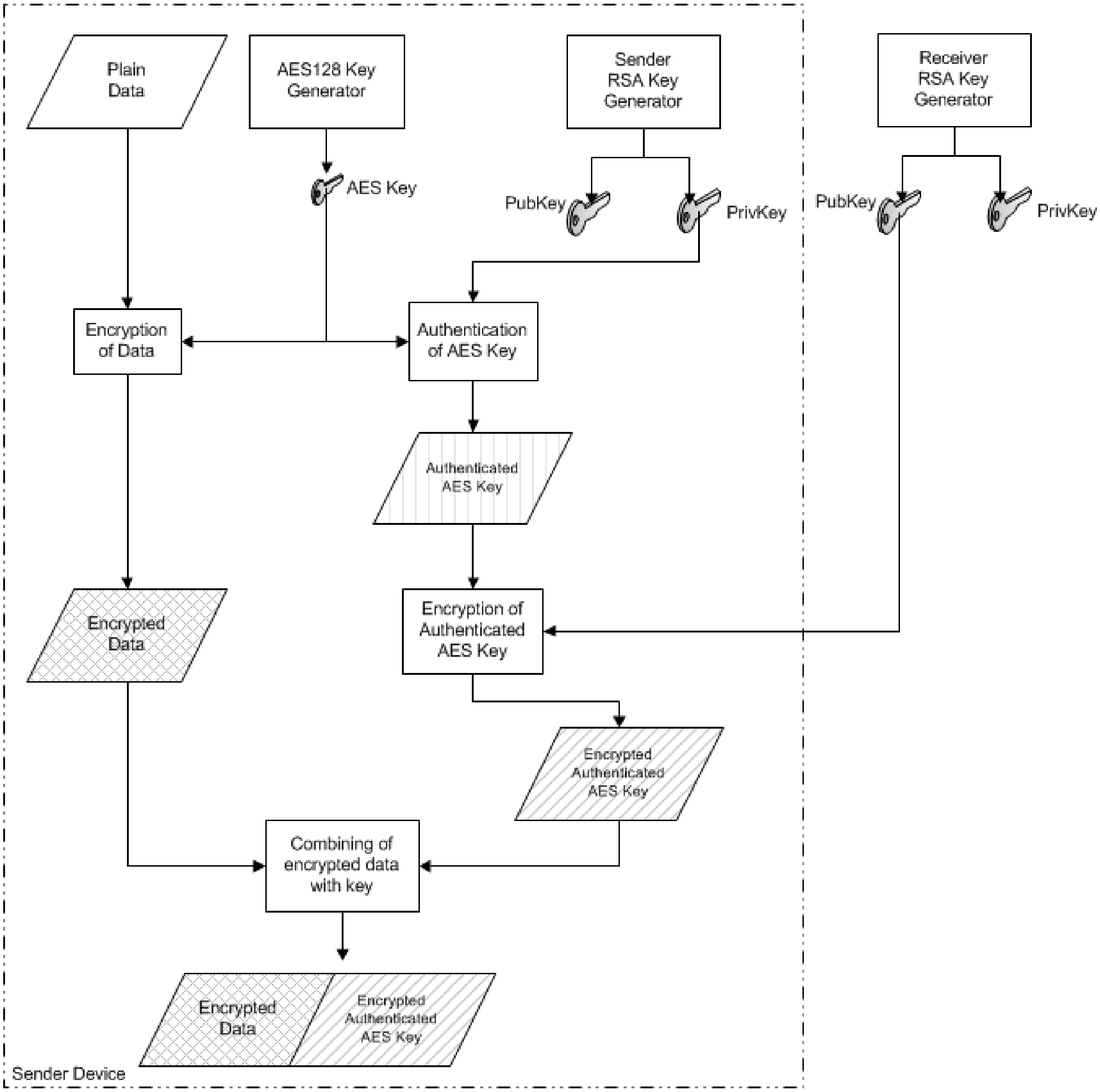
