The study of the HSM as a solution to file encryption and security

Gliqiri Riza

University of Tirana, Faculty of Economics, Tirana, Albania

Abstract

Nowadays with the development of technology and the fact that it has influenced every aspect of people's lives, the information is the most essential factor to be considered, since it is the fundamental in which technology is being built. On the contrary, cybersecurity attacks are growing rapidly year after year since everyone around the world is seriously involved in this wild race of having access to the information owned by the others. Data breaches are also growing since hackers and non-authorized parties are attacking organizations, companies and other institutions of high importance. As a result, more and more companies are considering solutions that not only will protect the information, but will also keep un-authorized parties away from having access to it. Encryption is considered one of the most classic ways to ensure that sensitive information is converted into secret codes that will hide the information from un-authorized access. But what's the point to encrypting the information, when the keys are not totally safe and are at risk of being exposed. Cryptographical keys should be stored, kept and managed carefully, since their exposure would disclose the sensitive data the same way it as not even encrypted. That is the reason why companies, organizations and institutions globally are using Hardware Security Modules as the solution to secure and protect their sensitive data and to provide functionality for their cyber security purposes, including authentication, authorization, data confidentiality and data integrity. This paper aims to give a brief analysis of what a HSM is, how it functions and how it helps the companies and organizations in the context of cybersecurity. Furthermore, it will provide information on types of HSMs, architecture, different options offered on the market and most common use cases inthe recent years.

Keywords

Encryption, HardwareSecurityModule, cryptography, randomness, PKI

1. Introduction

While there are a lot of measures widely used to protect sensitive data and keep the un-authorized parties away from having access to certain information, encryption is one of the most used and well-known methods to hide data's true meaning behind secret codes, in such a way that even if someone has access, he/she can't understand the content without having the proper key to decrypt it. On the other hand, encryption has its own disadvantages because key management brings potential security vulnerability since hackers can steal encryption keys and with the stolen key it is quite easy to decrypt the information and have access to sensitive and supposedly "protected" data. Thus, in order to ensure the security, during the recent years it is being used a key protection solution based on the hardware component of an information system called HSM.

HSM, which stands for the Hardware Security Module is a physical computing device which can provide functionalities for cybersecurity purposes, including authentication, authorization, data confidentiality and data integrity [1].

The Hardware Security Modules can be used in a wide variety of platforms and information systems environments, where cryptographic functionalities are used to offer a wider spectrum of protection and security. There are different types of HMSs, such as stand-alone devices, plug-in cards or even Hardware Security Modules that are built into other devices. Recently, the HSMs are well-known for being used in cloud infrastructure, web servers, card payment systems, banking and similar fields where important cryptographic operations such as key generation, encryption, decryption, signature generation and hash functions are combined with hardware components to ensure better data protection [2].

Proceedings of RTA-CSIT 2023, April 2023, Tirana, Albania
EMAIL: gliqeririza@hotmail.com

There are two main functions of a Hardware Security Module:

1. First of all, one of the most important factors in using a HSM is that it provides secure management of cryptographic keys used in encryption process. This includes safe storage of keys, that is made possible by keeping and encapsulating the keys in an isolated environment which has dedicated mechanisms that prevent tampering [3].

2. On the other hand, another important function of an HSM is also to provide hardware acceleration for various cryptographic operations in order to increase the speed by which they are performed.

During the recent years, when the development of technology has entirely changed every aspect of people's lives, they are all being exposed to hackers, who are more and more exploiting vulnerabilities of people, companies or systems to profit more. It is now a wild race not only to have access to other's personal data, but also to sensitive or confidential information being processed in the systems that are being used by companies, hospitals and governmental institutions. As a result, since cryptography is always necessary when having something to protect, like information or money, HSMs combine additional hardware features, to ensure a more secure background for critical information and financial assets. Thus, considering that cryptography is being used to protect secret data, the most important components that should be taken special care of are the encryption and decryption keys used in the process, in order to be kept secret from adversary and also to ensure confidentiality, integrity and availability. On the other hand, every time data or information security is being discussed, it is said that physical component are always more vulnerable when it comes to attacks or exposure. In this perspective, how safe is it for companies, banks or institutions to use the Hardware Security Modules as a critical component for the encryption key management?
In the context of banks or other financial institutions that use the card payment method and the HMSs architecture, physical protection has always been one of the most important things to be remembered, even though in this case attacks via the software running the host computer of the HSM are neither less dangerous nor serious.

