Software Requirements Specification

for

Hardware Security Module

requirements for
version 1.0

Prepared by MAHOHARD
software Inc.

Contents

1- Introduction

1-1- Purpose

This document includes software and hardware requirements for Hardware Security Module (HSM) release number 1.0. A hardware security module is a type of secure cryptoprocessor targeted at managing digital keys, accelerating cryptoprocesses in terms of digital signings/second and for providing strong authentication to access critical keys for server applications. They are physical devices that traditionally come in the form of a plug-in card or an external TCP/IP security device that can be attached directly to the server or general purpose computer. Hardware security module (HSM) is a useful tool to deploy Public Key Infrastructure (PKI) and its application. Within the context of this document, an HSM (or Hardware Security Module) is defined as a piece of hardware and associated software/firmware that usually attached to the inside of a PC or server and provides at least the minimum of cryptographic functions. These functions include (but are not limited to) encryption, decryption, key generation and hashing. The physical device offers some level of physical tamper – resistance and has a user interface and a programmable interface.

1-2- Document Conventions

When writing this document it was inherited that all requirements have the same priority.
First there is presented an overall view about HSM and then all features and functions are analyzed in detail.

1-3- Intended Audience and Reading Suggestions

This requirement document contains general information about HSM, marketing researches, hardware requirements, features and special technologies. Because this project is a low level embedded project, it is difficult to give brief explanations about high level software requirements. This report actually contains main overview of HSM and diagrams about main parts of HSM.

This document is intended for

Developers: In order to be sure they are developing the right project that fulfills requirements provided in this document.

Testers: In order to have an exact list of the features and functions that have to respond according to

requirements and provided diagrams.

Users: In order to get familiar with the idea of the project and suggest other features that would make it even more functional.

1-4- Project Scope

Hardware Security Module (HSM) is simply a secure communication module. As it is already known, these kind of applications can be done by using pure software algorithms, but when HSMs are involved, a lot of advantages can be obtained like,

- More trusted than software based security
- Low cost on analogous-digital system conversions.
- Higher performance than the software security algorithms –because of the parallel programming-

Also, HSM production and usage is regulated by various institutions. This means, the producers shall comply with these security and production regulations in order to enter the market. With the help of these regulations (e.g. FIPS, NIST, DES) clients can obtain secure and trusted modules.

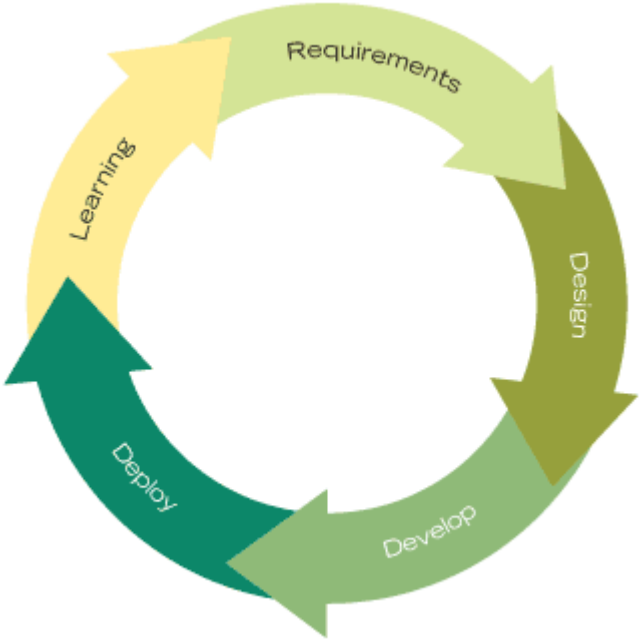
More about HSM you can visit http://www.cren.net/crenca/onepagers/hsm2.html

2- Process

2-1- Process Model

As MahoHard we prefer rapid application model for our project development. We have some reasons to choose this model. The main reason of why we have chosen this model is that RAD uses minimal planning in favor of rapid prototyping. The "planning" of software developed using RAD is interleaved

with writing the software itself. The lack of extensive pre-planning generally allows software to be written much faster, and makes it easier to change requirements.



2-2- Team Organization

In our team, Ali Cander is responsible for coordinating team members and assigning tasks to each member. Ali is also contact person of our group. Team members will not be strict, especially considering our process model, no one will be specialized on one single concept. In other words everyone will be, in general perspective, aware of what the other members are working on. We agreed on this structure because design is the most problematic issue in such a model.

3- Research

In order to determine the most suitable tools, hardware sources and necessary information related to HSM project we made an all around research.

