International Journal of Engineering & Technology, 7 (2.13) (2018) 281-286

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research Paper

Enhancing e-banking security: using whirlpool hash function for card number encryption

Doaa Yaseen Khudhur [1] *, Saif Saad Hameed [2], Shokhan M. Al-Barzinji [2]

[1] College of Computer Science and Information Technology, University of Anbar
[2] College of Education for Humanities , University of Anbar
**Corresponding author E-mail: Duaa_82yaseen@yahoo.com*

Abstract

The Internet played - and still - the key that continuously changing our ways of interaction with people. As a result, several electronic services had emerged allowing businesses to grow by effectively allowing wide and easy interaction with customers and other businesses. The security and privacy of information over the internet in general and in electronic services providers have been the focus of widely published studies and researches, such that several software and hardware based solutions or hybrid of both is required. E-banking services grown significantly in the last decade where all financial matters of customers and businesses can be done online, and therefore, e-banking security and privacy is important. In this paper, I propose the use of Whirlpool hash function to enhance the security of e-bank service providers by encrypting customer's card sensitive information. In addition, based on the review of several articles, I found that Whirlpool outperformed several hashing functions and resists several well-known attacks.

Keywords: *Whirlpool; E-Banking Security; Card Number.*

1. Introduction

The use of electronic banking is increasing day by day. Customers and businesses can interact and manage their financial transactions, paying bills, investments, e-cheques, etc. [1]. With such increase, the possibility of attacking e-banking services inherently increased as well. These attacks considerably threatens banks customers, bank reputation and trustworthy.

It is widely conceived that the security of e-banking rely only on the security and layer of protections employed by the e-banking service provider itself, where the truth is attacks can target customers side, end-to-end network, and also service provider side. Petr, Kamil, and Jiri summarized numerous types of attacks related to e-banking security and categorized them into three main categories: e-banking service providers (such as exhaustive search, malware, and phishing), authentication (such as safe confidentiality, and transections integrity), and trusted devices (represented by the use of secure hardware) [12]. Through literatures, one can observe that new ways of attacks are continuously being found following the widespread use of information technology and its application in the banking industry. [2]

CNBC news published an article which highlights that in 2017 alone, customers of e-banking services had lost 16 billion dollars to e-banking identity theft and fraud [13]. Therefore, security is of great importance in the e-banking industries. In general, consumers of e-banking services will always be concerned about the safety and protection of their financial information from being exposed to unauthorized access. Traditionally, the expected protection is always thought of, as it should come from e-banking service provider, and nowhere else [3-5].

The aim of the research presented in this paper is to study the feasibility of replacing previously used hashing functions with Whirlpool hash function to encrypt customers' card information used in e-banking systems. It also present a complete comparison with regard to hashing speed and security.

Organization

The following sections of this paper present a complete discussion of related work, brief introduction of Whirlpool hash function, weakness and strength points, and cryptanalysis with regards to three main attacks: Rebound, Preimage and Collision, and Meet-In-The-Middle attacks, hardware and software implementation, then last, the proposed implementation and our final conclusion.

2. Related work

To my knowledge, I have found only three published research related to enhancing the security of e-banking systems through implementing of hashing. Bello Alhaji Buhari et. Al. [6] developed a new approach for prepaid-scratch payment card as he suggested using alphanumeric code instead of using digits only. In addition, he employs standard security mechanism using MD5 cryptographic hash function to encrypt the scratch card details saved in the database which renders sensitive information to be unreadable to unauthorized access.

In another work, A.Salma ET .al. [7] proposed authentication framework comprised of Personal Identification Number (PIN) and Fingerprint for users' identification and authentication in Automatic Teller Machine (ATMs) to prevent money theft and other illegal activities in ATM machine. In addition, the authors introduce the use of GSM based alert system such that the nearest police station will be notified during detection of any illegal activities.

M. F. Mridha et. Al. [8] explains a secure protocol and certificate verification mechanism as it checks the authenticity of the sender first; then if appropriate, processes the incoming messages and stores them for further processing. This covers everything from

phishing site detection to two-factor authentication. Having declared all current schemes for protecting online banking lacking in some way, the key aspects of the problem are identified. And presents for a more robust defense system which uses a small security device to create a trusted path to the customer, rather than depend upon trusting the customer's computer.

