MSH Biswas Crypto-Intensive Technique

Md. Shamim Hossain Biswas*

* Department of Software Engineering, Daffodil International University,
102/1, Shukrabad, Mirpur Road, Dhanmondi, Dhaka-1207
Bangladesh

DOI: 10.14299/ijser.2019.10.01
https://doi.org/10.14299/ijser.2019.10.01

Abstract—The paper is devoted to constructing a cryptosystem. This is a combination of key generation algorithms, encryption algorithms, signature generation algorithms, signature verification algorithms, and decryption algorithms. The Michael O. Rabin Signature Scheme uses random padding to validate signatures. Similarly, the proposed techniques use the same padding system with additional quotients and remainders. The only difference is that the Michael O. signature is double or triple in some special cases, but the proposed model uses a 4-tuple signature system. One of the major problems with the Michael O. Rabin Cryptosystem is that it can generate the same ciphertext from different plaintexts as well as multiple plaintexts from a single ciphertext. To solve those issues, I proposed a mathematical solution, namely "A mathematical model for ascertaining the same ciphertext generated from distinct plaintext in the Michael O. Rabin Cryptosystem." But that model did not have an authentication system to verify the sender and validity of the message because there was no signature generation facility. The proposed crypto-intensive technique uses a two-time security key by slightly altering the Diffie-Hellman key exchange protocol. The advantage of the proposed crypto-intensive technique is that the sender generates a signature using encrypted text, and the intended receiver can retrieve plaintext from the signature through a signature verification system. The proposed crypto-intensive technique is secure against man-in-the-middle attacks. It is unforgeable, while Rabin's signature is forgeable in a forgery attack.

Index Terms—Cryptography, cryptosystem, key exchange protocol, modular arithmetic, Bezout's Coefficient, Extended Euclidean Algorithm, Chinese Remainder Theorem, Congruence, ASCII- Code, Quadratic reciprocity, Jacobi Symbol, Legendere Symbol.

1 Introduction

The Rabin signature algorithm is a method of digital signature. It was one of the first digital signature schemes that related the hardness of forgery directly to the problem of integer factorization. In the random oracle model, it was existensitally unforgeable, assuming the integer factorization problem was intractable and closely related to the Rabin crptosystem [1]. Since its publication in January 1979. A large amount of research was carried out by several researchers. [2]

A digital signature is a mathematical technique for verifying the authenticity of digital messages or documents. Authentication means that a valid digital signature gives a recipient very strong reason to believe that the message was created by a known sender, and integrity ensures that the message was not altered in transit. It is a standard element of most cryptographic protocols and is commonly used for software distribution, financial transactions, contract management systems, and to detect forgery or tampering, especially the intentional modification of products.

The term tempering refers to many forms of sabotage. The term authentification can refer to a computer communication protocol. A cryptographic protocol is specifically designed for the transfer of authentication data between two entities. The term data integrity can be refered to maintenance and the assurance of the accuracy and consistency of data over its entire life-cycle.

The encryption mechanism uses quadratic residue to produce cipher text. The encryption of a message $m \in z*_N$ is presented by $c = m^2 \ mod \ N$, where $N = p * q$ is a product of two prime numbers, and decryption is performed by solving the equation $x^2 = c \ modN$, N which has four roots;thus,s for complete decryption, further information is needed to identify m among these roots. It hasa vulnerabilityy to chosen-plaintextattacksk [3–6]. There isa timingg attack onthe modularr exponentiation algorithm [7]. The observer actually observes the exponentiation time of the algorithm. An attacker can reveal information abouta messagee because the execution time depends on the numberones in the binary representation of the message.

The decryption was accomplished by computing two square roots, Bezout's coefficient, using an extended Euclidean algorithm and combining them with the Chinese Remainder Theorem. Similarly to the RSA and ElGamal cryptosystems, the Michael O. Rabin cryptosystem is described in a ring under addition and multiplication modulo composite integers. One of the main disadvantages is that it generates four results during decryption, and extra effort is needed to sort out the right one out of the four possibilities. Michael O. Rabin's signature is vulnerable to a forgery attack.

The Rabin cryptosystem may also be used to create a signature by exploiting the inverse mapping. In order to sign m, the equation $x^2 = m \ mod \ N$ is solved and any of the four roots(S) can be used to form the signed message (m, S). However, if $x^2 = m \ mod \ N$ has no solution, the signature cannot

be generated directly. To overcome this issue, a random pad U is used until $x^2 = \text{m} * \text{U mod N}$ is solvable, and the signature is the triple (m, U, s). A verifier compares s^2 with $\text{m} * \text{U mod N}$ and accepts the signature as valid when these two numbers are equal.