3-1- Hardware Research

Since this project is low level embedded project, hardware components are highly important.

3-1-1- Altium Nanoboard

Altium's NanoBoards are unique, reconfigurable hardware platforms that harnesses the power of today's high-capacity, low-cost programmable devices to allow rapid and interactive implementation and debugging of your digital designs.

With a NanoBoard connected to a PC running Altium Designer – the world's first truly unified electronic design system – you can develop, implement, test and bring to market more intelligent digital products faster than previously possible.

Use a NanoBoard as a system prototyping and development platform, as an educational hands-on teaching tool, or as a standalone product running a custom system design. Its power, flexibility and tight integration with Altium Designer opens up the exciting possibilities of soft system design to you today

For more info : http://wiki.altium.com/display/ADOH/NanoBoards

3-1-1-1 Nanoboard 3000

Part of Altium's growing family of Nanoboard, the 3000-series NanoBoards provide the perfect entry-point to discover and explore the world of soft design in a low-cost, fun way. In true NanoBoard style, each board in the series offers a reprogrammable hardware platform that harnesses the power of a high-capacity, low-cost programmable device to allow rapid and interactive implementation and debugging of your digital designs.

The NanoBoard 3000 provides a fixed User FPGA that is located on the motherboard itself and provision for the attachment of a single peripheral board.

3-1-1-2 Nanoboard NB2DSK01

Altium's Desktop NanoBoard NB2DSK01 is a unique, reconfigurable hardware platform that harnesses the power of today's high-capacity, low-cost programmable devices to allow rapid and interactive implementation and debugging of your digital designs.

The Desktop NanoBoard NB2DSK01 takes the nano-level breadboarding concept introduced with the NanoBoard-NB1 to a whole new level, enhancing your ability to design, implement and debug an entire design before moving to the production PCB. Swappable daughter boards now support a much larger number of I/O connections from the target FPGA to the connected peripherals.

With the Desktop NanoBoard NB2DSK01, peripherals available for the daughter board FPGA are delivered on removable peripheral boards, providing a simple and cost-effective method for rapid prototyping of hardware concepts. The Desktop NanoBoard NB2DSK01 is designed to be a perfect complement to Altium Designer, the unified electronic product development system that transforms your desktop into a complete and interactive electronics design laboratory that uses LiveDesign.

3-1-2 FPGA Boards

There is 3-connector daughter boards available for use with the NanoBoard and additional boards. The following is a listing of the currently available daughter boards, grouped by vendor of the physical FPGA device provided.

3-1-2-1 Xilinx Spartan-3AN Daughter Board

- Xilinx Spartan-3AN FPGA (XC3S1400AN-4FGG676C)
- On-board memories available for use by FPGA design:
    - 256K x 32-bit common-bus SRAM (1MByte)
    - 16M x 32-bit common-bus SDRAM (64MByte)
    - 16M x 16-bit common-bus Flash memory (32MByte)
    - Dual 256K x 16-bit independent SRAM (512KByte each)

3-1-2-2 Altera Cyclone III Daughter Board

- Altera Cyclone III FPGA (EP3C40F780C8N)
- On-board memories available for use by FPGA design:
    - 256K x 32-bit common-bus SRAM (1MByte)
    - 16M x 32-bit common-bus SDRAM (64MByte)
    - 16M x 16-bit common-bus Flash memory (32MByte)
    - Dual 256K x 16-bit independent SRAM (512KByte each)

3-1-2-3 LatticeECP2 Daughter Board

- LatticeECP FPGA (LFECP33E-3FN672C)
- On-board memories available for use by FPGA design:
    - 256K x 32-bit common-bus SRAM (1MByte)
    - 16M x 32-bit common-bus SDRAM (64MByte)
    - 16M x 16-bit common-bus Flash memory (32MByte)
    - Dual 256K x 16-bit independent SRAM (512KByte each)

3-2 Software Research

3-2-1 Hashing Algorithms

3-2-1-1 MD5

In cryptography, **MD5** (**Message-Digest algorithm 5**) is a widely used cryptographic hash function with a 128-bit hash value. As an Internet standard (RFC 1321), MD5 has been employed in a wide variety of security applications, and is also commonly used to check the integrity of files. However, it has been shown that MD5 is not collision resistant; as such, MD5 is not suitable for applications like SSL certificates or digital signatures that rely on this property. An MD5 hash is typically expressed as a 32-digit hexadecimal number.

