ISSN 2278-3091
Volume 8, No.4, July – August 2019
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse63842019.pdf
https://doi.org/10.30534/ijatcse/2019/63842019

High Speed Cryptography Architecture for Health Information Exchange

[1]Mahendra Vucha, [2]A L Siridhara
[1,2]MLR Institute of Technology, Dundigal, Hyderabad, Telangana, India
[1]mahendra.1548@gmail.com, [2]siridhara@gmail.com

ABSTRACT

Since countries like India promoting their healthcare departments to be recognized as clinically advanced in the world. The worldwide recognition for the countries in the domain of healthcare is possible only when they adopt best practices like cost effective quality healthcare standards. The secure Health Information Exchange (HIE) is an important way to maintain Electronic Health Record (EHR) which improve healthcare system of a country. Development of electronic health records adds potential benefit in providing reliable access to health information and thereby improving the national healthcare system. However, the prospect of storing, transmitting, and sharing health information through electronic media raises many new challenges on information integration and protection. Protection of patient health information electronically is very crucial and need to develop systems and structures that support secure health information exchange among healthcare providers, patients and customers. The core competency of EHR and HIE is to maintain confidential and supportive relationship among patient and healthcare providers in maintaining patients health information. The confidentiality in health records can be achieved through mutual legal agreements, advocacy and technology that ensure appreciable secure information interchange. So this article also demonstrates effective practices which can be applied to the development of HIEs and presents a security architecture which can be adopted for exchange of health information to leverage common government and commercial practices in electronic health record.

Key words: Health Information Technology, Health Information Exchange, Electronic Medical Records, Electronic Health Record.

1. INTRODUCTION

In recent days, the medical field has shown interest towards Electronic Health Record (EHR) for digitalize patients health record and medical data processes to provide instant access of health information at the right place in right time. The EHR provide efficient exchange of health information for improvement of societal life. EHR has the goal to develop fully inter operable, patient centric and easy accessible systems and it demands strong security for storing, transmitting and accessing health information over a common network. The adoption of EHR in medical domain enables the nations look forward to digitize patient record in usable format. The ability of exchanging health information brings at most social benefits promised by EHRs [2]. The exchange of information can be made possible by transmitting data through some networking technology for promoting health information security standards [3]. The substantial security can be obtained only when health information exchange (HIE) is implemented effectively [4]. So, the HIE got significant attention in both government initiatives and also academic research. Irrespective of method of exchange, the attention rising on sharing of patient health information with several healthcare entities can raise few concerns on patient information privacy and security. This article addresses few concerns in providing security during HIE.

1.1. Electronic Medical Records (EMRs) vs. Electronic Health Records (EHRs)

EMRs are digitized information of paper based patients clinical data and charts that help in delivering effective health care services. The EMRs holds individual patients treatment history. EMRs might be having specific data which cannot be shared. EMR constitute the information of HIE plans and further health care indications. EHRs support comprehensive processes of continuous lifelong healthcare indications for an individual. The EHR presents the ability of sharing patient medical information with other healthcare providers and have patient's health information follow him or her during the various health care modalities engaged by that individual. A secure and efficient storing, retrieving and transforming health information must be adopted to manage both EMRs and EHRs.

1.2 Benefits of HIE

Safety – Healthcare need safety measures to maintain patient health information and their medications safe for ordering new treatments. Emergency care is possible for patients only when health information of the patient is exchanged safely.

Timeliness – Fastest patient health diagnosis is possible only when emergency ward or consultant physician can have accessibility to patient health record from the primary healthcare provider rather than getting it from the scratch.

Cost reduction - Healthcare cost can be minimized by providing access to patient health record while avoiding duplicate tests.

Confidentiality – Patient health information is confidential and need to be secured with safety, privacy and confidentiality. Protecting the information stored in medical records is very important due o the patient lives sensitive to the correctness and security of their health information. Thus this article presents a secure framework for HIE.

1.3 Privacy and Security of health record

Patient life and health conditions depend on his health information and hence patient health records need to be secured. The objective of HIE is to provide secure access to patient's health information when it is required. Access control and Authorization are very essential parameter of electronic health record system to share health information of patients with external healthcare provides. Accessing the health information virtually may bring the complications and it demands implementation of efficient secure system [16]. Extensive research has been done on security issues of HIE and researchers have suggested few architectures for secure sharing of health information over distributed and interconnected health information systems.