Digital signatures employ asymmetric cryptography. In many instances, they provide a layer of validation and security to messages sent through a non-secure channel. Properly implemented, a digital signature gives the receiver reason to believe the message was sent by the claimed sender. Digital seals and signatures are equivalent to handwritten signatures and stamped seals, but properly implemented digital signatures are more difficult to forge than the handwritten type. It can also provide non-repudiation, meaning that the signer cannot successfully claim they did not sign a message while also claiming their private key remains secret. Further, some non-repudiation schemes offer a time stamp for the digital signature, so that even if the private key is exposed, the signature is valid. Digitally signed messages may be anything representable as a bitstring: Examples include electronic mail, contracts, or a message sent via some other cryptographic protocol. It typically consists of three algorithms:

1. *The key generation algorithm* selects a private key uniformly at random from a set of possible private keys. The algorithm outputs the private key and a corresponding public key.
2. *The signing algorithm* produces a signature.
3. *The Signature-verification algorithm* claims the message's authenticity.

Two main properties are required: in the beginning, the authenticity of a signature generated from a fixed message and fixed private key can be verified by using the corresponding public key. In addition to that, it should be computationally infeasible to generate a valid signature for a party without knowing that party's private key. A digital signature is an authentication mechanism that enables the generator of the message to attach a code that acts as a signature.

I have designed new crypto-intensive techniques based on the concept of square modular arithmetic from the Michael O. Rabin Signature Scheme. MSH Biswas encryption and decryption mechanisms [8]. Floor function and absolute value function. A shared secure key has been generated from the Diffie-Hellman key exchange protocol [9]. In the proposed cryptosystem, an entity A sends a 4-tuple signatur (Q, R, U, r_e) to B. The entity B verify the signature by $X(X + G) = Q * R * U \bmod N$. where X is the hash value of H(X), G is the generator, $Q = \lfloor m^2 / K_c \rfloor$, $\text{R} = m^2 \bmod K_c$.

If equality is found, the signature is accepted by the verifier and the message is revealed by $\left| \sqrt{Q * k_c + R} \right|$ and gets only one desired plain text, unlike Rabin's cryptosystem, in which she gets four different decryption results.

The rest of the paper is organized as follows: Section 1.1: Preliminaries, Section 1.2 gives an overview of Rabin's signature scheme; Section 2 gives a literature review; Section 3 provides an overview of the Diffie-Hellman key exchange protocol; Section 4 presents the author's contribution; Section 4.1 illustrates MSH. Biswas cryptointensive technique; Section 4.2 gives a comparison, Sections 5 and 6 give a conclusion and acknowledgement.

1.1 Preliminaries

Assuming that $N = p * q$ be a product of two odd primes p and q. Using the generalized Euclidean algorithm to compute the greatest common divisor between p and $q \in N$.

1. Initialize r_0 = q and r_1 = p.
2. Compute the following sequence of equations:
   $r_0 = q_1 r_1 + r_2$, where q_1 is quotient.
   $r_1 = q_2 r_2 + r_3$,
   $r_{n-3} = q_{n-2} r_{n-2} + r_{n-1}$,
   $r_{n-2} = q_{n-1} r_{n-1} + r_n$,
   until there is a step for which $r_n = 0$
   while $r_{n-1} \neq 0$.
3. The greatest common divisor is equal to r_{n-1}.

From which two integer numbers can be achieved after extending the theorem, and that is Bezout's coefficient $\lambda_1, \lambda_2 \in \mathbb{z}$, such that $\lambda_1 p + \lambda_2 q = 1$, are efficiently computed. Thus, setting $\psi_1 = = \lambda_2 q$ and $\psi_2 = \lambda_1 p$, so that $\psi_1 + \psi_2 = 1$, it is easily verified that ψ_1 and ψ_2 satisfy the relations.

$$\begin{cases} \psi_1 \psi_2 = 0 \ mod \ N \\ {\psi_1}^2 = \psi_1 \ mod \ N \\ {\psi_2}^2 = \psi_2 \ mod \ N \end{cases}$$

and that $\psi_1 = 1 \bmod \text{p}$, $\psi_1 = 0 \bmod \text{q}$, and $\psi_2 = 0 \bmod \text{p}$, $\psi_2 = 1 \bmod \text{q}$. According to the Chinese Remainder Theorem (CRT), using ψ_1 and ψ_2 every element a in $\mathbb{z}_N$ can be represented as $a = a_1 \psi_1 + a_2 \psi_2 \ mod \ N$, where $a_1 \in \mathbb{z}_p$ and $a_2 \in \mathbb{z}_q$ are calculated as $a_1 = a \ mod \ p \ and \ a_2 = a \ mod \ q$. The four roots $x_1, x_2, x_3, x_4 \in \mathbb{z}_N$ of $x^2 = C \ mod \ N$ represented as positive numbers, are obtained using the CRT from the roots u_1, $u_2 \in \mathbb{z}_p$ and $v_1, v_2 \in \mathbb{z}_q$ of the two equations $u_2 = C \ mod \ p$ and $v_2 = C \ mod \ q$, respectively.

