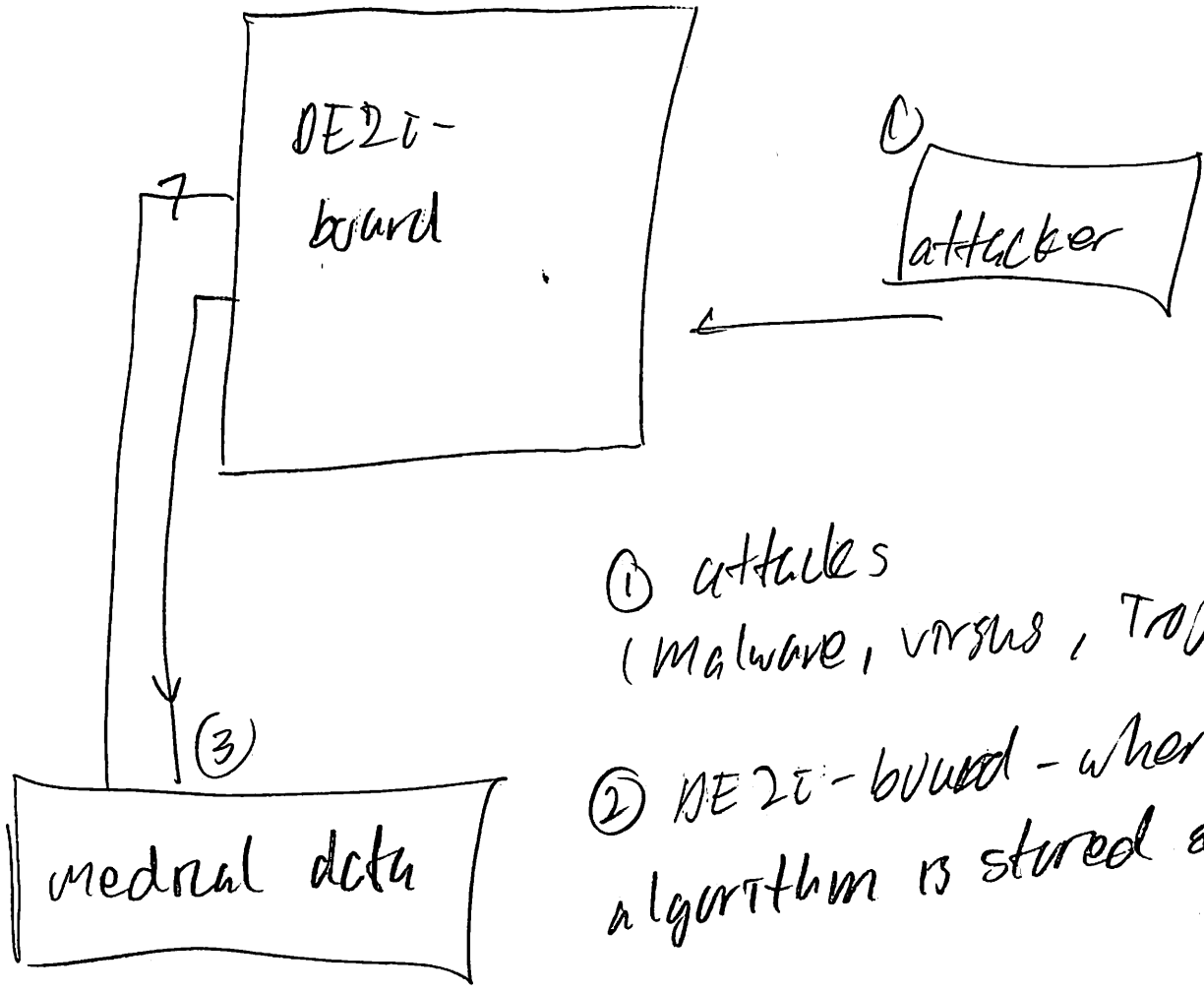
Jonathan Lopera

Project: To protect medical devices from attack that were aim to steal and change the information of patient's.