Nowadays everyone that has access on a network such as office or school workstation, accessing the Internet as a personal commodity or being part of a more modern computing environment, has been faced even with the system vulnerabilities. Some of them are for example different applications with flawed interfaces, operating systems where protection is not at the best state, security policies that are often hard to be implemented and users as the weakest part, can easily cause the biggest problems from flawed authentication, absence of knowledge or bad intentions. As a result, even though it is harder to access an encapsulated circuit board (HSM) from the network side, network interface can be as weak as the physical one. Moreover, HSMs are not only used to protect the information from un-authorized individuals, but can also be used to protect against problematic sites, for example, a company can store on an HSM not just SSL private keys but the entire server end of a Web application, thus protecting its customers information too [4].

1.1. Protection in three states

HSMs ensure security and protection of sensitive data in all three states of the information:

- Data-in-transit

The secure transmission of data-in-transit relies on both encryption and authentication. Encryption does not make data secure, but not using encryption, however, means that any data-in-transit is totally easy to be read [5]. This is mainly achieved through TLS, Transport Layer Security, which is the protocol that works on the top of the transport layer to secure application traffic and to provide end-to-end secure communication [6].

- Data-at-rest

This is also considered as storage encryption. It protects data while it resides on the media/ device and it involves encrypting data that will be decrypted when it flows through the same point in the opposite direction [7].

- Data-in-use

HSM ensure security of data-in-use predominantly by TEE, Trusted Execution Environment which is commonly known as an isolated processing environment in which applications can be securely executed irrespective of the rest of the system [8].

2. Cryptography and Security

Applying cryptographical operations in the real word systems requires some specifications and one of the most important factors is randomness. Processes like key generation and management, algorithms, encryption of information need unpredictable randomness even by a lot of efforts. As a result, this can be quite challenging and difficult to be performed on a standard computer or computing environment. Many types of cryptographical algorithms such as RSA, DSA, and Diffie-Hellman can be difficult for a standard CPU, since at the same time, this machine should also handle Web requests, update information, manage network packets, organize interactive user responses

and so on. Thus, these processes are really time-consuming and the machine has to do them in cycles and still manage to be performant [9]. This is the reason why, in order to ensure the process is going to work fine and data protection and security are still a priority, standard machines are supplied with additional specialized hardware that will perform key generation and management. In this case, the additional hardware is the Hardware Security Module. HSMs support both symmetric and asymmetric (public-key) cryptography. For some applications, such as certificate authorities and digital signing, the cryptographic processes are managed in asymmetric key pairs. On the other hand, with applications such as data encryption or financial payment systems, the cryptographic algorithm consists mainly of symmetric keys [10].

From the Public-key cryptography context, basic RSA encryption usually requires an algorithm carefully designed to avoid cryptographic weaknesses. Signature verification for example requires performing cryptographic hashes and because of such processes and operations, an HSM's cryptographic performance can incur an unexpected dependence on the internal CPU, especially if the HSM is matched with a too-slow CPU [4]. On the contrary, as it was stated above, another application of HMSs is related to symmetric cryptography. It often operates on very large data items and since a fast machine does not help the process if it is fed slowly. This can result in an HSM that performs fast symmetric cryptography, only when data items are sufficiently massive and on smaller data items, the speed will be smaller too.