The roots u_1 and $u_2 = p - u_1$ have different parities; likewise, v_1 and $v_2 = \text{q} - \text{v}_1$. If p is congruent 3 modulo 4, the root u_1 can be computed in deterministic polynomialtime as $\pm C^{(p+1)/4} \ mod \ p$, $\pm C^{(q+1)/4} \ mod \ q$. However, u_1 can be computed in probabilistic polynomial-time using Tonelli's algorithm [10] once a quadratic non-residue modulo p is known (this computation is the probabilistic part of the algorithm), or using the (probabilistic) Cantor-Zassenhaus algorithm [11, 12,

https://doi.org/10.14299/ijser.2019.10.01

13] to factor the polynomial $u^2 - c$ modulo p. Using the previous notations, the four roots can be written as

$$\begin{cases} x_1 = u_1\psi_1 + v_1\psi_2 \ mod\ N \\ x_3 = u_2\psi_1 + v_1\psi_2 \ mod\ N \\ x_3 = u_2\psi_1 + v_1\psi_2 \ mod\ N \\ x_4 = u_2\psi_1 + v_2\psi_2 \ mod\ N \end{cases}$$

Lemma 1: Let $N = p * q$ be a product of two prime numbers, and C be a quadratic residue modulo N. The four roots x_1, x_2, x_3, x_4 of the polynomial $x^2 - C$ are partitioned into two sets $X_1 = \{x_1, x_4\}$ and $X_2 = \{x_2, x_3\}$ such that roots in the same set have different parities, i.e., $x_1 = 1 + x_4 \ mod\ 2$ and $x_2 = 1 + x_3 \ mod\ 2$.

Proof. Since u_1 and v_1 have the same parity by assumption, x_1 and x_4 also have the same parity. The connection between x_1 and x_4 is shown by the following chain of equalities: $x_4 = u_2\psi_1 + v_2\psi_2 = (p - u_1)\psi_1 + (q - v_1)\psi_2 = -x_1 \ mod\ N = N - x_1$, because $p\psi_1 = 0 \ mod\ N$ and $q\psi_2 = 0 \ mod\ N$, and x_1 is less than N by assumption, thus $-x_1$ mod $N = N - x_1$ is positive and less than N. A similar chain connects x_2 and $x_3 = N - x_2$, because N is odd and thus x_1 and x_4 as well as x_2 and x_3 have different parities.

1.2 An illustration of Michael O. Rabin Signature

Signature Algorithm: The unique signature S is given by the following equation:

$$S = ((p^{q-2}\ H\ (m)^{\frac{q+1}{4}} \bmod q)\ p + (q^{p-2}\ H\ (m)^{\frac{p+1}{4}} \bmod p)\ q) \bmod N$$

Verification by $H\ (m) = s^2 \bmod N$, where N is the composite number of $p * q$. The signature can be verified by everyone, as N is a public key. A hash function H is collision resistant if it is hard to find that hash with the same output. If H is a collision resistant hash function, that does not mean that no collision exists; simply, they are hard to find. Such as,

$$H\ (m)^{\frac{p-1}{2}} mod\ p = 1 \text{ and } H\ (m)^{\frac{q-1}{2}} mod\ q = 1.$$

The cryptographic hash function is any mathematical equation. Message m is being hashed (encrypted). The hash value 1 is generated by using the private keys p and q. The same hash value from different hashed inputs is so called collision resistant, and the algorithm works as follows:

The workout example assumes that $p = 7$ and $q = 11$ uses $4k + 3$ prime formation. The public key $N = p * q = 77$. The Hashed message $H(m) = 20^2 \ mod\ 77 = 15$.
Let's see the collision-resistant hash value:

$15^{\frac{7-1}{2}} \bmod 7 = 1$ and $15^{\frac{11-1}{2}} \bmod 11 = 1$ that is vulnerable to collision attacks, because a collision attack on a cryptographic hash tries to find two inputs producing the same hash value.

$$S = ((7^{11-2}\ 15^{\frac{11+1}{4}} \ mod\ 11)7 + (11^{7-2}\ 15^{\frac{7+1}{4}} \ mod\ 7)\ 11) mod\ 77$$
$$= (8 * 9 \ mod\ 11)7 + 2 * 11) \ mod\ 77$$
$$= (6 * 7 + 2 * 11) \ mod\ 77 = 64.$$ So, the signature is unique.