3-2-1-1 SHA

The **SHA hash functions** are a set of cryptographic hash functions designed by the National Security Agency (NSA) and published by the NIST as a U.S. Federal Information Processing Standard. SHA stands for **Secure Hash Algorithm**. The three SHA algorithms are structured differently and are distinguished as *SHA-0*, *SHA-1*, and *SHA-2*. The *SHA-2* family uses an identical algorithm with a variable digest size which is distinguished as *SHA-224*, *SHA-256*, *SHA-384*, and *SHA-512*.

SHA-1 is the best established of the existing SHA hash functions, and is employed in several widely used security applications and protocols. In 2005, security flaws were identified in SHA-1, namely that a possible mathematical weakness might exist, indicating that a stronger hash function would be desirable.Although no attacks have yet been reported on the SHA-2 variants, they are algorithmically similar to SHA-1 and so efforts are underway to develop improved alternatives. A new hash standard, SHA-3, is currently under development – the function will be selected via an open competition running between fall 2008 and 2012.

3-2-2 Encryption Algorithms

3-2-2-1 DES

The **Data Encryption Standard** (**DES**) is a block cipher (a form of shared secret encryption) that was selected by the National Bureau of Standards as an official Federal Information Processing Standard (FIPS) for the United States in 1976 and which has subsequently enjoyed widespread use internationally. It is based on a symmetric-key algorithm that uses a 56-bit key. The algorithm was initially controversial with classified design elements, a relatively short key length, and suspicions about a National Security Agency (NSA) backdoor. DES consequently came under intense academic scrutiny which motivated the modern understanding of block ciphers and their cryptanalysis.

DES is now considered to be insecure for many applications. This is chiefly due to the 56-bit key size being too small; in January, 1999, distributed.net and the Electronic Frontier Foundation collaborated to publicly break a DES key in 22 hours and 15 minutes (see chronology). There are also some analytical results which demonstrate theoretical weaknesses in the cipher, although they are unfeasible to mount in practice.

3-2-2-2 Triple DES

Triple DES is based on the DES (Data Encryption Standard) algorithm, therefore it is very easy to modify existing software to use Triple DES. It also has the advantage of proven reliability and a longer key length that eliminates many of the attacks that can be used to reduce the amount of time it takes to break DES. However, even this more powerful version of DES may not be strong enough to protect data for very much longer. The DES algorithm itself has become obsolete and is in need of replacement. Triple DES will be kept around for compatibility reasons for many years after that..

3-2-2-3 AES

The AES will be at least as strong as Triple DES and probably much faster. Many security systems will probably use both Triple DES and AES for at least the next five years. After that, AES may supplant Triple DES as the default algorithm on most systems if it lives up to its expectations. But Triple DES will be kept around for compatibility reasons for many years after that. So the useful lifetime of Triple

DES is far from over, even with the AES near completion. For the foreseeable future Triple DES is an excellent and reliable choice for the security needs of highly sensitive information.

3-2-3 Key Generation Algorithms

3-2-3-1 RSA

In cryptography, **RSA** is an algorithm for public-key cryptography. It is the first algorithm known to be suitable for signing as well as encryption, and was one of the first great advances in public key cryptography. RSA is a public-key algorithm used ubiquitously in Internet and electronic communication security, as an important part of document authentication and user identity verification. It is present as a key component of many e-commerce and email security systems, VPNs, security suites, as well as in most popular security protocols.

3-2-3-2 DSA

The Digital Signature Standard, created by the NIST, specifies DSA as the algorithm for digital signatures like RSA. The DSA is used by a signatory to generate a digital signature on data and by a verifier to verify the authenticity of the signature. When a message is received, the recipient may need to verify that the message has not been altered in transit. Furthermore, the recipient may wish to be certain of the originator's identity. Both of these services can be provided by the Digital Signature Algorithm (DSA). A digital signature is an electronic version of a written signature in that the digital signature can be used in proving to the recipient or a third party that the message was, in fact, signed by the originator. Digital signatures may also be generated for stored data and programs so that the integrity of the data and programs may be verified at any later time.

3-3- Marketing Research

3-3-1-Thales Group

3-3-1-1 Thales nShield Connect 6000

Thales nShield Connect 6000, part of the nCipher product line, is a network-attached, general-purpose hardware security module (HSM) that protects up to 100 clients by safeguarding their

encryption and digital signing keys and processing sensitive data on the trusted appliance.

3-3-1-2 Thales Nethsm

The netHSM can form the core of a wide-reaching security strategy. Once you've added it to your network, it provides encryption and digital signing services for your most sensitive systems and applications.