3. Whirlpool hash function

Whirlpool hash function was developed by Paulo, Barreto, and Rijmen [14] in 2000 and it was one of the few hash functions recognized by NESSIE (New European Schemes for Signatures, Integrity, and Encryption) project alongside SHA-256, SHA-384 and SHA-512 in the NESSIE portfolio [10][15]. Whirlpool is a one-way hash function, uses Miyaguchi-Preneel compression scheme, and based on 512-bit block-cipher called W. From Figure 1, the first step in Whirlpool hashing algorithm is Message Padding: to ensure the message bits to be hashed are aligned on the appropriate bit boundary (change). The resulting data blocks are then organized in 8x8 array of bytes, where 10 iterations of W cipher is then performed on array individually. Finally, the Miyaguchi-Perneel compression is performed. This sequence is repeated for all the remaining blocks.

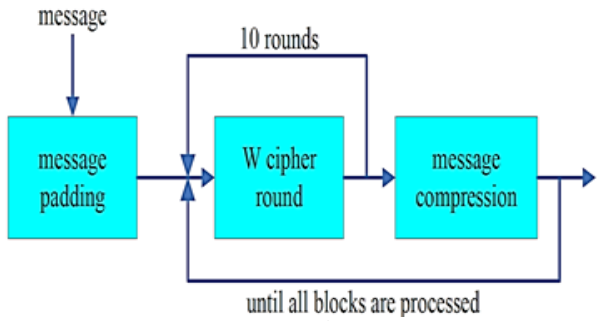

Fig. 1: High Level Block-Diagram of Whirlpool Hash Function [16].

The W cipher block, however, is comprised of four main stages (see Figure 2 for details). And they are [9] [10] [16] [17]:

1) Non-linear Stage (denoted by the symbol γ): the sole purpose of this stage is to eliminate avalanche effect, such that the differences among input bits should not propagate into similar differences among output bits. Second, the removal of linear correlation between input and output bits. A matrix of 16x16 containing all possible permutations of 8-bit values, known as S-box, is used where for each 4x4 S-box takes a 4-bit data input and produces a 4-bit output.
2) Cyclic Permutation (denoted by the symbol π): for each column represented by j, then cyclically downward shit it by j. Such that the 2nd column is shifted by one position, the 3rd by two, and so forth.

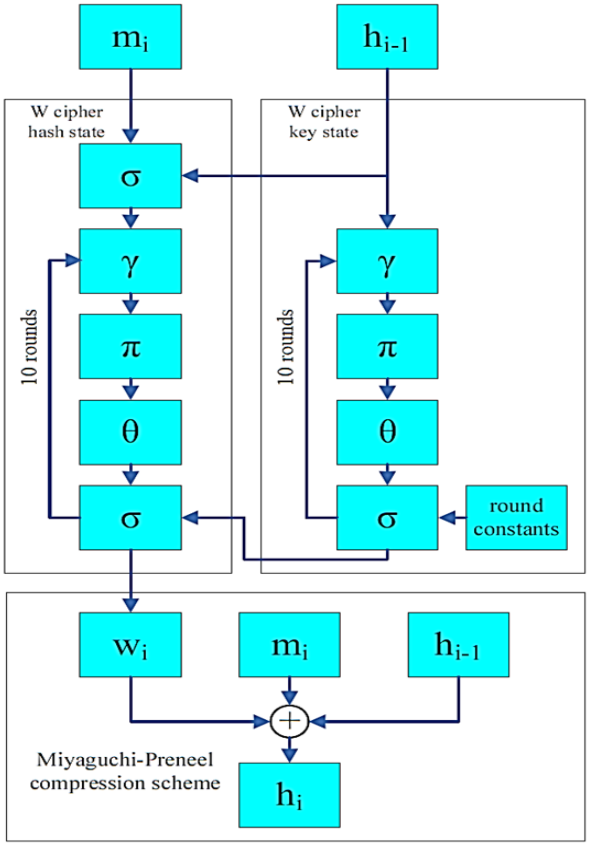

Fig. 2: Detailed View of Whirlpool Hash Function [16]

3) Linear diffusion (denoted by the symbol θ): the data generated from the previous stage is then multiplied by a circulant matric; C. a circulant matrix is defined as: each row vector is rotated one element to the right relative to the preceding row vector. It is important to mention that the matrix multiplication is performed in the Galois field, GF (28). The circulant matrix used in Whirlpool hash function is as follow:

$$C = \begin{bmatrix} 1 & 1 & 4 & 1 & 8 & 5 & 2 & 9 \\ 9 & 1 & 1 & 4 & 1 & 8 & 5 & 2 \\ 2 & 9 & 1 & 1 & 4 & 1 & 8 & 5 \\ 5 & 2 & 9 & 1 & 1 & 4 & 1 & 8 \\ 8 & 5 & 2 & 9 & 1 & 1 & 4 & 1 \\ 1 & 8 & 5 & 2 & 9 & 1 & 1 & 4 \\ 4 & 1 & 8 & 5 & 2 & 9 & 1 & 1 \\ 1 & 4 & 1 & 8 & 5 & 2 & 9 & 1 \end{bmatrix}$$

4) The key addition (denoted by the symbol σ): as the final stage, all the bits from previous step are with key generated by the key scheduler for this stage. Then the output is used as the input to next round of the ten rounds.