Signature verification: $H\ (m) = s^2 \ mod\ 77 = 642 \ mod\ 77 = 15$. Since $H(m) = H(m)$, the signature is valid and accepted by the verifier.

Adescription of pairing signature algorithm is as follows.

Key generation:
In most presentations in modern cryptography, the algorithm is simplified by choosing b = 0, where b is actually the least prime (basement). The signer S chooses two primes p and q privately and computes the product $N = p * q$, where N is declared as a public key.

Signature generation:
Signer S picks random padding U to sign a message m and calculates $H\ (m) * U \ mod\ N$. S then solves the equation $X(X + b) = H\ (m) * U mod\ N$. where b is the basement (least prime). If there is no solution, S picks up a new pad U and tries again. Otherwise, the signature on m is (U, x)

Signatue Verification:
Given a message m and a signature (U, x), the verifier V calculates the equality of $X(X + b) \ mod\ N$ and $H(m) * U \ mod\ N$, where $X = H(X)$. If equality is found, the signature is accepted. As an example, assume that an entity A wants to send secret information $(X = 20)$ to another entity B using a valid signature. It first hashes the secret by $m^2 \ mod\ N = 20^2 \ mod\ 77 = 15$. Where N is a composite number of two secret private keys, moduli $p = 7$, moduli $q = 11$, both prime are Blum prime (4k+3). Public key or modulus $N = p * q = 7 * 11 = 77$. The Hashed value of 15 will be used to generate signatures.

Signing: Signer S chooses the number U probabilistically and see that the value of a random oracle modulo N matches any quadratic residue modulo N that is $X\ (X + b) \ mod\ N = m * U \ modulo\ N$. This process continues until both sides of the equation match the hash.

$X(X + b) \ mod\ N$	$m * U \ mod\ N$
$= 15\ (15 + 2) \ mod\ 77$	$= (15 * 17) \ mod\ 77$
$= 24$	$= 24$

The equation is solvable, which is why the signature on m is the pair (17, 15)

2 Literature Review

Rabin's signature on a message m may consist of a single and a pair (m, S). However, if there is $x^2 = m \ mod\ N$ has no solution, this signature cannot be directly generated. To overcome this obstruction, a random pad U was proposed by J. Pieprzyk et.al. [14], and attempts are repeated until $x^2 = m * U \ mod\ N$ is solvable, and thus the signature is the triple (m, U, S). A

https://doi.org/10.14299/ijser.2019.10.01

verifier compares $m * U \ mod \ N$ with S^2 and accepts the signature as valid when these two numbers are equal.

Hugh Cowie William [15] devised a modification of the Rabin system that allows the cryptographer to definitively decide which of the four-square roots the original message is. The Rabin-Williams Signature scheme relies on finding difficulties in square root. But it did not offer multiple signature facilities on a single document. It avoids the forgery vulnerability. While that scheme requires the use of two primes, respectively, congruent to 3 and 7 modulo 8. Moreover, in the Rabin-Williams scheme, a message cannot be signed twice in two different ways; otherwise, the factorization of N might get exposed.

Michele Elia, et.al. [16 -17] presented a modification of the H. C. William scheme based on the computation of a Jacobi symbol, where a deterministic pad is used for calculating non-Blum prime and Blum prime when m is QNR, as follows:

$$f_1 = \frac{m_1}{2}\left[1 - \left(\frac{m_1}{p}\right)\right] + \frac{1}{2}\left[1 + \left(\frac{m_1}{p}\right)\right]$$
$$f_2 = \frac{m_2}{2}\left[1 - \left(\frac{m_2}{q}\right)\right] + \frac{1}{2}\left[1 + \left(\frac{m_2}{q}\right)\right]$$
$$m = (m_1\psi_1 + m_2\psi_2) \ mod \ n$$

$x^2 = (m_1\psi_1 + m_2\psi_2)(f_1\psi_1 + f_2\psi_2) = (f_1 m_1\psi_1 + f_2 m_2\psi_2) \ mod \ N$, where $f_1 m_1$ and $f_2 m_2$ is a quadratic residue *modulo* p and *modulo* q respectively,
Since $\left(\frac{m_1}{p}\right) = \left(\frac{f_1}{p}\right), \left(\frac{m_2}{q}\right) = \left(\frac{f_2}{q}\right)$ so that
$\left(\frac{m_1 f_1}{p}\right) = \left(\frac{m_1}{p}\right)\left(\frac{f_1}{p}\right) = 1$ and $\left(\frac{m_2 f_2}{q}\right) = \left(\frac{m_2}{q}\right)\left(\frac{f_2}{q}\right) = 1$
$u = R^2 \ [f_1\psi_1 + f_2\psi_2]$

Signed message: (m, U, s)

Verification: Signer S verifies the equation $x^2 = \text{m} * \text{U mod N}$, since $L.H.S = R.H.S$ the signature is valid for message m. This is deterministically true as x^2 pre-calculated, but probabilistically, there is no such x value for which the $x^2 = \text{m} * \text{U mod N}$ is true.