3-3-1-3 Thales Nshield

nShield protects encryption keys on servers in a highly secure, tamper-resistant hardware environment. A dedicated module installed on a single server, it provides encryption and signing for a specific application.

3-3-1-4 HSM 8000

Easy to customise and capable of delivering unrivalled protection for ATM, POS, corporate banking, card issuing, funds transfer, and share trading technology, the HSM 8000 is a tamper-resistant device that provides the powerful cryptographic facilities that are needed to secure financial transactions.

3-3-2- Lightsource Technology

3-3-2-1 LunaPCI HSM

Luna PCI is a FIPS validated hardware security module designed to protect cryptographic keys and accelerate sensitive cryptographic operations across a wide range of security applications. Luna PCI offers dedicated hardware-based key management to protect sensitive cryptographic keys from attack. The high-security hardware design ensures the integrity and protection of encryption keys throughout their life cycle. All digital signing and verification operations are performed within the HSM to increase performance and maintain security. Luna PCI HSMs provide hardware-secured key generation, storage, secure key backup and accelerated encryption in a range of models and configurations offering a wide selection of security, performance and operational capabilities.

3-3-2-2 LunaSP HSM

Luna SP's HSM offers hardware key management and ensures that cryptographic keys and processes are stored and managed exclusively within FIPS validated hardware. Code signing and verification maintain the integrity of custom Java application code and prevent unauthorised application execution. Additionally, strictly enforced access and usage policies prevent unauthorised access to sensitive applications or data. With tamper-resistant hardware, network connectivity, and secure remote administration, Luna SP makes it easy to deploy high-assurance Java Web service applications with confidence.

3-3-2-3 ProtectServerBlue HSM

ProtectServer Blue is typically employed to provide cryptographic services such as high-speed bulk encryption, user and data authentication, message integrity, secure key storage and management for E-commerce, PKI and Financial Security applications.

3-3-3- Digital Signature Company

3-3-3-1 PrivateServer HSM

PrivateServer conducts sensitive cryptographic operations, secure key storage, and management of a large number of keys. Due to Flexible and highly secure infrastructure, PrivateServer enables customers and partners to develop their own custom modules using .NET programming languages that are executed inside the HSM.

3-3-3 SPYRUS Group

3-3-4-1 LYNKS Series 2 Hardware Security MOdules

The LYNKS Series II Hardware Security Module (HSM) offers a high-security solution for client, server and embedded security applications. The LYNKS Series II HSM, with upgraded flash memory and FPGA capabilities, supports new, stronger and faster cryptographic algorithms, including elliptic curve cryptography with EC-DH and ECMQV key establishment, AES, and the SHA-2 algorithms that exceed the U.S. Government's Suite B standard. Available

in either PCMCIA or stackable USB models, the new LYNKS Series II HSM provides the strongest and most economical future-proof protection for sensitive data.

4- Overall Description

4-1- Product Perspective

HSM consists of mainly four parts: A Microcontroller, PKI support (Cryptographic Algorithm modules), Host Interface and Secure Storage. Microcontroller is built on FPGA. On the microcontroller, there will be a real time operating system that will manage all the operations in the HSM. The operating system provides resource management and task scheduling. The PKI support area provides overall system management function including status operations and self-tests of the cryptographic components. All of the cipher and hashing mechanisms are found here. Cryptographic Algorithm modules are also developed on the FPGA board. The aim of this part is to encrypt data, hashing data and key generation for user. The host interface is responsible for accepting request packets from a server (it performs no cryptographic processing). Secure storage is used to keep data such as public key, private key and hash values.