4. Whirlpool weaknesses and strengths

In numerous articles, researchers have demonstrated that the use of block-cipher based algorithms can have certain weaknesses [18]. For example, Joe summarizes few in as noted in [17]:

1) Block ciphers typically exhibit certain regularities. Therefore, it is possible to compromise many hash algorithms based on the defining properties of the underlying block cipher [19].
2) Block-cipher-based hash functions are significantly slower than specifically designed compression based hash functions.
3) One of the measuring principles of hash functions strength is the length of hash code in bits. Since the traditional block-

cipher-based algorithms are limited to 64 bits, therefore, the produced hash code is of questionable strength. However, with the advent of AES block-cipher algorithm, there has been grown interests by researchers to develop strong and good performing hash function. The introduction of Whirlpool hash function is intended to meet several goals that is comparable – if not better – than that found in nonblack-cipher-based hash functions. From [14] [16] [17], Whirlpool has the following features criteria:

1) The hash code length is 512 bits, the longest hash code available with SHA.
2) The overall structure shown to be resistant to the usual attacks on block-cipher-based hash codes [20], [21].
3) The underlying block cipher is based on AES, and is designed to be scalable for both software and hardware with minimal calculations footprint.
4) Security Goals: if we assume the resulting hash value is of n-bit length, then:

4.1. Generating a collision is of the order of 2n/2 executions of whirlpool

4.2. Given an n-bit value, the expected workload of finding a message that hashes tohat value is of the order of 2n executions of whirlpool.

4.3. Finding a second message that hashes to the same value is of the order of 2n executions of whirlpool.

4.4. It is infeasible to detect systematic correlations between any linear combination of input bits and any linear combination of bits of output result.

5. Whirlpool cryptanalysis

Since the adoption of Whirlpool hash function by NESSIE, several cryptanalysis and attacks reports were published testing Whirlpool security strengths and weaknesses. To review few, Whirlpool is resistant against differential attacks, due to infeasibility to detect any systematic correlations [14]. According to NESSIE published report titled "The Statistical Evaluation of the NESSIE Submission Whirlpool", the statistical results indicates that Whirlpool doesn't have randomness weakness [22], and also shown to be resistant to traditional block-cipher-based hash codes attacks [10].

Yet, and to my knowledge, there hasn't been a successful attack made against Whirlpool (full version, i.e. 10 rounds). Though the reason might be the Whirlpool hash function is rather new (the second edition was released in 2003), or for the reason that Whirlpool hasn't been adopted widely. Different type of attacks were performed on Whirlpool found in the literatures and since it is not the scope of this paper, I will only summarize the widely adopted ones. In addition, they are as follow:

1) Rebound Attack: Florian defines rebound attack as "the use of available degrees of freedom in a collision attack to efficiently bypass the low probability parts of a differential trail" [23]. Whirlpool hash function was a subject of study to several known rebound attacks. Florian successfully constructed collision attack on 4.5 rounds of Whirlpool hash function. The attack was also extended to 7.5 rounds of compression function of Whirlpool and to 8.5 rounds of Maelstrom hash function [23]. In another work, Mario proposed an improved version of rebound attack and collision, near-collision attack on the compression function of Whirlpool hash function, 4.5, 5.5, 7, and 7.5 rounds respectively. He also shows that Whirlpool does resist distinguisher based attack were an adversary has control over the key-input and the goal is to distinguish the block cipher W from an ideal cipher [24].
2) Preimage Attack: Preimage attack is better understood as how to find a message that has a specific hash value. Yu Sasaki proposed in [25] an improved version of Preimage attack with less computation complexity on Whirlpool function. The table below concludes and the obtained results [25].

