Technical status Zifra card

Torbjörn

May 28, 2018

Contents

Abstract

This document will summarize have bin done and what is needed to do to get to the final product.

1 todo

Here i will list stuff i come up whit so that i can write about them later.

- crypto scheem whit curv 25519
- Alex algorithm
- hiding files in fat
- fat format

2 Brief description

We will here shortly describe what the Zifra card is meant to be. It is an memory card that is supost to protect data by encrypting the data that is writen to it. This is done whit the help of multiple key(s) and cipher. i

3 System overview

We will here describe the way we want the system to work. Fore the case of explenation we will use an camera as example case. We will first talk about the key generation secondly encryption and finally about decryption step.

3.1 Key generation

There is three type of keys used in the crypto scheme. The first two keys are pre-generated as an key pair on an safe device one private key (Pri_key) and one Public key (Pub_key). The pri_key is saved fore later use and the pub_key is transfered and saved on the sd-card see Figure 1. Then the user puts the sd card in an camera. When the camera boots up an random number is generated this is the Random generated session Key (Rand_key) this number is then encrypted whit the Pub_key and saved on the memory.

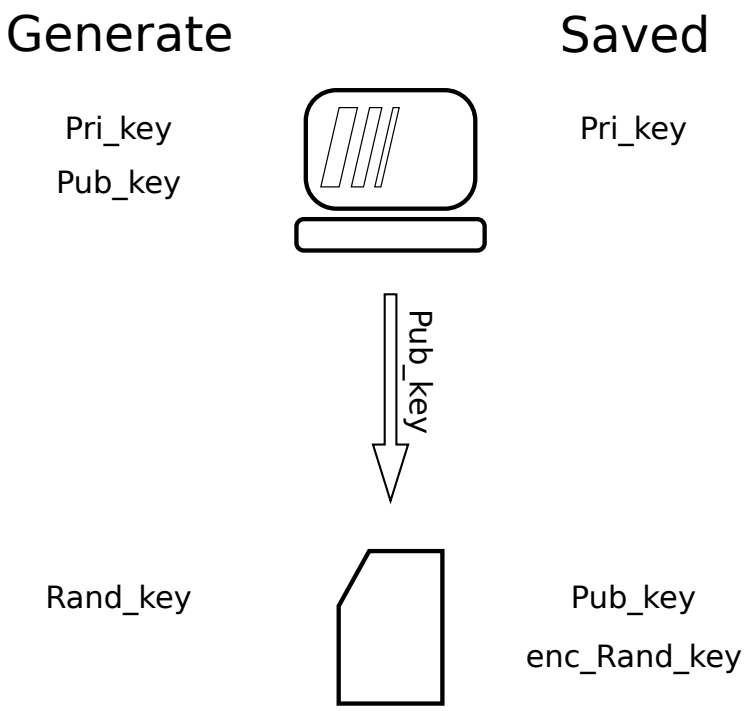

Figure 1: This illustration shows where the keys are generated and where they are stored

3.2 Encryption

The Rand_key[1] is then used as the key input in the chacha cipher to encrypt the data before it is saved to the memory se Figure 2. When the system is powered of the Rand_key that is only stored in the FPGA disappeared due to the nature of an FPGA. The generation and encryption of the Rand_key is repeated fore each session. So there will be multiple enc_Rand_key:s stored on the memory one fore each session.

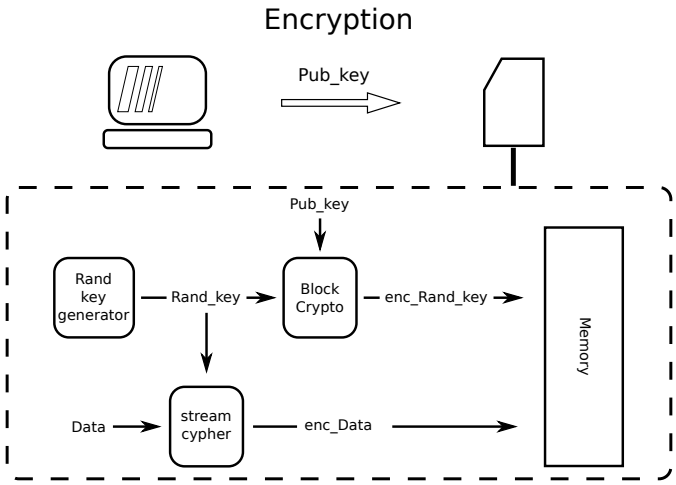

Figure 2: Basic data flow in encryption step

[1]This is the non encrypted random key.

3.3 Decryption

When decrypting the data one extracts the encrypted data and the Encrypted Random generated session Key (enc_Rand_key) on to the safe device were the Pri_key is saved. Then we use the Pri_key to decrypt the enc_Rand_key:s and then we can use the Rand_key:s to decrypt the data se Figure 3.