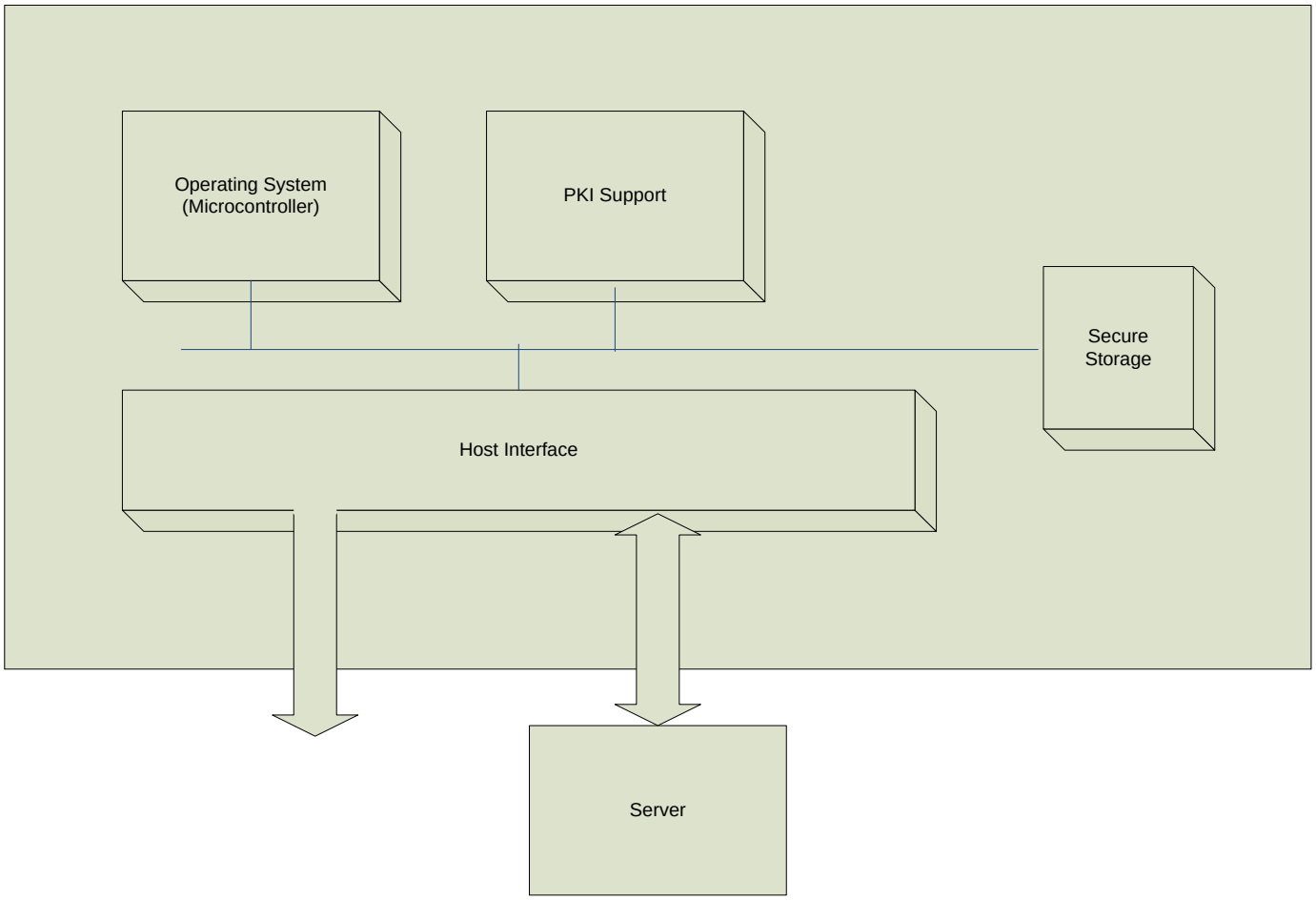


Figure1: HSM Overview

4-2- Product Features

An HSM can perform a number of important security-related functions. It provides accelerated cryptographic operations such as encryption, digital signatures, hashing and Message Authentication Code (MAC). MAC is an algorithm that mathematically combines a key with a hash to provide a "code" that can be appended with a given piece of data to ensure its integrity.

- **Digital Signature Generator**
  HSM will generate a digital signature for users. For this generation RSA and DSA are widely accepted algorithms.
- **Random Number Generation**

Random number generation (RNG) or pseudo-random number generation is critical to many cryptographic functions including key generation.

- **Encrypt Data**

  HSM will encrypt data for secure data transfer. Transferring data securely is extremely important. In order to encrypt data, there are widely accepted algorithms that can be used. 3 – DES is an example of encryption algorithms.

- **Hash Data**

  In order to access to a data into secure storage and apply some operations on it, we will use hash function(s). SHA-1 and MD5 are examples of hashing algorithms.