Table 1: Results of Preimage Attack on AES Based Block Hash Functions [25]

5-Round Attacks		b r w	D_b D_r m	Compression Function	Second Preimage	Last Block	Preimage
chosen-key	-	5 4 2	128 96 128	2^{416},2^{96}	2^{465},2^{96}	2^{425},2^{96}†	2^{448},2^{96}†
	ml	4 3 1	128 64 128	2^{448},$O(1)$	2^{481},2^{32}	2^{457},2^{32}	
fixed-key	-	4 3 2	64 64 64	2^{448},2^{64}	2^{448},2^{64}†	2^{457},2^{64}	
	ml	5 4 2	64 48 128	2^{464},$O(1)$	2^{464},$O(1)$‡	2^{473},$O(1)$	2^{465},$O(1)$‡
fixed-key chosen padding	-	4 3 2	55 63 64	-	-	2^{457},2^{55}	
	ml	5 4 2	54 48 128	-	-	2^{464},$O(1)$‡	

6-Round Attacks		b r w g	D_b D_r D_g m	Compression Function	Second Preimage	Last Block	Preimage
chosen-key	-	6 4 2 6	256 64 192 256	2^{448},2^{256}	2^{481},2^{256}†	2^{457},2^{256}†	2^{481},2^{256}†
	ml	7 6 2 6	128 32 96 256	2^{480},$O(1)$	2^{497},2^{16}	2^{489},$O(1)$‡	
fixed-key	-	6 5 1 2	64 16 48 64	2^{496},2^{64}	2^{496},2^{64}	2^{505},2^{64}	
	ml	7 5 1 5	128 8 120 256	2^{504},$O(1)$	2^{504},$O(1)$‡	2^{513},$O(1)$	
fixed-key	-	6 4 1 3	118 16 96 128	-	-	2^{496},2^{112}	2^{504},$O(1)$‡
chosen padding	ml	7 6 1 3	54 8 48 128	-	-	2^{506},$O(1)$	

†: The attacks with the lowest computations.
‡: The attacks with the lowest memory.
Ml: The memoryless MitM attacks.
The researcher concluded that several risks
Exist when using similar diffusions for the key and data. However, he also shows that the ten rounds Whirlpool hash function still secure in practice.

3) Meet-In-The-Middle Preimage Attack: Yu Sasaki proposed applying recently developed meet-in-the-middle Preimage attack on three AES based hash function modes (namely Davies-Meyer, Matyas-Meyer-Oseas, and Miyaguchi-Preneel, 7 rounds each) to study the classical and important security notions [26]. Since Whirlpool is deeply based on AES (consists of 10 expanded AES rounds), the proposed analysis by Yu Sasaki can be directly applied on Whirlpool hash function. The below table details the results published in [26]:

Table 2: Results of Meet-In-Middle Attack [26]

Attack	Rounds	Key-size	Mode	Comp. Func. (Time, Mem.)	Hash (Time, Mem.)
Attacks on AES Hasing modes					
Collision	6	128/192/256	MMO,MP	$(2^{56}, 2^{32})$	$(2^{56}, 2^{32})$
2nd preimage	6	128/192/256	MMO,MP	$(2^{112}, 2^{16})$	$(2^{112}, 2^{16})$
2nd preimage	7	128/192/256	MMO,MP	$(2^{120}, 2^{8})$	$(2^{120}, 2^{8})$
Preimage	6	128/192/256	DM	$(2^{112}, 2^{16})$	$(2^{121}, 2^{16})$
Preimage	7	128/192/256	DM	$(2^{120}, 2^{8})$	$(2^{125}, 2^{8})$
Near collision	7	128/192/256	MMO,MP	$(2^{32}, 2^{32})$	$(2^{32}, 2^{32})$
Distinguisher	8	128/192/256	MMO,MP	$(2^{48}, 2^{32})$	-
q-multicollision	14	256	DM	$(q \cdot 2^{67}$, negl.)	-
Attacks on Whirlpool					
Collision	5	-	-	$(2^{120}, 2^{64})$	$(2^{120}, 2^{64})$
2ne Preimage	5	-	-	$(2^{504}, 2^{8})$	$(2^{504}, 2^{8})$
Near collision	7	-	-	$(2^{112}, 2^{64})$	$(2^{112}, 2^{64})$
Collision	7	-	-	$(2^{184}, 2^{8})$	-
Near collision	9	-	-	$(2^{176}, 2^{8})$	-
Distinguisher	10	-	-	$(2^{176}, 2^{8})$	-

The researcher successfully apply a Preimage attack on [7] rounds DM-AES hashing algorithm and second Preimage attack on [7] rounds of MMO-AES and MP-AES. He also suggested that the same could be used to generate second Preimage attack on reduced [5] rounds Whirlpool hash function.