2.1. Randomness

As it was mentioned above, one of the most important factors required for the HSMs to function properly is randomness. Furthermore, for ensuring correct cryptographical operations, the seed used should be expanded into a longer sequence that has to be random and undiscoverable from any adversary [4]. As a result, in the key generation and management process, unpredictable and high-quality randomness is considered fundamental since it should avoid worries and confusion related to the undiscoverable seed, or the failure of the algorithm used for the process [3]. This is the reason why HMSs are used to generate keys. Their architecture is built as a hardware-based random number generator, which generates random bits from the laws of physics, such as for example via a noisy diode. On the other hand, as a physical component of a system, HSMs as a hardware-based solution, have their disadvantages, because sometimes the bits need to be reprocessed to correct for statistical bias. According to NIST's standards [11], when this happens, hardware-derived bits can be reprocessed through a pseudorandom number generator. There are two types of generators for producing random number sequences: True Random Number Generators (TRNGs) and Pseudo Random Number Generators (PRNGs). Furthermore, since bits are generated at some finite rate, it is better to consider if there are enough bits available for the process.

2.1.1. TRNGs and PRNGs

Random number sequences are the fundamental for many cryptographic algorithm and applications, especially for the generation of strong and secure keys that cannot be discovered by hackers or other attackers. This is the reason why it is important for the generated random numbers to be unpredictable. There are different types of statistical tests that can be applied to a sequence to evaluate the fact that the sequence is truly random. To check the randomness of sequences many different indexes are suggested. Each index emphasizes only one part of the phenomenon and measures its different aspects, for example, the distribution of elements in the sequence, dependencies between contiguous reactions, counting tendencies, etc. They are classified as Miyake et. al., 2000; Friedman & Miyake, 2004; Towse & Neil, 1998. The first checks equality of distribution of different possibilities, the second checks indexes concerning relationships between consecutive responses and the third concerns repetitions of the same options in different distances [12].

In the context of random number sequences there are two main sources to generate them. TRNGs are systems which extract randomness from non-algorithmic random phenomena, like temperature fluctuations, radioactive decay, ambient radio noise, hard disk access times, or user interactions with the PC. Since the phenomena used are unpredictable, TRNGs produce real random data instead of just random periodic sequences [13]. A PRNG, on the other hand, is an algorithm that generates numbers that appear random. They are normally constructed from primitives such as block ciphers, hash functions, and stream ciphers [14]. PRNGs require some input (seeds), along with some deterministic algorithms to generate multiple pseudo random numbers. They are faster than True Random Number Generators and as a result are preferable when several random-like numbers or sequences are required. TRNGs, as it was stated above, make use of non-deterministic sources along with some post-processing functions for generating randomness [15].

2.1.2. Noise sources

Random number sequences need entropy and the main source how it is caused is from a noise source. Noise sources are divided into two categories, physical and non-physical sources. Physical noise

sources use dedicated hardware to generate randomness; whereas non-physical noise sources use system data or human interaction input to generate randomness. As system data is used for example the output of API functions or data derived from RAM, while human inputs include for example keyboard strokes or mouse movements that are caused by the interaction with the machine.

Since physical noise sources are considered to offer greater randomness, they are used in HSMs in order to ensure more secure cryptographical processes. Moreover, it is easier to find physical noise sources in the environment. These sources include physical phenomena such as thermal noise, atmospheric noise, radioactive decay, keyboard strokes or coin tossing. Noise sources are divided in two classes:

- Quantum-based noises
- Non-quantum-based noises

Noise sources based on quantum effects are very complex in implementation, but give stable results, while non-quantum sources are more economical, yet give unstable results. The quality of them depends on various factors such as temperature [13].

2.2. Tamper resistance

In cryptography and data encryption, the algorithm used in the process is public and anyone can have direct access to it. On the contrary, the difficult part and the factor which ensures the security is the key being used. It is the most essential part since it is also being used again in the decryption process. In symmetric cryptography key length is 64-256 bits while for the asymmetric algorithms the keys used are 256- 4096 bits. Since the way to get the key is from analyzing the plain-text and the cipher-text, for algorithms such as AES, it takes a very long time to calculate the result. Thus, they are considered quite secure.
On the other hand, the hacker who wants to reveal the key in order to decrypt the information being interested for, will find a way to this data. Since the key is being stored in the memory, due to system flaws or the effort of the hacker, it can be disclosed. The resistance against the attacks or the hacker's efforts to gain access to the key is called tamper resistance and is one of the most important things to be considered while designing a computer system [16].