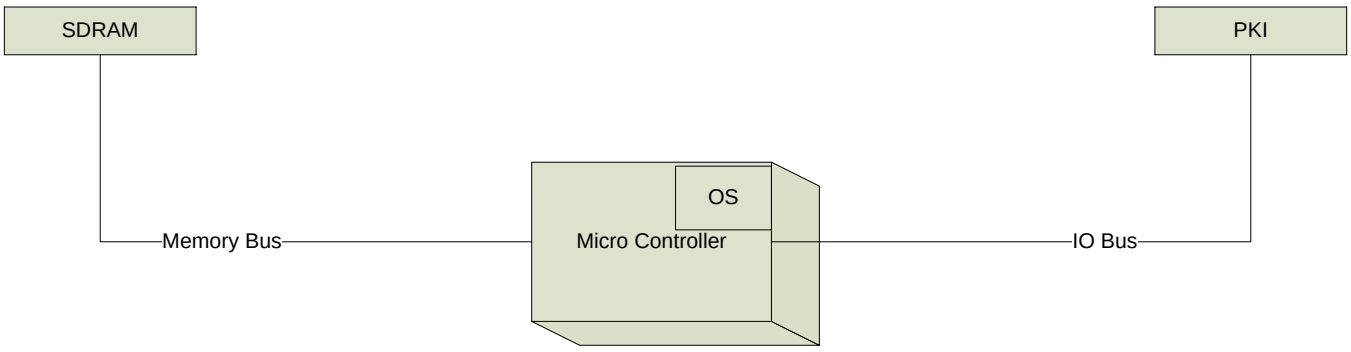


Figure 2: MicroController connections

Another feature of HSM is "**TRUST CHANNEL**". HSM will implement Trusted Channel which enables operators to securely communicate data sent over the Ethernet interface.

4-3- Operating Environment

There will be an embedded real time operating system on FPGA which is part of HSM. It is not important whether it is Linux based or not. The important thing is that, whether this real time operating system is the most helpful one or not.

4-4- Design and Implementation Constraints

- For encryption:

DES (Data Encryption Standard)

Successors : Triple DES, G-DES, DES-X, LOKI89, ICE

Cipher detail

Key sizes : 56 bits

Block sizes : 64 bits

Structure : Balanced

Rounds : 16

AES (Advanced Encryption Standard)

Successors : Anubis, Grand Cru

Certification : AES winner, CRYPTREC, NESSIE, NSA

Cipher detail

Key sizes :128, 192 or 256 bits

Block sizes : 128 bits

Structure : Substitution-permutation network

Rounds : 10, 12 or 14 (depending on key size).

- For Hashing:

MD5

Block sizes: 128 bits.

Round : 4

SHA-1

Block Size: 512 or 1024 bits

Round: 64 or 80

- For Digital Signatures:

PKCS#11 (Public Key Cryptography Standard)

An API defining a generic interface to HSM. Often used in single sign-on, Public-key cryptography and disk encryption systems.