Need to do: implement cryptographic algorithms into a DE2i-150 board to see which algorithms are the best for securing data.

The algoridhm that I found out could be useful is the WHIRLPOOL cryptographic hash function.
I understand that this algorithm is more secure than the commonly employed hash functions. The reason is that the it have an increased message digest size. The algorithm was not patented by it's authors so it can use for free.

algorithm (WHIRLPOOL)



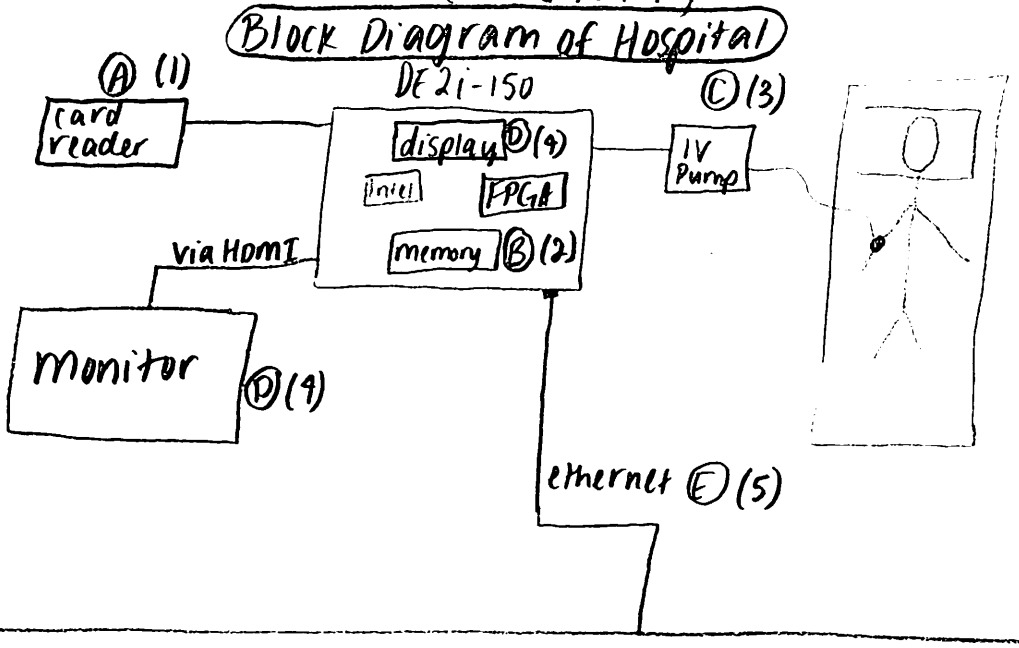
① attacks
(Malware, virus, Trojans)

② DE2i-board - where the algorithm is stored & used

③ medical data
- Patient's medical record.
- Doctor's prescription

Intruder Candace Ross (@02671771)

Block Diagram of Hospital



- We have 5 different cryptographic algorithms that can encrypt/decrypt data
- We also have different types of medical data that we need to protect in an efficient manner
- Different algorithms have different strengths and weaknesses; I propose using each algorithm in a different realm of medical security

A. Card Reader (1)

- identify medical personnel who are working with patient
- used for identification/ authenication purposes
- there is not a lot of info to confirm (probably just a password for the worker checked against info such as name, employee ID, etc); therefore, algorithm will not need to handle large amounts of data and can be small
- people also probably will frequently sign in/out (changing shifts, breaks, etc) therefore algorithm should be quick and lightweight
- suggestion: Hummingbird algorithm

→

B. Internal Memory (2)

- there is internal memory storing all of patients personal medical data (allergies, medication, social security num, etc.); this data is a hacker's goldmine and is crucial to protect
- a reliable, large robust algorithm is necessary
  - data must be quickly decrypted in case of emergency
  - sick patients will require a lot of data that needs to be encrypted so algorithm should be large
  - data should be easy to access/update as health improves or declines
- suggestion: RSA algorithm ⇒ reliable, widely-used