2. LITERATURE SURVEY

The need of confidentiality for health records demand the design and implementation of cryptographic technique to bring security for health records and health information exchange. In [2], a security framework was presented health information exchange between Malaysia and other countries. Authors in [3] demonstrated a secure algorithm for sharing across two un-trusted program model. In [4], a secure outsourcing framework proposed for large-scale systems using linear equations. The framework in [4] has utilized the sparse matrix and proposed a secure outsourcing method for linear equations in completely malicious model. Now a day's security is becoming necessary in wireless devices, wireless data transfer and networking applications. Practically the Advanced Encryption Standard (AES) has been used for most of the real life applications. The major optimization methods of cryptography frameworks are (1) Architectural optimization and (2) Algorithmic optimization. Architectural optimization can be obtained by adopting the techniques like pipelining, paralleling and loop unrolling. Commonly speed can be enhanced by operating multiple rounds concurrently and which leads to increase in area of architecture. The loop unrolling can enhance speed while rapid increase in area.. Optimization develops algorithmic strength by processing inside every round unit. There are various techniques developed in literature that reduce the area and critical path of each unit in architecture of cryptographic technique. Analyzing the energy consumption of RC4 and AES algorithm in wireless LANs have been presented by authors in [5] and the performance metrics considered are encryption throughput, energy cost, key size and CPU workload. The experimental results in [5] shown that the AES provides high security than RC4, but the RC4 is fast and energy efficient for encrypting large packets in a network. Data encryption standard and triple data encryption standard provide better security but advanced encryption standard and blowfish not so for [6]. RSA is suitable for business applications whereas the Blowfish is considered as strong encryption algorithm based on its minimum collision. AES requires more power for processing the algorithm but it has poor performance with other alternatives. Marwa AbdEi-Wahed et.al (2008) presented efficiency and security performance of four image encryption algorithms suitable for practical applications. The purpose of choosing these image encryption methods is to evaluate the execution time, image quality in encryption and the memory requirement for encryption. Gurjeevan Singh et al (2011) presented throughput analysis for various encryption algorithms on a laptop having 4GB RAM Core-2-Duo processor with 2.2GHZ CPU, and Windows7 of 32-bit operating system. Throughput in context of encryption and decryption is equal to the ratio of total plain text or cipher text in k bytes to average encryption or decryption time. The conclusions of the presented [7] results is, 1. Blowfish has better performance than AES and DES, 2. 3-DES has least efficiency compared with alternative symmetric encryption algorithms [7]. Rajan Mishra and Shashi Mehrotra et.al. (2011) [8] made a comparative analysis of all encryption algorithms for secure data communication in Network. The analysis was performed based on parameters like memory usage, processing time, area, output bytes, delay, power and throughput. The authors in [8] found that the RSA algorithm has least output bytes, but memory usage is very high [8]. The DES was accepted as an ANSI standard after published by the National Bureau of Sciences (NBS) in 1981 [9] to protect financial information. Rapidly increasing networks demand message data in both the public and private sectors which need Authentication, Availability, Integrity, and confidentiality. The DES operates on some basic mathematical operations and it transforms the message known as plain text into Cipher text (Secret data) and also transforms the secret data into an original text message known as plain text.

DES algorithm plays vital role in encrypting the data for secure financial information and most of the American financial services depend on it to encrypt the financial transactions for their business. DES consists of some essential operations to encrypt data block, such as permutation primitives, several substitution methods. Permutation primitives are used to reverse the encryption operation of algorithm. The Permutation primitives and substitutions are applied to data blocks iteratively for a particular number of times [10]. In Data Encryption Standard algorithm each set of primitive operations are known as round, to meet the security goals DES uses total 16 rounds to ensure the dataeffectively.DES has one secret key, which is used to control the whole operation of DES. The secret key contains total 56bits of block information. In Recent paper proposed Triple DES using 112-bit symmetric key algorithm is secure until 2050 years [11]. There are some generic attacks on DES algorithm. To overcome the attacks on DES there is need of new cryptography encryption algorithm to protect the data securely called as triple DES is proposed in [11]. The message Digest-4 is most popular hash function developed by cryptographer in the year 1990. The lowest message digest

length is 128 bits which is used for shorter messages. Message Digest-4 algorithm has supported various hash algorithms, namely Message Digest-5, Secure Hash Algorithm-1, RACE Integrity primitives Evolution Message Digest(RIPEMD). RIPEMD is depending on the MessageDigest-4 and it has followed the design methods of MD4.RIPEMD performance is almost familiar with the most popular hash algorithm known as Secure Hash Algorithm-1. Message Digest -5 is popular for various security applications [12] to verify data integrity of the message.