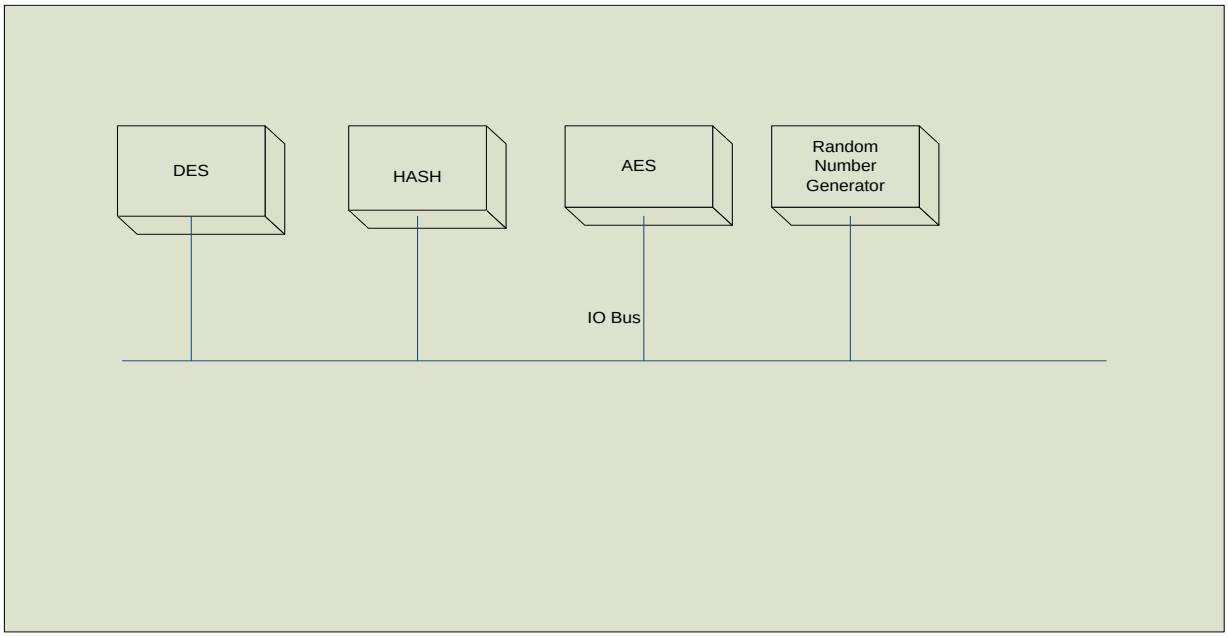


Figure 3: PKI block diagram

5- System Features

In HSM system, server continuously sends requests to HSM and HSM respond to these requests with secure. For there are a lot of requests from server, HSM should respond to requests very quickly. For this purpose, microcontroller and other FPGA modules should be highly optimized. In this stage of process, it is not possible to decide how software progress will take form but there is explanation about well-known algorithm which is used on FPGA modules for encryption, hashing and key generation.

5-1- Hash Data

5-1-1- Stimulus/Response Sequences

Data Flow

5-1-1-1 Basic Data Flow

1- Data which will be hashed comes to HSM

2- Data is sent to a suitable FPGA module by microcontroller.

3- Hashing algorithm is applied.

4- Hashed data is sent to suitable place according to server's demand.

5-2- Encrypt Data

5-2-1- Stimulus/Response Sequences

Data Flow

5-2-1-1 Basic Data Flow

1- Data which will be encrypted comes to HSM

2- Data is sent to a suitable FPGA module by microcontroller.

3- Encryption algorithm(s) is applied.

4- Encrypted data is sent to place according to server's demand.

5-3-Random Number Generator

5-3-1- Stimulus/Response Sequences

Data Flow

5-3-1-1 Basic Data Flow

1- Random number will be generated when there is a need to this number by microcontroller. (it will be used for encryption and hashing)

2 – Generated number is sent to a suitable FPGA module which this number will be used by.

5-4-Digital Signature Generator

5-4-1- Stimulus/Response Sequences

Data Flow

5-4-1-1 Basic Data Flow

1- Data is sent to HSM

2-Data is sent to a suitable FPGA module by microcontroller.

3- Digital Signature Generator algorithm is applied.

4- Unique digital signature is sent to place according to server's demand.