C. IV Pump (3)

- IV pump is actually distributing the fluids and medication
- similarly to memory, reliability is crucial
- task is relatively simple so large algorithm is probably not needed
- speed is also not crucial because changes will likely be infrequent and minor
- suggestion: simple, easy to implement small algorithm
- also, I would probably connect command to encrypt/decrypt this data i.e. make changes to card reader so only authorized users can make alterations

D. Display (external monitor and on-board) (4)

- this can be low-level encryption that displays information to employees who need updates while not being understood by outsiders
- display on external monitor or on LED/7-seg display on board
- cryptography here is key for privacy
- notifications would convey simple messages (such as fluid level low, bring food for dinner, etc.) so algorithm need not be too powerful →

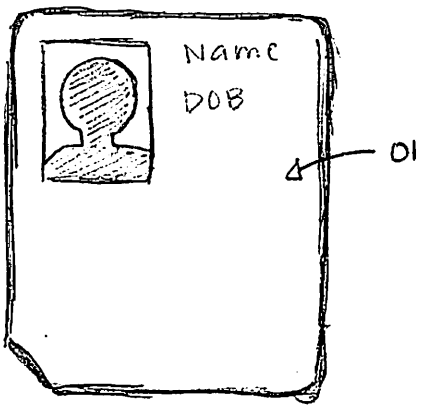
- algorithm only needs to encrypt data and display to screen in pre-determined fashion; decryption not necessary
- Suggestion: block or stream cipher (1-way encryption) private-key

E. Ethernet connection (5)
- DE2i-150 board will connect to Internet (preferably via Intranet, internal server) so patient info, employee log, etc. can be viewable from other locations
- because Internet is gateway to outside world of hackers and adversaries, strong encryption is paramount
- large amounts of data will be encrypted and sent as well as received and decrypted; need large algorithm
- method for specific encryption (e.g. cryptographic key used) should be able to be updated
- Suggestion: RC5 algorithm

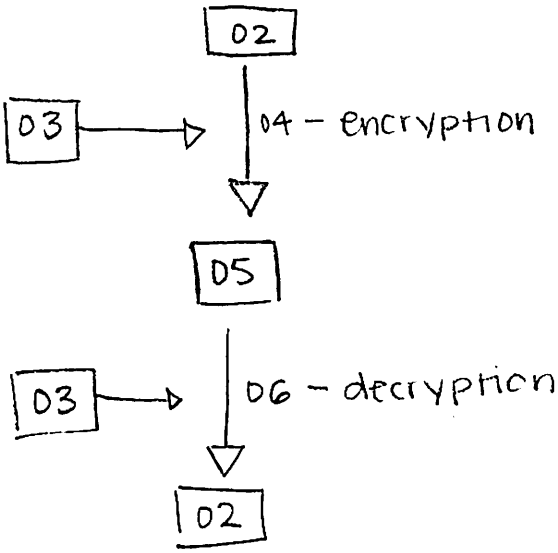
Individual Idea Generation

Naja J Green
@02665005
Intruder
Oct. 22, 2014

Implement Hummingbird cryptographic algorithm on FPGA to protect medical data. The algorithm should be the appropriate size, high performance, high speed and resilient to known cryptographic attacks.