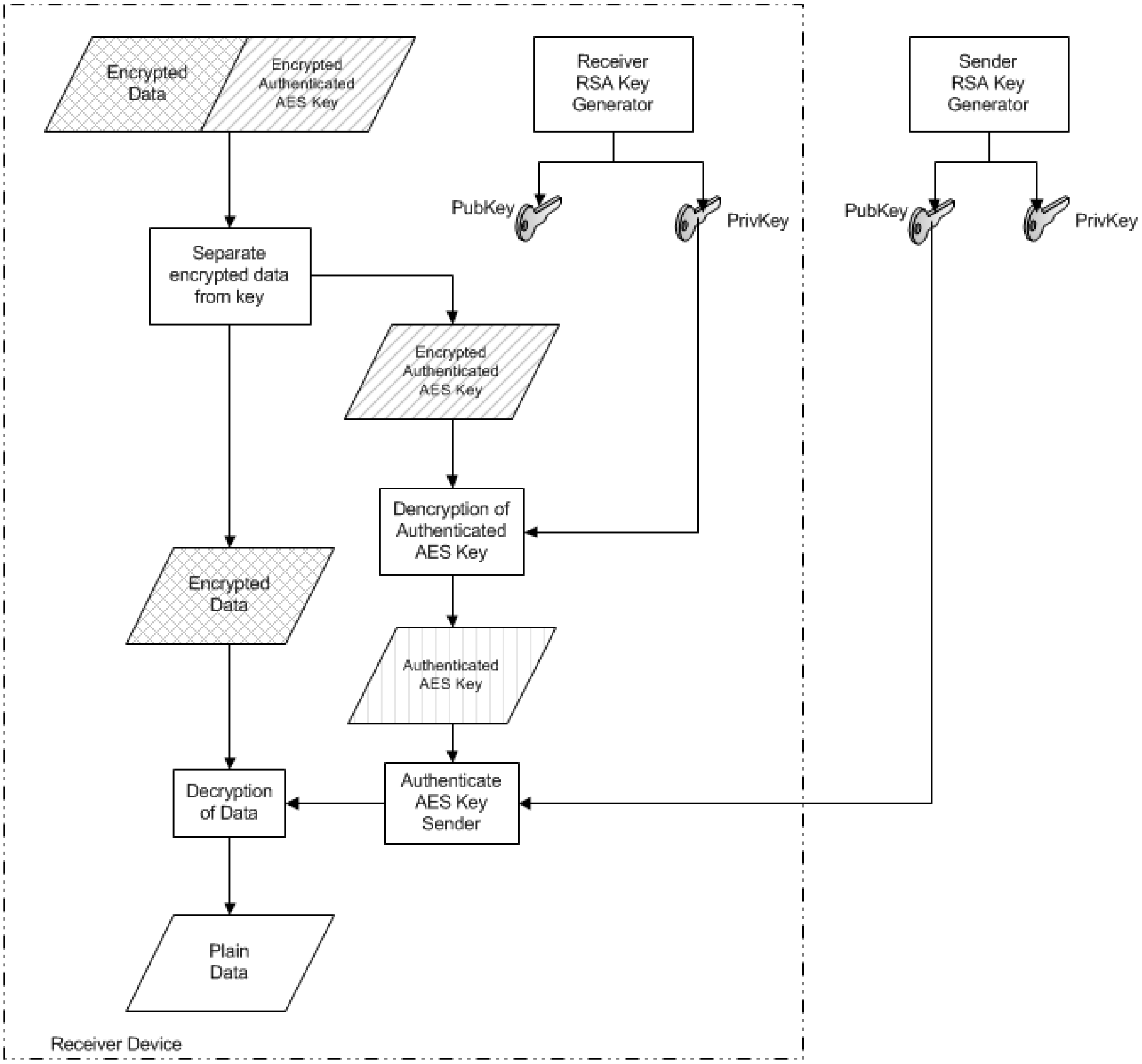