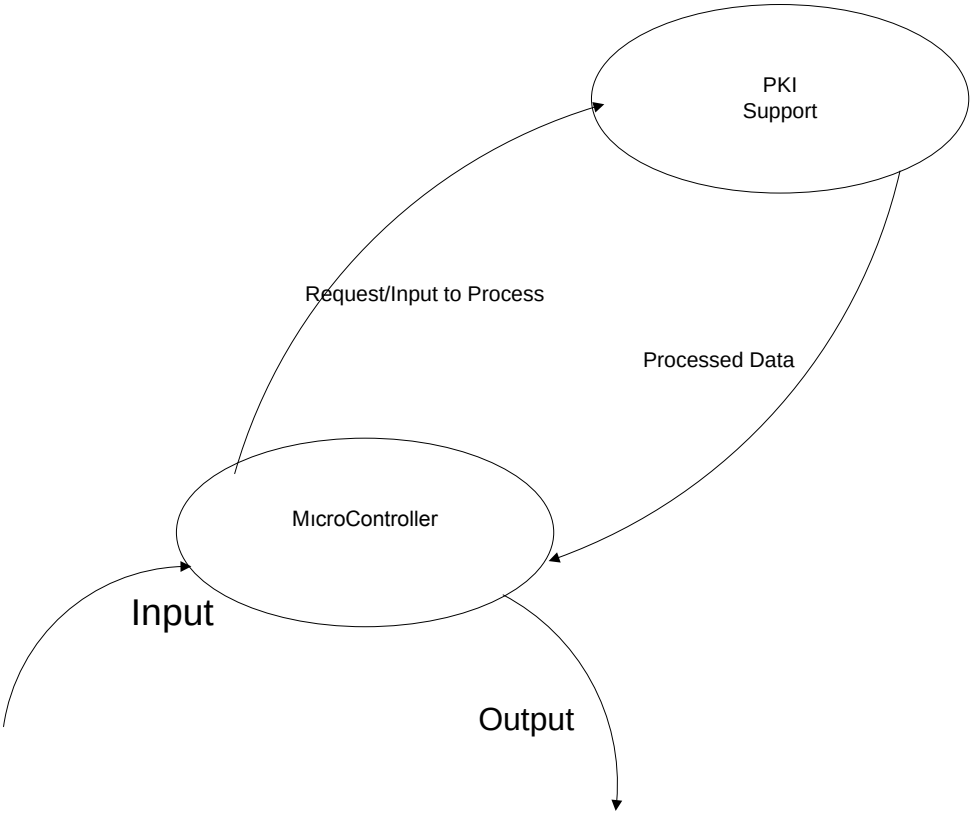
6- Other Nonfunctional Requirements

6-1- Hardware Requirements

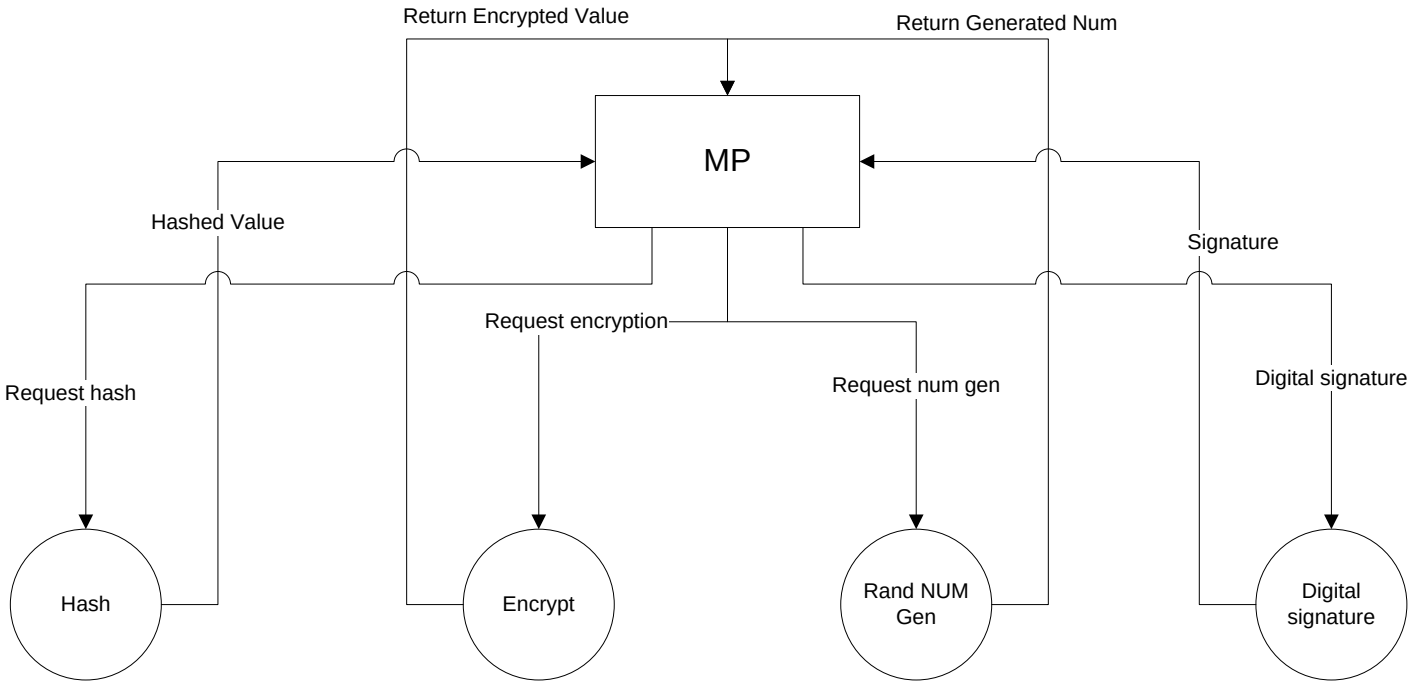
- Altium Nanoboard 3000 (Reprogrammable Hardware Development Platform)
- FPGA (Spartan 3-AN) (XC3S1400AN)
- Smart Card reader (flexible)

7- Data Flow Diagrams

7-1 Level 0 DFD



7-2 Level 1 DFD



8- References

- ***http://resources.bnet.com/topic/hsm.html***
- ***http://iss.thalesgroup.com/en/Products/Hardware%20Security%20Modules/HSM%208000.aspx***
- ***http://iss.thalesgroup.com/en/Products/Hardware%20Security%20Modules/HSM%208000.aspx***
- ***http://www.spyrus.com/products/lynks_hardware.asp***
- ***http://en.wikipedia.org/wiki/Data_Encryption_Standard***
- ***http://en.wikipedia.org/wiki/Data_Encryption_Standard***
- ***http://www.vocal.com/cryptography***
- ***www.doegrids.org/Docs/ESnetHardwareSecurityModuleEvaluation.pdf***
- ***www.iis.se/docs/hsm-20090529.pdf***