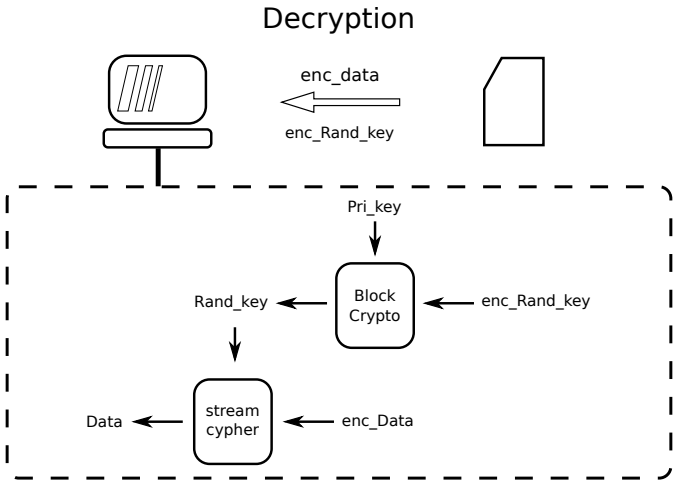

Figure 3: basic data flow in decryption step

4 Hardware

The hardware was intended to be an Printed Circuit Board (PCB) whit an Field Programmable Gate Array (FPGA) whit an memory and some suport component. The FPGA that we use is the Artix7 t35.

5 sd slave progress

The sd slave comes form the Google vault project [2]. This part of the project dose not work yet.

We have extracted the blocks that we think are needed fore the project. There test bench to test and send commands to the design and see what responses the device sends back. So fare we can read and write from the first sector in the memory but not to any other once .

need to bouble check this whit mesages from antti.

6 chacha

The stream cipher that was decided to use fore the data encryption ip called chacha [1] developed by Joachim Strömbergson [3]. We have made an axi ip block out of it one stream port fore the data and one address port fore the setting example iv and key se Figure 4. There is an test bench that show how the axi block is intended to bee used.

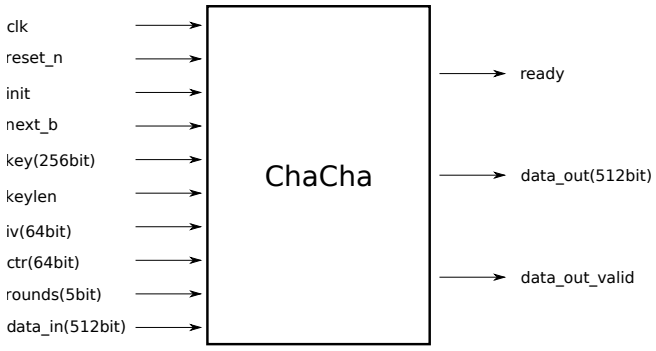

Figure 4: Port configuration chacha cipher

7 Helpful data

This section will list information that have bin collected during the project that might help further development.

7.1 CSD list

During the debugging we listed the CSD values that was used in the gv_sd_slave we also made an list of the other CSD values.

add the CSD tables here

8 Whishlist

Here we will list some features that we wold like to have in the future. Thees are extras but good to have.

8.1 Hidden files

We wold like to have the possibility to hide the files so that every time you start up the memory card it looks empty. This wold be don by making an temporary FAT table that is used during the session then hidden between the MBS and the real FAT table.

8.2 See pictures when camera is still on

We wold like to be able to look at the pictures during the session. This wold mean that the cipher needs to work in both directions during the session.

8.3 Fake pictures

We wold also like to be able to prep the memory card whit fake data in the camera case that wold be preloaded whit pictures. Thees wold show up if someone wold inspect the card. This wold also be helped by the see pictures while session is active if the system is tested.

Glossary

cipher Is an mathematical algorithm that takes two inputs data and an key and outputs an new data string that can not be read whit out an corresponding key. 3

gv_sd_slave This is the sd_slave from the google vault project. 7

key(s) This refers to an sires of bits used as an input to the cipher. 3

MBS Master boot record is an boot sector in the beginning of an memory that tells the system where to start. 7

Acronyms

enc_Rand_key Encrypted Random generated session Key. 6

FPGA Field Programmable Gate Array. 6

PCB Printed Circuit Board. 6

Pri_key private key. 3

Pub_key Public key. 3

Rand_key Random generated session Key. 3

References

[1] ChaCha. `https://git.cryptech.is/core/cipher/chacha.git`. Accessed: 2018-05-22.

[2] GoogleVault. `https://github.com/ProjectVault/orp`. Accessed: 2018-05-22.

[3] joachim strömbergson. Chacha (cypher).