Tamper resistance can be physical and logical. Since HSMs are tamper-resistant on both these sides, makes them a secure solution for data protection. There are many tests and mechanisms to detect either physical or logical tampering. HSM vendors, together with their product offer the tamper detection tools or functionalities too. Intel for example offers the possibility to enable the frequency, temperature and voltage detection features, offered together with the user's guide to each method in order to customize his/ her own needs.

2.3. Main functionalities

The Hardware Security Modules have a wide variety of uses that not only help in the encryption of information, but also ensure the data protection and security. Thus, there are some fundamental purposes in the HSMs usage such as:

2.3.1. Generate, store and protect cryptographic keys for the system's PKI.

HSMs, whether they are standalone network-connected or plug-in devices, have built-in random number generators that provide randomness and unpredictability of the keys [17]. Also, since they are separate from servers, keys are kept secure from generation to revocation or any possible destruction. Furthermore, since HSMs are temper-resistant on both sides, physical or logical, their architecture is built such that erases or destroys all the cryptographic data in order to prevent corruption. This is known as 'zeroization'. According to NIST, after a key or a piece of secret data is destroyed/ zeroized, no information about its value can be recovered [11].

2.3.2. Protection of keys from extraction.

An HSM encrypts the encryption keys in order to prevent the extraction of the plaintext, which would make the decryption process very easy. Furthermore, systems can use the encrypted keys even without having direct access to them [10]. Even more, the keys can be protected in all the workflow phases, production, testing and implementation. But it is suggested that the same HSM should not be used across different computing environments. Using a stand-alone HSM helps to prevent key exposure so the affected processes are more secure and protected.

2.3.3. Improve server performance.

Since the HSMs are stripped-down and standalone devices/ processors, they that can perform operations on their own. On the other hand, some types of HSMs are equipped to act as web traffic accelerator. As a result, behaving in such a way, offloading cryptographic operations and maintaining load balance, they improve the overall server performance.

2.3.4. Ensure compliance with security regulations and audit processes related to data being processed.

Another important characteristic of the HSMs is that recently, with the development of technology and security policies, they are validated hardware components too. This ensures security compliance because, they meet specific industry standards. For example, as a typical computing device they provide logs that inform the host about different processes such as the cryptographic operations they are programmed to perform, the time when these operations were carried out, and the responsible party for authorizing the operations and processes.

According to NIST, the Hardware Security Modules as critical key management components of the physical infrastructure that makes secure key storage and cryptographic operations possible [18]. Globally, the HSMs are used for various usages in all industries, either of critical importance, or less serious, yet important, such as [18]:

- Certificate authorities (public and private CAs),
- Government and public sector organizations,
- Cloud service providers and vendors,
- Banks, credit card companies and other financial institutions,
- Blockchain platforms and entities,
- Automotive manufacturers,
- Entertainment service providers, and
- IoT device developers and manufacturers.

3. How does a HSM work?

The way how HSMs work is similar to a vending machine or an ATM, even though their purposes are totally different. Both vending machines or ATMs are isolated environments where things like food, drinks or money are kept. So, in order to take something from a machine, a person should at first put something in to take the desirable thing out; for example, put some coins in to take the food out, or put the card and withdraw the money back from the ATM. In this entire cycle, the involved person never changes or interferes with something else inside the machine. The HSM is also an isolated computing environment that accepts user inputs and generates outputs, but the user can neither see nor access the internal operations of the device that made the process go like it was intended. HSM generates for example a signed certificate, but the user cannot see, access or modify the cryptographic key that made it possible. All in all, what is enough for the user is the encryption keys safety, because data can be encrypted, yet information is not secure if the keys are exposed. If it is supposed for example that an authorized user exploits a system vulnerability and leaks sensitive data to the public, the private key that is used to secure the financial information of the company and its customers is being exposed. As a result, even if the information is encrypted, with the discovered key, the information is neither secure, not protected any more. This can be avoided if the keys are stored in a secure HSM, which will not only protect them, but also ensure strong security.