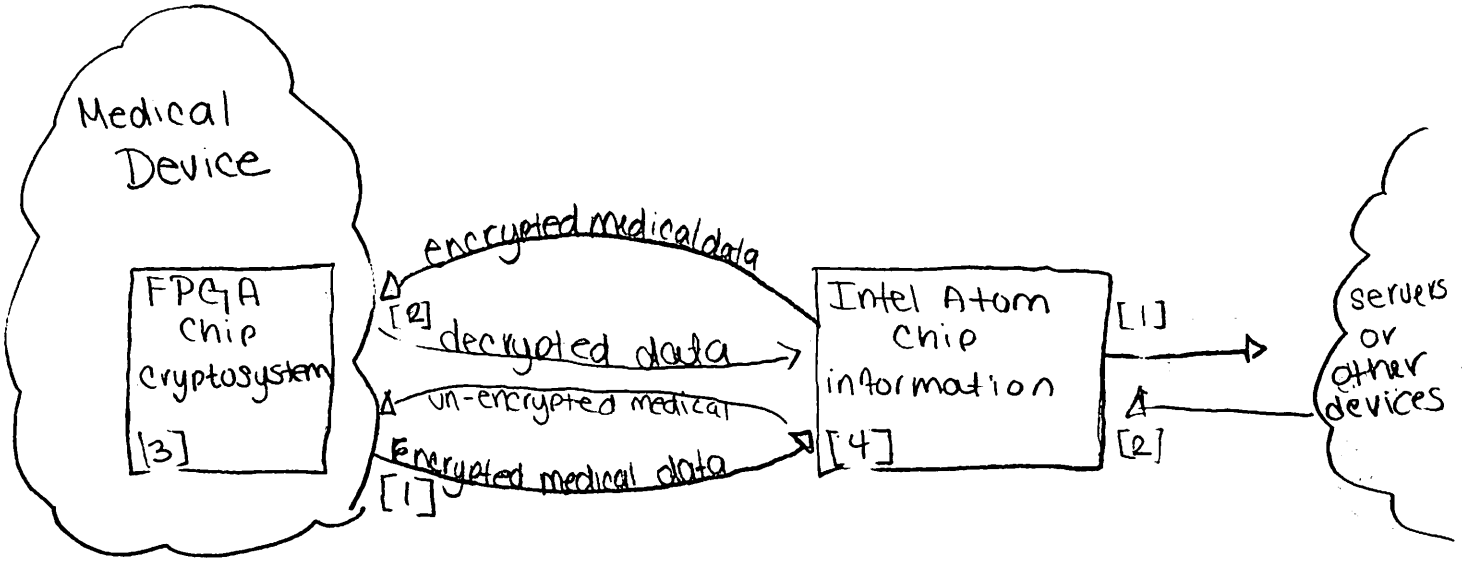


This particular algorithm is lightweight with small area requirement and low cost. It is currently used on smart cards; thus, we consider medical cards 01 that hold a patient's entire medical history to be protected by the Hummingbird algorithm.