Evgeny Sidorov et.al. [18] described a bug in the implementation of Rabin-Williams digital signature in the crypto++ framework, which is a popular cryptographic framework. The bug is in the misuse of blinding techniques that are aimed at preventing timing attacks on the digital signature system implementation. To fix the bugdoors, one should ensure that the value used for blinding is a quadratic residue modulo p and q. This condition guarantees that the blinding value will be removed at the unblinding step and won't affect the result of the signing procedure. Although the authors of crypto++ aimed at improving the security of the Rabin-Williams signature system implementation, they eventually made the system completely insecure, as admitted by authors themselves. The Rabin-Williams signatures became more efficient with the state-of-the-art modular–root signature system which was far beyond the simple signature system introduced by Daniel J. Bernstein [19].

Michele Elia et.al.[20] described a variant aimed at countering Rabin's signature vulnerability as follows:

Signed Message: $(m, U * R^2 \ mod \ N, \ S * R^3 \ mod \ N, \ R^4 \ mod \ N)$, so the signature is a fourtuple where U is the padding factor and R is a random number selection, Here S is the x's value for which the equation $x^2 = m * U \ mod \ N$ is true. It is clearly seen that x and U are both unknown numbers which have to be chosen by entity A in order to generate a signature.

Verification: Compute $(S * R)^2 \text{mod N}$ and $m * U * R^2 * R^4 \ mod \ N$; the signature is valid if and only if the aforesaid two numbers are equal.

As a workout example, assuming preprocessed values for $m/ = 15, \ U * R^2 = 25 * 3^2 \ mod \ 77 = 71, S * R^2 = 12 * 3^2 = 108 \ mod \ 77 = 48$ and $3^4 \ mod \ 77 = 4$. So, the signature $(15, 71, 48, and \ 4)$ is a 4-tuple. The verification computations are as follows:

1. $(12 * 3^3)^2 \ mod \ 77 = (12^2 * 3^6) \ mod \ 77 = 25$ and
2. $15 * 25 * 3^2 * 3^4 mod \ 77 = 25$

Counter forgery 4-tuple signature (15, 71, 48, and 41) verification is successful, so the signature is valid and accepted.

Jaweria Usmani, et.al [21] proposed a secure gateway discovery protocol using the Rabin Signature Scheme in MANET that ensures confidentiality in heterogeneous environments. The registration process was included to remove the malicious nodes. This protocol removes the threat of anti-confidentiality, anti-authentication, and anti-duplication. The efficiency of this protocol is shown through the AVISPA tool.

Chaoyang Li et.al. [22] proposed an efficient ID-based signature scheme based on Rabin's cryptosystem by using the forking lemma theorem. This scheme has a lower exponential operation. It is secure against existential forgery under adaptively chosen identity and message attacks in the random oracle model.

Daniel Bleichenbacher [23] presents a method to compress Rabin signature. The rabin signatures and compressed signatures are equally difficult to forge. Compression requires a continued fraction expansion and takes time $0(log(n)_2)$. Decompression requires two multiplications and an inverse over $\mathbb{Z}\mathbb{Z}/_n\mathbb{Z}\mathbb{Z}$, a square root of $\mathbb{Z}\mathbb{Z}$ which requires time $0(log(n)_2)$.

https://doi.org/10.14299/ijser.2019.10.01

3 Diffie-Hellman Key exchange protocol

The first published public-key algorithm appeared in the seminal paper by Diffie and Hellman that defined public-key cryptography. It is generally referred to as the Diffie-Hellman key exchange protocol. A number of commercial products employ this key exchange technique. The purpose of the algorithm is to enable two users to securely exchange a key that can then be used for subsequent encryption and decryption of messages. The algorithm itself is limited to the exchange of secret values. The Diffie-Hellman algorithm's effectiveness depends on computing discrete logarithms.

Global public elements:
N is a prime number that can define a domain, so-called curve area or elliptic curve, α is a primitive root of N such that α < N.