3. PROPOSED SECURE CRYPTOGRAPHY ARCHITECTURE

The security architecture is presented in this article can be adopted for exchange of both patient text and image data. The functional diagram of security architecture for patient health information exchange is shown in figure 1. The figure 1 describes data flow of how the patient health information is digitized and shared with medical consultant. The patient health information can be preserved in both text and images after digitalization using standard methods. The digitalized information can be secured while storing or exchanging through a medium.

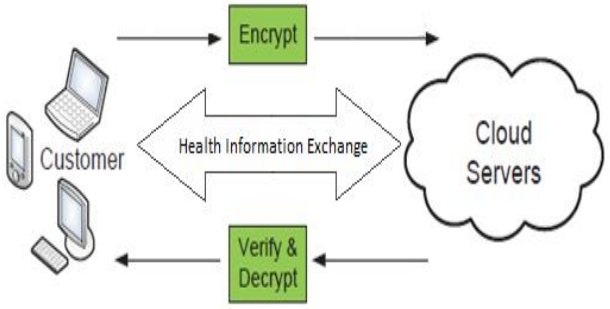

Figure 1: High Speed Security framework

Therefore, level of the security for the health information is completely depends on type of encryption standards utilized and it also decides information exchange speed. In this article, an extensive survey has been conducted and proposed SHA-3 as cryptography standard that provide required security for health records at sufficient speed. The encrypted health data can get updated in the cloud server and made available to the patient and his authorized health practitioners. The SHA-3 cryptography technique architecture is described in the subsection.

3.1 Secure Hash Algorithm - 3

The SHA-3 algorithm popularly known as Keccak, operates on sponge construction with iterative transformation to increase level of security. Thus, SHA -3 minimizes encryption iterations there by enhances the security as compared to the techniques presented I literature [7]. The Keccak algorithm can be adopted as hashing function to provide security for information authentication, password protection and digital signatures. The major objective of keccak architecture is to enhance the security algorithm speed and it is also proved in literature. The keccak also helps in identifying preeminent algorithms among the new hash algorithm principles. These kinds of security architectures provide enhanced security, throughput, speed, area and power consumption. This research is presented to demonstrate the behavior of SHA-3 and also design cryptographic architecture for HER and HIE. The SHA-3 is a kind of sponge function produces hashes of size equal to sum of bit rate 'r' and capacity 'c'. Selection of r and c is depends on length of hash output or hash digest for example, for 256-bit hash output, r is 1088 bits and c is 512bits. Similarly for 512 bit output, r is 576 bit and c is 1024 bit. The 1600-bit state space 'A' of SHA-3 can be described as 5x5 matrix of each element size 64-bit and the state space matrix A is shown in figure 2.

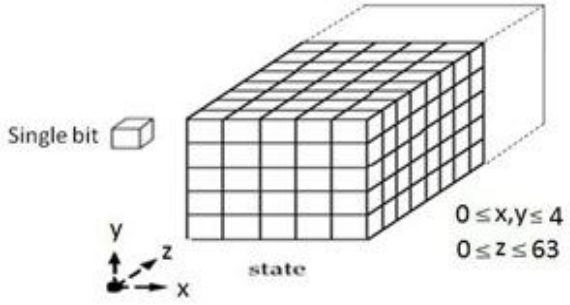

Figure 2: State Matrix (A) of SHA-3 [5]

The SHA-3 block diagram consists of state function, buffer function, round constant and keccak algorithm as demonstrated in figure 3. The state function holds the1600 bits of data in the form of matrix of size 5x5 with each element 64 bits called state space matrix. The buffer function considers data or information and state matrix A, as input and generates hash value of fixed length. The buffer function accepts elements of size 64 bit as message input and generates hash output. The objective of sponge function is to provide incontestable security against the attacks and also make the security framework simple and flexible. The sponge construction could have two functional phases that enhances the security level of Keccak algorithm. The sponge function is kind of constant length transformation that operates on the number of bits equal to sum of 'r' and 'c'.