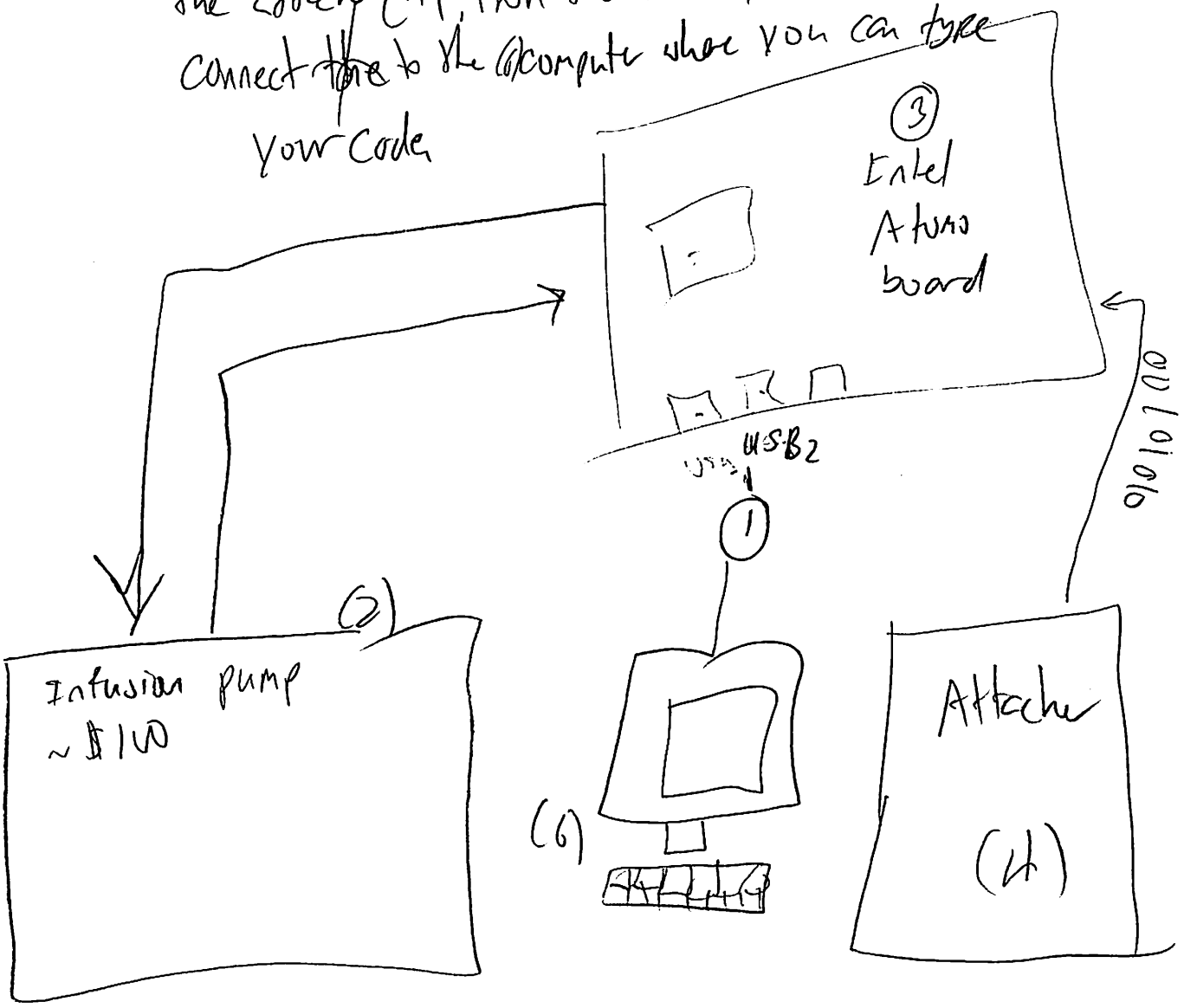
The secret key 03 is known only by necessary and authorized personnel, including the patient. The key is used to encrypt the readable medical data 02 using the Hummingbird encryption algorithm 04, this results in scrambled medical data 05 that is protected and irreversible on the card. When the information is needed, the Hummingbird decryption algorithm 06 is used with the secret key 03 to unscramble the data, resulting in readable medical data 02.

Jonretta M. Bratcher



The Intel Atom chip communicates with the database or other devices to send the medical information from a medical device that has the FPGA chip within it, that is equipped with the crypto system. [1] The FPGA encrypts the data and sends it to the Atom chip [4] which could then communicate with other devices or the database. In [2] the other devices can communicate with the Atom chip [4] which then sends the encrypted data to the FPGA which then decrypts it and sends it back. The FPGA chip [3] has a fully optimized crypto system that is size, power and performance-efficient in regards to the amount of data being processed

The (1) USB will be connected to the infusion pump (2). In inside the Intel Atoms board (3) there will be a microprocessor that will encrypt / decrypt the data. Then, we will design a security algorithm to protect the data from the attacker (4). From the USB (1) you will be able to connect it to the computer where you can type your code



Justin Powell
02662212

1. Study security and security primitives (hash, private key cryptography, public key cryptography, digital signature, random number generator.
2. Study and implement one cryptographic algorithm on FPGA
3. Optimize it in terms of power, performance and size
4. Study hardware Trojan attacks
5. Implement a couple of hardware