Key generation for user A: Select private key X_a, such that $X_a < N$. Calculate public *key* $Y_a = \alpha^{xa} \ mod \ N$
Key generation for user B: Select private key X_b such that $X_b < N$. Calculate public key $Y_b = \alpha^{xb} \ mod \ N$
Secret key for user A: $K = (Y_b)^{xa} \ mod \ N$
Secret key for user B: $K = (Y_a)^{xb} \ mod \ N$
Let's consider a workout example, an integer number N = 353 is the domain size, and its primitive root α = 3. Alice A and Bob B select secret keys A = 97 and B = 233, respectively.
Each of them computes public key:
A compute $X = 3^{97} \ mod \ 353 = 40$.
B compute $Y = 3^{233} \ mod \ 353 = 248$.
Secret key computation: They exchange public keys between each other.
A compute $K = (Y)^A \ mod \ 353 = 24897 \ mod \ 353 = 160$.
B compute $K = (X)^B \ mod \ 353 = 40233 \ mod \ 353 = 160$.

4 Author Contributions

Presumably, let an entity A communicates information to another entity B. Entities A and B both should have some confidentiality. Both entities A and B create a shared secret key using the aforesaid key exchange protocol, and then both of them add an additional pre-negotiated key with the newly generated key. Alice encrypts secret information with a secret key so that an unauthorized entity cannot presume and disclose real information. A encrypts information and chooses an equivalent residuum to generate a signature by solving an equation $R(R+G) = Q * R * U \ modulo \ K_c$, where R is quadratic residue modulo K_c, G is generator, Q is quadratic quotient modulus K_c and U is selected arbitrarily to justify truthiness of the equation. Alice sends only 4-tuple signature (Q, R, U, r_e) to receiver Bob. The entity B verifies the signature by checking truthfulness of equation $r_e = \mathrm{Q} * \mathrm{R} * \mathrm{U}$ modulo K_c . B opens the message by $\left|\sqrt{Q * k_c + R}\right|$ iff above equation is true; otherwise, reject.

Key Generation Algorithm:

$$
\begin{aligned}
K &= (Y_b)^{xa} \ mod \ N \\
&= (\alpha^{xb} mod \ N)^{Xa} \ mod \ N \\
&= (\alpha^{xb})^{Xa} \ mod \ N \\
&= \alpha^{xb \,.\, Xa} mod \ N \\
&= (\alpha^{Xa})^{xb} \ mod \ N \\
&= (\alpha^{Xa} mod \ N)^{Xb} mod \ N \\
&= (Y_a)^{xb} mod \ N
\end{aligned}
$$

Current key $(K_c) = Prenegotiated \ symmetric \ key + Exchange \ key \ (K)$

Encryption Algorithm: To encrypt a message (m), following logic is considered:
$Q = \lfloor m^2 / K_c \rfloor$
$\mathrm{R} = \mathrm{m}^2 \bmod K_c$
$C = (Q, R)$, where $Q = Quotient, R = Residuum, and \ C = Cipher$.

Signing Algorithm: To sign a message (m), the signer (S) tries to find the solution of the equation $\mathrm{R(R+G)} = \mathrm{C} * \mathrm{U} \bmod K_c \Rightarrow \mathrm{R(R+G)} = \mathrm{Q} * \mathrm{R} * \mathrm{U} \bmod K_c$
∴ Signature is 4-tuple (Q, R, U, r_e), where $r_e = \mathrm{R(R+G)} \bmod K_c$, $r_e = Equivalent \ residuum$

Decryption Algorithm:

The verifier (V) verifies the signature by calculating the equation. $r_e = \mathrm{Q} * \mathrm{R} * \mathrm{U} \bmod K_c$ and opens message by $\left|\sqrt{Q * k_c + R}\right|$.

4.1 A summary of proposed Cryptosystem

Assuming that Alice wants to send secret information ($A = 65$) to Bob using a valid signature, she first hashes the secret message ($m^2 \ mod$) with the shared secret key (K_c) and floor value of m^2/K_c. She sends a signed and hashed message to Bob. Bob reveals the message after verifying the signature of the sender. The entire process is as follows:

Key generation process:

Pre-negotiated secret key $(P_{sk}) = 17 < 128(ASCII)$			
Alice		Bob	
known	un-known	Known	un-known
$Curve \ area \ N = 113$		✓	
$Generator = 5$		✓	
$key \ P = 7$	$q = 11$	$key \ q = 11$	$P = 7$

https://doi.org/10.14299/ijser.2019.10.01

$A = 5^7(113) = 42$		$B = 5^{11}(113) = 34$	
$A = 34^7(113) = 40 = k_a$		$B = 42^{11}(113) = 40 = k_b$	
$K_c = K_a + P_{sk} = 40 + 17 = 57$		$K_c = K_b + P_{sk} = 40 + 17 = 57$	

Current key$(K_c) = exchange\ key + prenegotiated\ key$. Alice and Bob have acquired a new key by mixing the exchange key with their predefined key. A predefined key protects against a man-in-the-middle attack as they both exchange their keys publicly. An exchange key may be obtained by an eavesdropper, but they do not have access to the pre-negotiated key. She will lock the written message by using a new key. Then, she will signg the encrypted message and send to Bob.