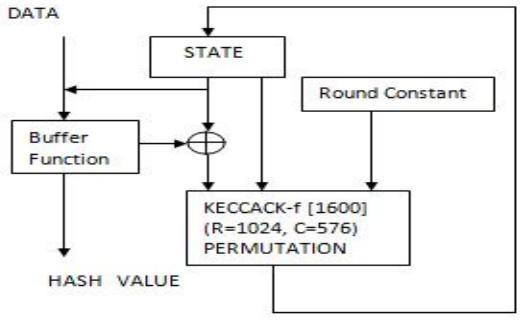

Figure 3: Block diagram of KECCAK [19]

3.2 Sponge construction

The sponge construction is a transformation that accepts fixed length bits and produces arbitrary length hashes through fixed set of permutations. The permutation function operates

on a fixed number of bits of size b = r +c. The sponge construction function is very crucial to the security of SHA-3 architecture. The behavior of the sponge construction is described in two phases known as absorbed phase and squeezed phase. The architecture of sponge construction is shown in figure 4.

Absorbing phase: The message or information block could perform XOR operation with the first r state bits in state matrix A and the resultant is stored back to state matrix A. The processed message blocks in state matrix are then sent to squeezing phase.

Squeezing phase: First r bits of the state matrix A returned as output hash blocks of the sponge construction. Initially, the input message or information bits are padded using padding technique and then the resultant message is grouped into blocks of r bits each. In absorption phase, the input message R bits are XORed with the predefined bits stored in first 1600 bits of state space matrix 'A' and then the resulted state matrix 'A' gets updated along with the message bits stored in remaining portion of the state matrix. The squeezing phase outputs the part of data from the first portion of state matrix 'A'. The sponge construction function could produce output of any fixed size from the internal state register. The Round Constant (RC) function hold the 24 value which are used for permutations and it assign 64 bit data to Keccak function.

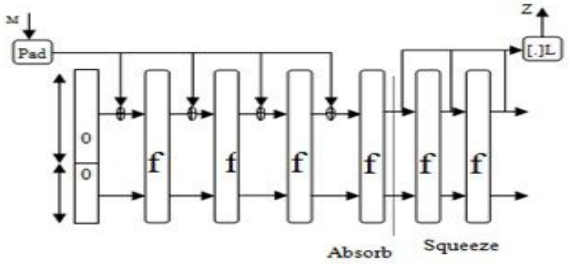

Figure 4: Sponge Construction [17]

3.3 Buffer Function

The state matrix function is designed with 1600 bits in the form of matrix of order 5x5 and size of each element in the matrix is 64-bit. The buffer function in SHA-3 accepts two kinds of input i.e. one is data or information and other is the state matrix, to generate random hash values. The buffer function accepts 64 or 256 bits of message as input and produces hash output of fixed size based on sponge construction.

3.4 Round Constant

The Round Constant (RC) function hold 24 constant values used for permutation and also it assigns 64 bit data sequence to Keccak function as one of the input. The RC constant values are listed in the table 1 in hexadecimal notation.

Table 1: Round Constant values in hexadecimal representation

RC[0]	0000000000000001	RC[12]	000000008000808B
RC[1]	0000000000008082	RC[13]	800000000000008B
RC[2]	800000000000808A	RC[14]	8000000080008089
RC[3]	8000000080008000	RC[15]	8000000000008003
RC[4]	000000000000808B	RC[16]	8000000080008002
RC[5]	0000000080000001	RC[17]	8000000000000080
RC[6]	8000000080008081	RC[18]	000000000000800A
RC[7]	8000000000008009	RC[19]	800000008000800A
RC[8]	000000000000008A	RC[20]	8000000000008080
RC[9]	0000000000000088	RC[21]	000000008000808B
RC[10]	0000000800008009	RC[22]	8000000800000001
RC[11]	000000080000000A	RC[23]	8000000080008008

3.5 Keccak – f

The Keccak function KECCAK-f (1600) perform 24 permutation rounds on state matrix A. The permutation round is described as five sequential stages called Theta, Rho, Chi, Pi and Iota respectively. Each permutation round accepts state matrix A and round constant as inputs and produces hash output of size b bits. The output of each round is then used as input to next round. The behavior of these sequential five stages of KECCAK-f is described in the following Boolean or logical equations.