6. Whirlpool hardware and software implementation

The need for secure communications among businesses or customers of various internet services is rising. Although, the strength of encryption is what mainly determines the use of any specific encryption/hashing algorithm, lately however, the factor of encryption speed (i.e. algorithm throughput measured by Megabits/seconds) is considered as important as encryption strength. High speed hashing/encryption algorithms became a necessity nowadays to cope with ever-growing internet users and online services customers varying from social media, online education, to e-banking.

It was found through literatures that widely adopted hashing algorithms are the subject of several studies were different hardware and software acceleration solutions are proposed and implemented. Such as using parallel processing through the use of multiple processors, or the use of special hardware such as FPGA circuits (Field Programmable Gate Array). As noted by Whirlpool designers in [14], the algorithm requires more hardware resources when compared with other hashing functions but it performs rather well in terms of throughput. And therefore, Whirlpool received several hardware and software implementation enhancements. In the following section, I will review a selected number of published articles in that regards.

1) Hardware Acceleration: P. Kitsos proposed FPGA-based implementation of Whirlpool. First, he proposed the implementation of S-Boxes to be either by using Look-Up-Tables (FPGA-LUT) or the use of Boolean Expressions (FPGA-BB). The proposed hardware and VLSI architecture consists of using combinational shifters to perform cyclic permutation, and more appropriate hardware based pseudo-code was given for the implementation of linear diffusion. Several implementations were proposed based on two solutions and compared with five well known different implementations of SHA-1, MD5, and SHA-2 from literatures. The first solution uses RAM to store data and key for each round whereas the second proposed solution uses two similar parallel data paths. The below table describes the obtained results [27].

Fig. 3: FPGA-Based Whirlpool Compared To Other Hashing Algorithms [27.]

Table 3: Results Obtained From [29]

Algorithm	Data Path Architecture		S-Box	Size	Operation frequency (MHz)	Throughput (Mbps)	Efficiency (Kbps/Gate)
Whirlpool	512-bit Parallel		$GF(2^8)$	132,213 gates	102.25	5,232	39.60
			Table	167,365 gates	187.27	9,588	57.29
			$GF(2^4)$	63,976 gates	113.82	5,818	90.94
			Table	88,329 gates	171.82	8,797	99.60
			$GF(2^4)$	62,107 gates	100.94	5,219	84.03
			Boolean	93,998 gates	154.08	7,889	74.10
	512-bit Interleaved		$GF(2^8)$	74,408 gates	127.06	3,098	41.03
			Table	105,819 gates	191.94	4,680	44.22
			$GF(2^4)$	38,911 gates	101.94	2,485	63.87
			Table	69,146 gates	182,82	4,457	64.87
			$GF(2^4)$	39,681 gates	101.83	2,483	62.57
			Boolean	63,934 gates	145.14	3,539	55.35
	512-bit Interleaved & Pipelined	(A)	$GF(2^8)$	78,872 gates	146.63	3,575	45.33
			Table	90,809 gates	200.40	4,886	53.80
			$GF(2^4)$	44,039 gates	127.88	3,118	70.80
			Table	54,608 gates	200.40	4,886	89.47
			$GF(2^4)$	43,624 gates	127.88	3,118	71.44
			Boolean	60,276 gates	171.82	4,189	69.50
		(B)	$GF(2^4)$	42,290 gates	128.04	3,122	73.82
			Table	52,788 gates	261.78	6,382	120.91
			$GF(2^4)$	40,648 gates	126.96	3,094	76.12
			Table	53,317 gates	208.33	5,079	95.27
SHA-256	Straightforward			11,022 gates	102.15	726	65.90
				15,400 gates	189.75	1,349	87.62
SHA-512	Straightforward			22,357 gates	101.94	1,186	53.06
				30,747 gates	169.20	1,969	64.03

In another work, Maire McLoone full Look-Up-Table (LUT) base design on high-speed hardware [28]. The proposed architecture aims at unrolling a complete W block-cipher round functionality in one clock cycle to reduce the overall latency of the design. The below figure depicts in comparison with proposed implementation [28].