But why should a company use a Hardware Security Module, when its cost is something to be considered and on the other hand web server's architecture is supposed to offer some built-in functionalities that can also ensure protection?
The answer is very simple. Since HSMs are isolated devices, with limited usage and as a result limited attack vectors too, they provide significantly more secure key storage than a traditional web server. Web servers are used to run many applications and functionalities and since the access is larger, the danger is on a larger scale too, because hackers have a higher chance to exploit the vulnerabilities. That is the reason why there are some industries where it is better to use the HSMs instead of a server's security functionalities itself. As we stated before, companies use these devices to use and store the keys they use to sign their PKI certificates, software code and documents, to keep them more protected. Furthermore, public certificate authorities use these devices to create, store and manage their sensitive keypairs too.

3.1 HSM vs TPM

Hardware security modules are isolated hardware components and tamper- and intrusion-resistant devices that are used by organizations and companies to store and protect cryptographic keys [4]. Then, all the cryptographic functions are being processed within the HSM's secure environment, ensuring security and data protection, since the private keys are hidden in the HSM where risk of exposure is very low and sensitive data is also immune from becoming corrupted or compromised. While all these operations are being processed, keys are available to be used by authorized users or employees who can use the keys without needing direct access to them [10]. Since their purpose is, not only to secure the keys but also to control the access, limiting the risk of exposure for the private keys, using HSM, the system can execute cryptographic functions and authentication without loading a copy of your private key into memory on your web server, where it can be hosted. This is more secure even for the system, since web servers, as it was stated above are more vulnerable to attacks and hackers. On the contrary, it is often thought that a Hardware Security Module is version of a TPM, which stands for Trusted Platform Module, because first of all, both HSMs and TPMS are tamper-resistant hardware components and are globally used by the organizations and companies for the cryptographic operations, in order to ensure security and protection. But they are not the same thing and

there is a huge difference between both of them.

A *Trusted Platform Module* is a hardware component that is incorporated into individual devices and is specific for the "parent" device. Thus, TPMs are computer chips that are physically attached to the device's motherboards to secure their PKI keys while keeping them separate from the device's CPU memory. This feature is used to ensure device integrity and provide an isolated environment for the cryptographic operations [19].

A *Hardware Security Module* is on the other hand an external device that can handle operations and processes related to many devices and applications across an organization's network. These hardware devices neither are limited to individual machines, nor incorporated into them and as a result they're intended for use at-scale by applications and servers across your organization.

3.3. Architecture

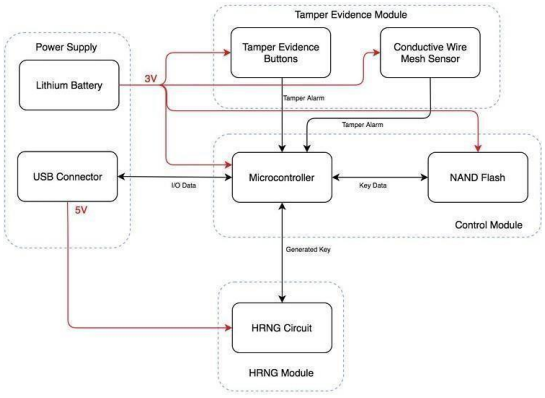

Figure 1 HSM Block Diagram [20]

As it is seen in the diagram above, there are four main parts in the HSM's physical and logical architecture [20]:

- The control/ encryption module

The control module performs operations like encryption, decryption or key management. Moreover, the control module uses Cipher Block Chaining (CBC) AES, a more complex version of AES for the encryption and decryption processes.

- The power supply

According to Papa et al [20], the main source of power for the HSM is a USB power supply with an output of 5 volts that will allow to power all the components of the device. The 3volt lithium battery, on the contrary is included for the tamper evidence module when the device is not connected by USB, to ensure the device's safety.