Theta

$$C[x] = A[x,0] \mathbf{XOR} A[x,1] \mathbf{XOR} A[x,2] \mathbf{XOR} A[x,3] \mathbf{XOR} A[x,4]$$

$$D[x] = C[x-1] \mathbf{XOR} \left(rot(C[x+1],1)\right), \forall in x = 0 \text{ to } 4,$$

$$\text{Where } A[x,y] = A[x,y], \quad \forall (x,y) in (0 \ldots 4, 0 \ldots 4)$$

$$A[x,y] = A[x,y] \mathbf{XOR} D[x] \forall (x,y) in (0 \ldots 4, 0 \ldots 4)$$

Rho and Pi

$$B[y, 2x+3y] = \mathbf{ROT}(A[x,y], r[x,y]); \ 0 \le x, y \le 4$$

Chi

$$A[x,y] = B[x,y] \mathbf{XOR} ((\mathbf{NOT}\ B[x+1,y]) \mathbf{AND}\ B[x+2,y]) \forall (x,y)\ in (0 \ldots 4, 0 \ldots 4)$$

Iota

$$A[0,0] = A[0,0] \mathbf{XOR} RC, R[x] in\ 0 \ldots .63$$

4. RESULTS AND DISCUSSIONS

The behaviour of Keccak function is described within a round function having sequence of steps **θ**, ***π***, ***ρ***, ***χ*** and ***i***. The functions described in the form of equations describe security evaluation of the Keccak- *f* permutations. The following are the notations used to describe the algorithm in executable program.

- *n*r - number of rounds
- b - data width *b*
- *r* – data rate represents number of unknown input bits
- n - output length

Performance evaluation of the keccak can be evaluated by considering *n* = *r* where r ranges up to 12, *n*r ranges up to 8 and widths ranges from 25 to 400.

Performance Evaluation

The programming language C++ was used to describe the proposed security architecture and evaluated for both small and large data sets of Known Answer Test (KAT). The SHA-3 cryptography architecture produces constant length hashes irrespective of input data set length and hence the execution time of the architecture to process the data is constant measured as 4μs. The sponge function of Keccak has dedicated constructions to generate fixed length output hash

functions. This kind of implementations reduces the computation overheads as compared to other hash function families. The number of permutation rounds ensures tradeoff between performance and security through suitable capacity and rate pairs. After implementing the keccak for the predefined input sets, the following observations were noticed.

• Doubling the data width while keeping remaining parameters constant makes the computation time double. Thus computation time grows linearly with respect to data width.

• When number of rounds increased from *n*r to *n*r + 1, keeping remaining parameters constant, makes growth in computation time linearly with respect to number of rounds.

• When the bit rate is increased from *r* to *r* + 1, keeping remaining parameters constant, makes weakly increase in computation time.

5. CONCLUSION

The proposed Keccak security scheme outperforms the SHA-2 in terms of hardware performance**,** speed and area tradeoff. Keccak has optimum hardware and software performance and i**t** is faster than SHA-2 on recent PCs. This framework shines when parallelism is introduced as additional feature. So, the cryptography framework using Keccak as security techniques enable the user to protect their health information and also help them to exchange the information through public data transmission channels. This article presented a software model of SHA-3 cryptography architecture and which runs at optimum speed while providing strong security for Health Information Exchange (EIH). Thus, presented cryptographic architecture offers optimum performance for health and also constant level of security for variable length health information exchange.