Message encryption process:
Message $A = 65(ASCII)$
Quotient$(Q) = \left\lfloor (65)^2/_{57} \right\rfloor = 74$
Residuum (R) $= (65)^2\ mod\ 57 = 7$
Cypher text$(C) = Q, R$

Signature generation process:
$R(R+G) = H(m) * U\ mod\ K_c \Rightarrow R(R+G) = C * U\ mod\ K_c$
$\Rightarrow R(R+G) = Q * R * U\ mod\ K_c$
$\Rightarrow 7(7+5) = 74 * 7 * 45\ mod\ 57$
$\Rightarrow 54 = 54\ (modulo\ 57)$.
Let equivalent residue r_e= 54, Hence Signature is a 4-tuple (74, 7, 45, and 54). Bob verifies his signature by computing the equality of the equation as follows:

Signature verification process:
$\Rightarrow r_e =$ Q * R * U mod K_c.
$\Rightarrow r_e = 74 * 7 * 45$ mod 57
$\therefore\ r_e =$ 54 (mod 57) verified.

Message opening process:
Now, Bob reveals the message by applying square root to the result of Q $* K_c +$ R, after that, he accepts the absolute value as the desired plaintext.
Decryption=D=$\left|\sqrt{Q * k_c + R}\right| = \left|\sqrt{74 * 57 + 7}\right| = \left|\sqrt{4225}\right| = 65 = A\ (reveal)$.

4.2 Comparison

Advantage of Michael O. Rabin Signature: This signature actually contains several interesting features: The signature is possible using every pair of primes. Different signatures on the same document are different. The verification needs only two multiplications, and therefore it is fast enough to be used in the authentication protocol.

Disadvantage of Michael O. Rabin Signature: It is vulnerable to forgery attacks. It is relatively easy to compute S^2 mod N, choose any message m/ and compute the multiplicative inverse of the $m/$(hash value of m); compute $U/ = S^2 * m/^{-1}$mod N and forge the signature as ($m/^{-1}$, $U/$, s) without knowing the factorization of N.

Advantage of MSH Biswas Cryptointensive Technique: The signature is generated using one part of a hashed message, a generator and a random padd computing equation of $R(R+G) = C * U$ modulo two step security keys. Where R is a quadratic rsidue modulo K_c and C is a pair of ciphertexts in the context of the MSH Biswas encryption mechanism. It is a strong aginst man-in-the-middle attack, a forgery attack. It does not rquire computing four roots. It requires less time and complexity compared to the Michael O. Rabin public key signature scheme.

5 Conclusion

The proposed MSH-Biswas crypto-intensive techniques are efficient for solving four-to-one mapping signatures. Its first objective is to identify each ciphertext separately because modular arithmetic can generate the same ciphertext from different plaintexts. The proposed model can efficiently identify each ciphertext separately generated from modular reduction arithmetic. Its second objective is to verify the sender and validate the message through a signature verification system where both authentication and integrity elements have been successfully deployed to implement a signature scheme. The proposed key generation technique is derived from the Diffie-Hellman key exchange protocol, but there was a security vulnerability in the symmetric key generation stage (the man in the middle attack) because it could not authenticate the participants. The proposed crypto-enabled techniques ensure security by combining an exchange key with a pre-negotiated key that is unknown to the adversary. I would like to leave an encryption scheme for future readers to make a concrete (single) ciphertext that can uniquely identify similar quadratic residues separately generated from different inputs.