Plain Data
AES128 Key Generator
Sender RSA Key Generator
Receiver RSA Key Generator
AES Key
PubKey
PrivKey
PubKey
PrivKey
Encryption of Data
Authentication of AES Key
Authenticated AES Key
Encrypted Data
Encryption of Authenticated AES Key
Encrypted Authenticated AES Key
Combining of encrypted data with key
Encrypted Data
Encrypted Authenticated AES Key
Sender Device

Figure 2 : Sender Device

At the recipient's side:-

1. First the encrypted session key is received and decrypted using the sender's public key and the recipient's private key.

2. Then the decrypted session key is used by the block cipher to decrypt the message and get back the Plaintext.

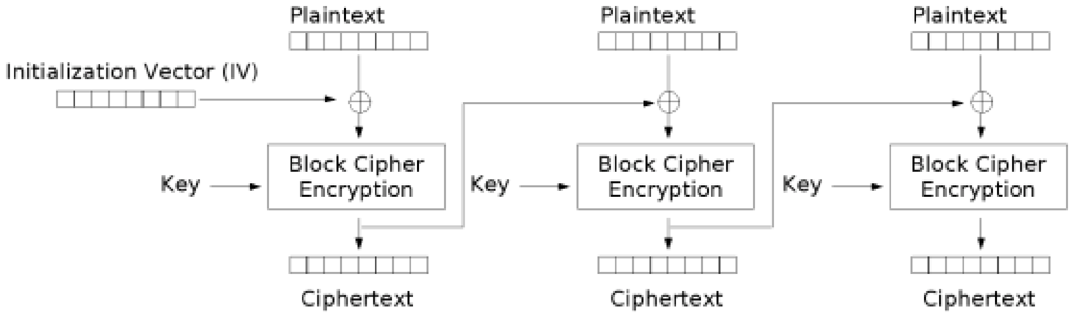


Encrypted Data
Encrypted Authenticated AES Key
Receiver RSA Key Generator
Sender RSA Key Generator
PubKey
PrivKey
PubKey
PrivKey
Separate encrypted data from key
Encrypted Authenticated AES Key
Dencryption of Authenticated AES Key
Encrypted Data
Authenticated AES Key
Decryption of Data
Authenticate AES Key Sender
Plain Data
Receiver Device

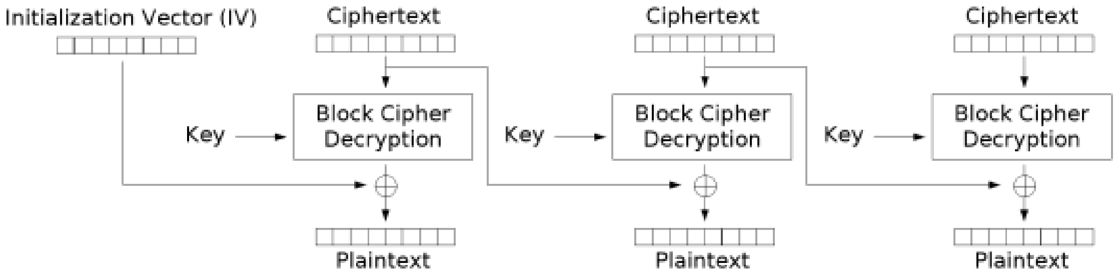
Figure 3 : Receiver Device

AES Encryption System

We chose for our system the AES-128 operating in the CBC (cipher-block chaining) mode for the symmetric-key encryption of the data and the RSA for the public-key encryption of the AES session keys.



Cipher Block Chaining (CBC) mode encryption



Cipher Block Chaining (CBC) mode decryption

Ideally the RSA keys are generated at each device and the public keys are broadcasted through a (network public-keys server), however in this project each communicating entity should request from the other entity its public key and the other entity should provide it whenever requested.

We will implement both AES encryption and decryption on hardware, as they are processing on the whole data, so encrypting it should be hardware accelerated, While keys encryption (The public-key cryptosystem "RSA") will be implemented on software as it requires high computational power so implementing it on hardware will use a huge resources however it is used one time per message, so implementing it on software would be more convenient.

The used communication link would be initially RS232 link, then we will try to provide to our system a RF and/or Ethernet drivers for more realistic communication links.

Work division and time plan

Week	Ahmed	Mohammed	Orges	
1	Searching + Brainstorming + Reading			
2	PPT + Proposal			
3	AES Encryption	Dummy components SW-HW Controller	Random number AES key generator	File operation: Read, Write, Padding, Segmentation
4	AES Encryption - Decryption	SW-HW Controller SW-HW interface	RSA Encryption + Decryption	
5	AES Decryption + HW integration	SW-HW integration	SW – integration	
6	Testing			
7	Presentation			