REFERENCES

1. P Sailaja and M Vucha, High Speed Architecture for KECCACK Secure Hash Function, International Journal of Computer Applications 139(9):19-24, April 2016. https://doi.org/10.5120/ijca2016909237
2. B. Zaidan, Ahmed Haiqi, A.A. Zaidan, Mohamed Abdulnabi, M.L. Mat Kiah, Hussaen Muzamel, Security Framework for Nationwide Health Information Exchange based on Telehealth Strategy, Journal of Medical Systems, Vol. 31, No. 51, March 2015. https://doi.org/10.1007/s10916-015-0235-1
3. Xiaofeng Chen; Xinyi Huang; Jin Li; Jianfeng Ma; Wenjing Lou; Wong, D.S., New Algorithms for Secure Outsourcing of Large-Scale Systems of Linear Equations, IEEE Transactions on in Information Forensics and Security, Vol.10, No.1, pp.69-78, January 2015. https://doi.org/10.1109/TIFS.2014.2363765
4. Xiaofeng Chen; Jin Li; Jianfeng Ma; Qiang Tang; Wenjing Lou, New Algorithms for Secure Outsourcing of Modular Exponentiations, IEEE Transactions on in Parallel and Distributed Systems,vol.25, no.9, pp.2386-2396, September 2014. https://doi.org/10.1109/TPDS.2013.180
5. National Institute of Standards and Technology: FIPS 197: Advanced Encryption Standard, November 2001.
6. SimarPreet Singh, and Raman Maini, Comparison of Data Encryption Algorithms, International Journal of Computer Science and Communication, Vol. 2, No. 1, January-June 2011, pp. 125-127.
7. Gurjeevan Singh, Ashwani Kumar Singla,K.S. Sandha, Through Put Analysis Of Various Encryption Algorithms, IJCST ,Vol. 2, Issue 3, September 2011.
8. ShashiMehrotra Seth, 2Rajan Mishra, Comparative Analysis Of Encryption Algorithms For Data Communication, IJCST Vol. 2, Issue 2, June 2011 pp.192-192.
9. Data Encryption Algorithm (DEA), ANSI X3.92-1981, American National Standards Institute , New York.
10. Horst Feistel, Block Cypher Cryptographic System, US Patent 3,798,359, March 19, 1974.
11. Arjen K. Lenstra and Eric R. Verheul, Selecting Cryptographic Key Sizes (http://www.cryptosavvy.com/), October 1999. https://doi.org/10.1007/978-3-540-46588-1_30
12. Chih-Pin Su, Tsung-Fu Lin, Chih-Tsun Huang, and Cheng-Wen Wu, National Tsing Hua University, A high throughput low cost AES processor, IEEE Communications Magazine, 0163-6804/03 © 2003 IEEE. https://doi.org/10.1109/MCOM.2003.1252803
13. R. Cramer and V. Shoup, Signature schemes based on the strong RSA assumption, ACM Transactions on Information and System Security (TISSEC) 3(3), pp.161-185, 2000. https://doi.org/10.1145/357830.357847
14. Pooja Singh, Nasib Singh Gill, A Secure and Power-Aware Protocol for Wireless Ad Hoc Networks, International Journal of Advanced Trends in Computer Science and Engineering, Volume 8, No.1, January – February 2019. https://doi.org/10.30534/ijatcse/2019/07812019
15. C.M. Hall, S. McMullen, D.L. Hall, Cognitive engineering research methodology: A proposed study of visualization analysis techniques, In Visualizing Network Inf. Neuilly-sur-Seine, France, 2006.
16. Deepthi Barbara Nickolas, Mr. A. Sivasanka, Design of FPGA Based Encryption Algorithm using KECCAK Hashing Functions, International Journal of Engineering Trends and Technology (IJETT) - Volume4, Issue6, June 2013.
17. Amir Hesam Yaribakht, Mohd Shahidan Abdullah, Alireza Ghobadi, A Novel Color Image Watermarking Method based on Digital Wavelet Transform and Hungarian Algorithms, International Journal of Advanced Trends in Computer Science and Engineering, Volume 8, No.2, March - April 2019. https://doi.org/10.30534/ijatcse/2019/09822019

18. Bradley Dunsmore, Jeffrey W. Brown, Michael Cross, Mission Critical! Internet Security, Syngress Publishing Inc., 2001, ISBN: 1-928994-20-2.
19. Richa Sharma, Purnima Gehlot, S. R. Biradar, VHDL Implementation Of AES- 128, International Journal of Advances in Electronics Engineering –IJAEE, Volume 3 : Issue 2, pp-17-20, 2013.