6 Acknowledgement

I am very grateful to my family members who supported me financially to conduct this study because, without their financial support, love, and affection, this work could not be carried out. I am very grateful to well-wishers, friends, and families. I thank Md. Maruf Hassan for his inspirational advice and Dr. Md. Mostafijur Rahman (Assistant Professor, Department of Software Engineering, Daffodil International University) for insightful discussion during the preparation of this paper. This work is part of the thesis and research activities of Daffodil International University for the academic curriculum fulfillment of an MSc in software engineering.

https://doi.org/10.14299/ijser.2019.10.01

References

[1] Michael. Rabin, probabilistic algorithm, algorithm and complexity, recent results and new directions, J. F. Traub, editor, ACADEMIC PRESS, INC. New York San Francisco, December 1976, pp. 21-40
[2] Michael. Rabin, Digitized signatures and public key functions as in tractable as factorization, technical report MIT-LCS-TR-212, MIT la boratory for computer science, 1979
[3] J.A. Buchmann, Introduction to Cryptography, Springer, New York, 1999.
[4] J. Hoffstein, J. Pipher, J.H. Silverman, An introduction to mathematic cal cryptography, Springer, New York, 2008
[5] A.J. Menezes, P.C. van Oorschot, S.A. Vanstone, Handbook of Ap plied Cryptography, CRC Press, Boca Raton, 1997.
[6] B. Schneier, Applied cryptography, Wiley, 1996
[7] PaulC.Kocher, "Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and OtherSystems ", Cryptography Research, Inc, Advances in Cryptology - CRYPT0 '96, LNCS 1109, pp. 104-113, 1996, Spnnger-Verlag Berhn Heidelberg 1996
[8] Md Shamim Hossain Biswas, "A mathematical model for ascertain ing same ciphertext generated from distinct plaintext in Michael O. Rabin Cryptosystem", In: International Journal of Scientific & Engi neering Research Volume 10, Issue 6 (2019). ISSN 2229-5518, DOI: 10.14299/ijser.2019.06.08.
[9] Diffie–Hellman key exchange protocol was introduced by Malcolm John Williamson (British mathematician and cryptographer) in1974.
[10] E. Bach, J. Shallit, Algorithmic Number Theory, MIT, Cambridge Mass., 1996.
[11] D.G. Cantor, H. Zassenhaus, A new Algorithm for Factoring Polyno mials over Finite Fields, Math. Comp., Vol. 36, N. 154, April 1981, pp.587- 592
[12] M. Elia, D. Schipani, Improvements on the Cantor-Zassenhaus Factori zation Algorithm, to appear in Math. Bohem.
[13] J. von zur Gathen, J. Gerhard, Modern Computer Algebra, Cam bridge Univ. Press, 1999.
[14] J. Pieprzyk, T. Hardjono, J. Seberry, Fundamentals of Computer Secu rity, Springer, New York, 2003.
[15] H.C. Williams, A modification of the RSA public-key encryption procedure, IEEE Trans. on Inform. Th., IT-26(6), November 1980, pp.726-729.
[16] Michele Elia, Matteo Piva, Davide Schipani, the Rabin cryptosystem revisited, (2011)
[17] Michele Elia, Davide *Schipani*, On the Rabin signature, In: *Journal of Discrete Mathematical Sciences* and Cryptography 16(6) .
[18 Evgeny Sidorov, Kandex LLC, "Breaking the Rabin-Williams digital signature system Implementation in crypto++ library" In: Journal of cryptology, iacr.org, April 16, 2015.
[19] Daniel J. Bernstein, "RSA and Rabin-Williams signatures: the state of the art", In: EUROCRYPT'08 Proceedings of the theory and applications of cryptographic techniques 27th annual international conference on Advances in cryptology Pages 70-87, Jan 31, 2008.
[20] Michele Elia∗, Matteo Piva†, Davide Schipani‡ The Rabin cryptosys tem revisited, arXiv: 1108.5935v3 [math.NT] 3 Nov 2013
[21] Jaweria, Usmani, Jay Prakash, A Secure Gateway Discovery Protocol Using Rabin Signature Scheme in MANET. October 2017, In: International Journal on Communications Antenna and Propagation 7(5):439 DOI: 10.15866/irecap. v7i5.12581
[22] Chaoyang Li1, Xiangjun Xin2◉ and Xiaolin Hua, Efficient ID-based Rabin Signature without Pairings, In: International Journal of Multi media and Ubiquitous Engineering Vol.12, No.3 (2017), pp.75-80 doi:10.14257/jimue.2017.12.3.08
[23] Daniel Bleichenbacher, Compressing Rabin Signatures, In: T. Oka mo to (Ed.): CT-RSA 2004, LNCS 2964, pp. 126–128, 2004. c Springer-Verlag Berlin Heidelberg 2004, Bell Labs – Lucent Tech nologies

Author Biography

Nane: Md. Shamim Hossain Biswas
MSc in Software Engineering, Daffodil International University
ORCID: 0000-0002-4595-1470 Cell: +8801531262445.
E-mail: shamim44-165@diu.edu.bd, shamim.ak.pico@gmail.com

https://doi.org/10.14299/ijser.2019.10.01