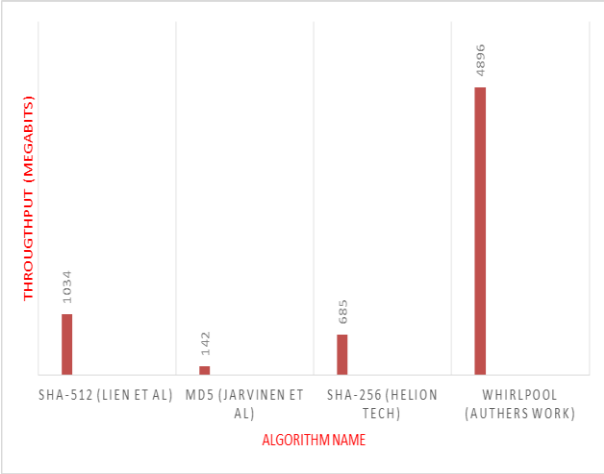

Fig. 4: Look-Up-Table Based Whirlpool Compared to Other Hashing Algorithms [28].

Last, the most promising implementation was published by Akashi Satoh in [29] with exceptional results. In summary, an architecture with three data paths, reduced number of XOR gates, sequential data flow, and the use of three different types of S-Boxes all helped to reduce the number of cycles required by half. Table 3 details the results of the proposed architecture.

It's important to mention that an attempt was made by Kamil, Pawel, and Maiusz to design SoPC (System on Programmable Chip) that implements the Whirlpool hash function. Their work and results can be found in [30].

3) Software Acceleration: Yedidya proposed the use of Parallel Table Lookup (PTLU) combined with novel cyclical permutation algorithm [31]. PTLU was previously used to accelerate block-cipher encryption algorithms, such as AES [33], and DES, Twofish, RC4, and 3DES [32]. The proposed method obtained speedup was 13.9x faster, equivalent to 7.2 cycle/byte (using RISC architecture and PTLU-128 bit module) compared to fastest SHA-2 512-bit implementation with 12 cycles/byte.

7. The proposed system

From the previously presented discussion, we can see that Whirlpool hash function is designed to be scalable on both software and hardware, such that it is highly unlikely to suffer from any bottleneck performance issues when compared with other hashing functions. Whirlpool shows immunity to five different attacks, with no successful attempt made against its full version, ten rounds. Whirlpool hash is a promising hash function which can be used and further developed to enhance the security of e-banking system.

Therefore, I propose using Whirlpool to encrypt customers' sensitive information in previously published e-bank solution in [34]. In my previously implemented work, we used MD5 message-digest function to encrypt customers' sensitive information, such as customer's account password, while card information is generated after user's completion of registration. The proposed system accepts Name, User Name, Password, Address, e-mail, and Phone Number. To enhance the security, the use of Whirlpool is introduced into three sections, see figure 5. First, during customers registration step, login, and third, while authentication and approval of financial transactions. The card number is algorithmically generated through hashing the information given by the user during the registration process, such that:

Card Number= Whirlpool (name, address, and phone number).

However, storing hashed version of all users' information will increase the space complexity of system database. Therefore, user's password, card number, and three digit card verification code are the only two fields opted to be hashed using Whirlpool. The rest of the information is stored as plan text. After the reception of Card Number, the user can proceed to login. It is also important to note that the results of Whirlpool function is truncated and only 16 digit is kept by using CRC algorithm [35].

Below is an example of card information and the resulted hash from using Whirlpool.

Card No.: 7465 1324 5590 3488

Whirlpool hash:

cde9c69188ed0d109012f2f53768e66278d700f0abcafa1f3c3bae73
cd818cc3d4600a61c26ab8aff007449923ae82c8ec35e2d02ed7c00
044c7c77d2e352046