- Random Number Generator module

The Hardware Random Number Generator, as it was stated above is essential for this encryption processes of a HSM. The HRNG does not use a seed, but transistor uses a noise source to generate random bits. In this case [20], a reverse biased or negative voltage applied transistor will generate noise, in order to have a readable yet unrecognizable output of statistically random bits.

- The tamper evidence module

This module will zeroize the key in two different scenarios: when the casing has been removed or when an attacker is drilling into the case [20]. The microcontroller will poll the alert signal from the circuit and when an alert is sent it will also zeroize the keystore.

3.4. HMSs and Moore's Law

Moore's Law is a techno-economic model that has enabled the IT industry to double the performance and functionality of digital electronics roughly every 2 years within a fix cost, power and area [21]. Moore's Law as an observation that the number of transistors on a microprocessor chip doubles approximately every two years, leads to the fact that the increase of number of transistors, will eventually increase the computing power of a machine, yet cost and other components remain the same. As a result, this change affects not only the hardware components of the systems, but has also a great impact on the security of cryptographical algorithms and processes. As computing power increases, it is easier for symmetric key algorithms such as DES (Data Encryption Standard) and AES (Advanced Encryption Standard) to be broken, even though there was a time these algorithms were considered secure. On the contrary, it is not always true that the only component that affects the break of a cryptographic algorithm is the increased computing power influenced by Moore's Law. The RSA (Rivest-Shamir-Adleman) algorithm for example is still considered secure. This happens because, its security is based on the difficulty of factoring large prime numbers, which is neither affected by Moore's Law, nor determined by increased computing power. Therefore, considering HMSs, Moore's Law result also affects their performance, creating stronger and more powerful computing devices over the time, but it is not the only factors that determines their security.

4. Types, options and common use cases

As it was stated above, HSMs as dedicated and stand-alone or plug-in processors, specifically designed for the protection of the secret keys, are used by enterprises to protect secret information such as transactions, identities or financial data applications, by securing cryptographic operations, helping in encryption, decryption, authentication and digital signing services processes.

4.1. HSMs in card payment systems

One of the most important uses of the HSMs is for sure in card payment systems [4], where they normally provide security via the cryptography principles. Some of their common use cases in the payment industry are:

- PIN generation, management and validation
- PIN block translation during the network switching of ATM and POS transactions
- Card, user and cryptogram validation during payment transaction processing
- Payment credential issuing for payment cards and mobile applications
- Point-to-point encryption (P2PE) key management and secure data decryption
- Sharing keys securely with third parties to facilitate secure communications.

Another important feature to be discussed is that with the development of technology, there are a lot of options to choose. Nowadays HSMs come in a wide variety of options related to physical sizes and applications. Some of them can be small plug-in cards or USB drives, while others are large external devices that should be stored and kept in secure locations. Hardware security modules can be very expensive because the initial price does not include other costs such as additional hardware, support, and maintenance. But, in case a company or organization cannot afford to buy one or more HSMs, it can use the cloud-based option, because some of the biggest HSM vendors like Amazon Web Services offer the cloud-based HSM's products and services. When using cloud as such solution, a company can:

- Rent a physical HSM appliance that can be stored in the company's off-site data center.
- Pay for access to the functionalities of an HSM vendor's device or appliance.
- Pay for access to a virtual environment within a vendor's shared HSM.

So, when it comes to using one of the options, or the other, there are advantages and disadvantages too. It is up to the company itself and what it needs for the work processes.

4.2. HSMs types

There are two main types of HMSs:

General Purpose Hardware Security Modules

In the General purpose HSM's group are included the HSMs that all the companies and organizations globally use in the context of cyber security. These devices are typically built according to some specified cryptographic principles, which emphasize how the HSM should behave in specific scenarios and under specific conditions, to ensure the functionalities are applied [22]. That is the reason why most of them use vendor-neutral APIs to facilitate communication and cryptographic services, the HSM is used for.

Payment Hardware Security Modules

As it was mentioned above, the second category of HSMs are the ones used in the payment industry. In the same way as general purpose HSMs, they are also isolated, temper-resistant and are used by businesses to store and secure the keys, but they are more specialized, since they are more custom-built in order to ensure a better key management in the financial applications, transactions and card payment processes [2], that also have a special importance. As a result, they should be compliant to different standards compared with the general-purpose ones. Often the interfaces used for these HSMs are also different, since they should meet the specified security requirements.

4.3. Quantum Computing and the future

One of the most important developments of the last decades is quantum computing. It combines information theory and quantum mechanics to create a new era of computation and technology. According to National Security Agency [23], quantum computers can perform mathematical algorithms exponentially faster than a classical computer. In place of ordinary bits used by today's computers, quantum computers use "qubits" that behave and interact according to the laws of quantum mechanics. This quantum physics-based behavior would enable a sufficiently large-scale quantum computer to perform specific mathematical calculations that would be infeasible for any conventional computer.

Even though recently quantum mechanics are still being discovered and their implementation in the real world's applications is still low, laboratory experiments have demonstrated quantum computations with several quantum bits performing dozens of quantum operations and this will totally change the future. For now, it does not provide efficient solutions to all problems. Strong limitations on the power of quantum computation are known since it has been proven that quantum computation provides no significant advantage over classical computation and yet, quantum information processing has changed forever the way in which quantum physics is understood [24].

From the cryptography and HSMs perspective, the development of quantum computing will also affect the future, but the first thing to be considered is that, since quantum computing technology is not yet in

general use, what is discussed is like a projection for the future. Despite considering this, since the basic principle of quantum computing is creating a new kind of super-performant computational environment, different from the traditional and actual one, quantum computing will process huge amounts of data. This will therefore lead to major advances in cybersecurity, such as Quantum Key Distribution. As a result, quantum cryptography and its applications in encryption and decryption processes, will create stronger algorithms and eventually, stronger and unbreakable keys. On the contrary, application of quantum techniques can also be seen as a threat since it could be used to break cryptographic keys currently in use because current web applications are very vulnerable to quantum techniques. Yet, this is one of the greatest challenges of the future, to manage the process really well and to combine quantum techniques for stronger and more secure hardware and software components of the systems.

5. Conclusions and future work

To sum up, HSMs are isolated hardware components that are used by companies and organizations to store, protect and secure the key used in the cryptographic operations that are being processed within the isolated computing environment. When discussing HSMs there are two main perspectives, security of the key and security of the data decrypted by the key. So, the HSM operates as "a separate server", where app server sends the encrypted data for decryption and the cleartext data comes back in the pp server. In this process, protecting the key is more important than protecting the data in packets, and this is what a HSM ensures.

As any other component of the system, using a HSM has its advantages and disadvantages too. A HSM can be quite expensive to purchase, maintain, and operate, and since they are usually priced on transaction throughput, the more data you need to decrypt at a time, the more expensive the HSM is. Furthermore, when buying an HSM, probably you as a company will spend a lot of time learning how to use it, how to perform key management and implementation into the existing systems.

On the other hand, if a company uses existing server's functionalities in order to ensure data security or chooses to build its own server to perform the decryption of the data, it should make a lot of efforts to harden it against attacks, issuing certificates to authorized app servers, performing audits, managing security policies and so on. As a result, this will cost less than buying an HSM, but the key is still at more risk than in a HSM. Given all that, it is a matter of expenses vs risk and it is up to the company itself to choose the best option. Considering how critical the information is and if it is worth spending a certain amount of money on security, there comes a point when a decision should be made.

When it comes to the future, as time progresses, host CPUs tend to get faster, thanks to Moore's Law, even though hardware stays the same, so the future trends have to consider this part, since complex cryptographic operations will need faster hardware devices too, the same way as the faster processors are implemented. Moreover, Quantum Computing and its future application in cryptographical algorithms and processes, will also affect the computational performance and security of HSMs.