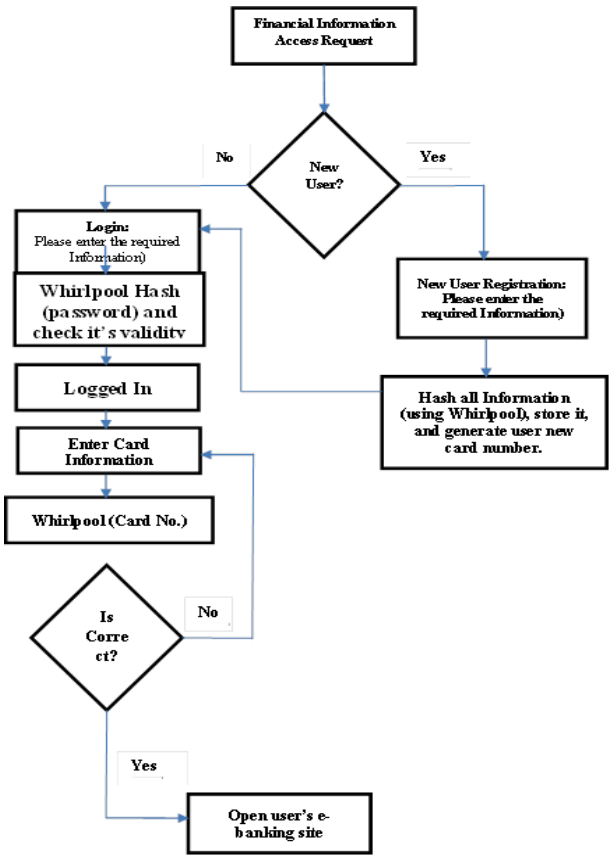

Fig. 5: Flowchart of the Proposed System.

8. Conclusion

The use of online banking services is increased gradually in daily life and will spread widely in coming years. Due to the sensitivity of information being exchanged between customers and e-banking services, layers of protection must be considered and implemented. Whirlpool is a promising hash function, and it is designed to be scalable than most modern hashing functions for both software and hardware. Although it requires more hardware resources footprint, but it has been shown that it performs better in terms of throughput in comparison to other hash functions. It resists several known attacks, and it was approved and suggested by NESSIE alongside SHA-256, SHA-384 and SHA-512.

In this paper, I proposed the use of Whirlpool to enhance and the security of e-banking systems through hashing sensitive information such as card information and account password. Whirlpool is also used to produce 16 digits card number selected from hashing the address, phone number, and the first seven characters of the user's name introduced during registration phase.

References

[1] Gaikwad S., Yadav A., Patil P., "The Study of E-Security in Internet Banking", International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 8, August 2015.