1. References

[1] Yu W., Li W., Wang J., Wei C., "A study of HSM based key protection in encryption file system", 2016 IEEE Conference on Communications and Network Security

[2] E. Bonner, J. O' Raw, K. Curran, "Implementing the Payment Card Industry (PKI) Data Security Standard (DSS)", Volume 9, Number 2

[3] M. Folkemark, V. Rydberg, "Performance evaluation of a Hardware Security Module in Vehicles", University of Gothenburg, 2021

[4] Smith S. W., "Hardware Security Modules." In B. Rosenberg (editor). Handbook of Financial Cryptography and Security". Chapman and Hall/CRC. 2010. 257--278.

[5] J. Vesperman, "Introduction to Securing Data in Transit", chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://tldp.org/REF/INTRO/SecuringData-INTRO.pdf

[6] A. Satapathy, J. Livingston, "A Comprehensive Survey on SSL/TSL and their Vulnerabilities", International Journal of Computer Applications (0975 – 8887), Volume 153 – No5, November 2016

[7] Storage Networking Industry Association, "Storage Security: Encryption and Key Management", August 2015

[8] M. Sabt, M. Achemlal, A. Bouabdallah, "Trusted Execution Environment: What It Is and What It Is Not", chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://hal.science/hal-01246364/file/trustcom_2015_tee_what_it_is_what_it_is_not.pdf

[9] P. Kocher, "Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. In Advances in Cryptology—Crypto 96". Springer-Verlag LNCS 1109, 1996.

[10] A. Ramesh, A. Suruliandi, "Performance analysis of encryption algorithms for Information Security" in 2013 International

Conference on Circuits, Power and Computing Technology.
[11] National Institute of Standards and Technology (2002) Security Requirements for Cryptographic Modules. (U.S. Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS) 140-2, May 25, 2001 (Change Notice 2, 12/3/2002). https://doi.org/10.6028/NIST.FIPS.140-2
[12] J. Barbasz, Z. Stettner, M. Wierzchon, K. T. Piotrowski, A. Barbasz, "How to estimate the randomness in random sequence generation tasks?", chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://core.ac.uk/download/pdf/154361007.pdf
[13] O. Petura, "True random number generators for cryptography: Design, securing and evaluation", University of Lyon, 2019, chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://theses.hal.science/tel-02895861/document
[14] H. Ng, "Simple Pseudorandom Number Generator with Strengthened Double Encryption (Cilia)", chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://eprint.iacr.org/2005/086.pdf
[15] R. Soorat, M. Kandukuri, A. Vudayagiri, "Hardware Random number Generator for cryptography.", https://www.researchgate.net/publication/282639432_Hardware_Random_number_Generator_for_cryptography
[16] T. Fujino, T. Kubota, M. Shiozaki, "Temper-resistant cryptographic hardware", IEICE Electronics Express, Vol.14, No.2, 1–13
[17] N. Saboonchi, "Hardware Security Module Performance Optimization by Using a 'Key Pool' Generating keys when the load is low and saving in the external storage to use when the load is high", Sweden, 2014
[18] The National Institute of Standards and Technology (NIST) Special Publication "Recommendation for Key Management: Part 2 — Best Practices for Key Management Organizations" (SP- 800-57 part 2, rev 1)
[19] J. D. Osborn, D. C. Challener, "Trusted Platform Module Evaluation", Johns Hopkins Apl Technical Digest, Volume 32, Number 2, 2013
[20] F. Papa, C. Fisher, N. Schiesl, "Hardware Security Module", chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://courses.engr.illinois.edu/ece445/
[21] L. Shalf, "The future of computing beyond Moore's Law", Lawrence Berkeley National Laboratory, https://www.researchgate.net/publication/338699741_The_future_of_computing_beyond_Moore's_Law
[22] T. Gendrullis, M. Wolf, "Design, Implementation, Evaluation of a Vehicular Hardware Security Module", part of Lecture Notes in Computer Science book series, volume 7259
[23] National Security Agency, "Quantum Computing and Post-Quantum Cryptography", chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://media.defense.gov/2021/Aug/04/2002821837/-1/-1/1/Quantum_FAQs_20210804.PDF
[24] E. Reiffel, W. Polak, "Quantum Computing, aGentle Introduction", chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/http://mmrc.amss.cas.cn/tlb/201702/W020170224608150244118.pdf