[2] Nwogu E. R., "Improving the Security of the Internet Banking System Using Three-Level Security Implementation", in IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: 2249-9555 Vol. 4, No.6, and December 2014.
[3] Choubey J., Choubey B., "Secure User Authentication in Internet Banking: A Qualitative Survey", International Journal of Innovation, Management and Technology, Vol. 4, No. 2, April 2013.
[4] Yazdanifard R., Fadzilah W., Alawa Y., Behora C., Sade A., "Electronic banking fraud; The need to enhance security and customer trust in online banking", International Journal in Advances in Information Sciences and Service Sciences, 3(10.61), 2011, pp. 505-509.
[5] Zeph A., Onyemachi O., Michael N., "Electronic banking and bank performance in Nigeria", West African Journal of Industrial & Academic Research Vol. 6, No. 1, 2013.
[6] Buhari B., Tambuwal A., "Security Enhanced Online Registration Prepaid Scratch Card Payment Approach", Journal of Engineering And Technology Research, Vol. 2, No. 6, pp. 53-59, 2014.
[7] Salma A., Devi C., Saranya V., "Smart Card for Banking with Highly Enhanced Security System", International Journal of Electronics and Communication Engineering (SSRG-IJECE), Vol. 1, Issue 2, 2014.
[8] Mridha M., Kamruddin N., Aloke K., Saha, Akhtaruzzaman A., "A New Approach to Enhance Internet Banking Security", International Journal of Computer Applications, Vol. 160, No. 8, 2017.
[9] Avik D., Subhasree D., Rajib G., "The Techniques behind the Electronic Signature based upon Cryptographic Algorithms", International Journal of Advanced Research in Computer Science, Vol. 5, No. 3, 2014.
[10] Stallings W.," Cryptography and Network Security Principles and Practices", Fourth Edition, http://www.inf.ufsc.br/~bosco.sobral/ensino/ine5680/material-cripto-seg/2014-/Stallings/Stallings_Cryptography_and_Network_Security.pdf.
[11] Kameswara R., Krishna Y., Kumar K., "An Image Authentication Technique Using Watermarking and Hash Function", International Journal of Advanced Research in Computer Science, Vol. 2, No. 2, pp. 86-89, 2011.
[12] Hanaek P., Kamil M., Jiri S., "E-banking security-comparative study." Security Technology, 42 Annual IEEE International Carnahan Conference, IEEE, 2008.
[13] CNBC Official website, https://www.cnbc.com/2017/02/01/consumers-lost-more-than-16b-to-fraud-and-identity-theft-last-year.html, retrieved at 4:30 PM, on 2/3/2018.
[14] Barreto P., Vincent R., "The Whirlpool hashing function." First, open NESSIE Workshop, Leuven, Belgium. Vol. 13. 2000.
[15] Preneel B., New European Schemes for Signature, Integrity and Encryption (NESSIE): A Status Report, Proceedings of the Fifth International Workshop on Practice and Theory in Public Key Cryptosystems: Public Key Cryptography. Lecture Notes in Computer Science, New York: Springer-Verlag, 2274, pp. 297–309.
[16] Zalewski P., "FPGA design and performance analysis of SHA-512, whirlpool and PHASH hashing functions", PhD Thesis, published in 5/1/2008.
[17] Stallings W... "The Whirlpool secure hash function." Cryptologia 30.1 (2006): 55-67.
[18] Francois-Xavier S., Piret G., Quisquater J., "Cryptanalysis of block ciphers: A survey", UCL Crypto Group, 2003.
[19] Miyaguchi, S., K. Ohta, and M. Iwate. 1990. Confirmation that Some Hash Functions are Not Collision Free, Proceedings, Advances in Cryptology—EUROCRYPT 090. New York: Springer-Verlag, pp. 326–343.
[20] Black, J., Rogaway P., Shrimpton T., "Black-Box Analysis of the Block-Cipher-Based Hash Function Constructions from PGV", Proceedings, Advances in Cryptology—CRYPTO 002, New York: Springer-Verlag, pp. 320–335, 2002.
[21] Preneel, B., Govaerta R., Vandewalle J., "Hash Functions Based on Block Ciphers: A Synthetic Approach". Proceedings, Advances in Cryptology—CRYPTO 093. New York: Springer-Verlag, 1993, pp. 368–378.
[22] Marcus S., "The Statistical Evaluation of the NESSIE Submission Whirlpool", Available: https://www.cosic.esat.kuleuven.be/nessie/reports/phase1/sagwp3-037_1.pdf
[23] Mendel F., "The rebound attack: Cryptanalysis of reduced Whirlpool and Grøstl", Fast Software Encryption. Springer, Berlin, Heidelberg, 2009.
[24] Lamberger M., "The rebound attack and subspace distinguishers: Application to Whirlpool", Journal of Cryptology, Vol. 28, No. 2, (2015): 257-296.
[25] Sasaki, Yu, et al. "Investigating fundamental security requirements on whirlpool: improved preimage and collision attacks." International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, 2012.
[26] Sasaki, Yu. "Meet-in-the-middle preimage attacks on AES hashing modes and an application to whirlpool." International Workshop on Fast Software Encryption. Springer, Berlin, Heidelberg, 2011.
[27] Kitsos, Paris, and Odysseas Koufopavlou. "Efficient architecture and hardware implementation of the Whirlpool hash function." IEEE Transactions on Consumer Electronics 50.1 (2004): 208-213.
[28] McLoone, Máire, Ciaran McIvor, and Aidan Savage. "High-speed hardware architectures of the Whirlpool hash function." Field-Programmable Technology, 2005. Proceedings. 2005 IEEE International Conference on. IEEE, 2005.
[29] Satoh, Akashi. "ASIC hardware implementations for 512-bit hash function whirlpool." Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on. IEEE, 2008.
[30] Krawczyk, Kamil, Paweł Tomaszewicz, and Mariusz Rawski. "Whirlpool SoPC Implementation-Hardware/Software Co-Design Example." International Journal of Electronics and Telecommunications 58.1 (2012): 21-26.
[31] Hilewitz, Yedidya, Yiqun Lisa Yin, and Ruby B. Lee. "Accelerating the whirlpool hash function using parallel table lookup and fast cyclical permutation." International Workshop on Fast Software Encryption. Springer, Berlin, Heidelberg, 2008.
[32] Fiskiran, A.M.: Instruction Set Architecture for Accelerating Cryptographic Processing in Wireless Computing Devices. PhD Thesis, Princeton University (2005).
[33] Fiskiran, A.M., Lee, R.B.: On-Chip Lookup Tables for Fast Symmetric-Key Encryption. In: Proceedings of the IEEE 16th International Conference on Application-Specific Systems, Architectures and Processors (ASAP), pp. 356–363. IEEE, Los Alamitos (2005).
[34] Al-Ani D., Shaban M., Noory R., "Billing system design based on internet environment", Editorial Preface, Vol. 3, No. 9, pp. 224 – 230, 2012.
[35] [Peterson W., Brown D., "Cyclic codes for error detection", Proceedings of the IRE, Vol. 49, No. 1, pp. 228-235, 1961.All the middle nodes represented as a blue circle surrounding them. When the energy of a specific node down to 18, the AODV evades that specific node. Here in this case there are [3] such nodes 0,2,10. The new route is shown below.