Sign-Then-Encrypt Scheme with Cramer-Shoup Cryptosystem and Dissanayake Digital Signature

Rahmad Bahri[1], Mohammad Andri Budiman[2] and Benny Benyamin Nasution[3]
[1]*Master of Informatics Program, Universitas Sumatera Utara, Medan, Indonesia*
[2]*Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Medan, Indonesia*
[3]*Politeknik Negeri Medan, Medan, Indonesia*

Keywords: Signcryption, Dissanayake, Cramer-Shoup, Running Time, Avalanche Effect.

Abstract: Exchanging information in the era of Internet-based technology still has security violations such as disclosure, modification, or destruction that make everyone worry about the exchange of information. The Dissanayake digital signature is part of asymmetric cryptography based on the factorization of prime numbers and has interesting mathematical properties. The property of such mathematics is that the sum of 2 odd numbers is a multiple of 4. The Dissanayake digital signature does not generate signatures directly through messages. The Cramer-Shoup algorithm is an asymmetric cryptographic algorithm that proved to be the first effective scheme to withstand Adaptive Chosen Ciphertext Attack (ACCA) attacks compared to existing cryptographic systems. The Cramer-Shoup algorithm is an extension algorithm of the Elgamal algorithm. This paper will implement a sign then-encrypt scheme using Dissanayake digital signature and Cramer-Shoup algorithm and analyze algorithms based on execution time and the avalanche effect. Based on simulation results, the characters' length affects the execution time. The result of the process shows the length of the character linear with execution time. From the results of the avalanche effect simulation, Dissanayake digital signature got the average value of the avalanche effect of 51%, and the Cramer-Shoup algorithm got the average value of the avalanche effect of 49%. Implementing the sign-then-encrypt scheme can maintain security by encrypting and guaranteeing authenticity by adding a digital signature.

1 INTRODUCTION

Protecting data from disclosure, modification, or destruction is essential in the development of today's technological era. The number of security breaches that continues to increase makes everyone who communicates through the global network (Internet) worry about protecting their data. With the current development of technology, computer security is needed because computer security rests on confidentiality, integrity, and availability (Stallings and Bauer, 2012; Manna et al., 2017). —many techniques in protecting data at this time by designing a sound computer security system—can be used in cryptography (Panhwar et al., 2019). Cryptography is a technique that relies on mathematics to secure information such as confidentiality, integrity, and entity authentication. Confidentiality, integrity, and authenticity are the basic requirements of asymmetric cryptography (Abood and Guirguis, 2018; Molk et al., 2021; Gençoğlu, 2019).

Asymmetric cryptography, or public key cryptography, has two keys (Public Key and Private Key) used in the implementation process (Hossain et al., 2013). In 1976, Whitfield Diffie and Martin Hellman publicly introduced the concept of public key cryptography (Lydia et al., 2021). Asymmetric cryptographic algorithms use two keys to perform the encryption and decryption process. Asymmetric cryptography distributes public keys by publishing and private keys stored by their owners or kept secret (Khan et al., 2018; Maqsood et al., 2017).

Many security schemes have been implemented by researchers in the field of cryptography, one of which is signcryption, a scheme to achieve confidentiality and authenticity (Elkamchouchi et al., 2018; Kasyoka et al., 2021). In this case, confidentiality is achieved by encryption schemes on asymmetric cryptography, while authenticity can be achieved by digital signature schemes (Pandey, 2014),(Matsuda et al., 2009).

One of the asymmetric cryptographic algorithms

Bahri, R., Budiman, M. and Nasution, B.
Sign-Then-Encrypt Scheme with Cramer-Shoup Cryptosystem and Dissanayake Digital Signature.
DOI: 10.5220/0012444900003848

In *Proceedings of the 3rd International Conference on Advanced Information Scientific Development (ICAISD 2023)*, pages 131-138
ISBN: 978-989-758-678-1

that can be used is Cramer-Shoup, a modern cryptographic algorithm(Jain, 2017). Compared to the current cryptographic systems, Cramer-Shoup proved the first effective scheme without Adaptive Chosen Ciphertext Attack (ACCA) attacks (Liu et al., 2014). The Cramer-Shoup algorithm is an extension algorithm of the Elgamal algorithm. The Cramer-Shoup algorithm has additional elements to guard against security attacks.

One of the digital signature schemes that can be used is the Dissayanke algorithm. The Dissayanake digital signature scheme algorithm is part of an asymmetric cryptographic algorithm. There are two main functions: forming digital signatures (signature generation) and checking or verifying the validity of digital signatures (signature verification). The Dissayanake algorithm is a very fast and simple digital signature scheme, similar to RSA digital signature, which uses prime number factorization. This digital signature scheme indirectly creates a signature from the message directly and adds a hash function to strengthen the scheme (Dissanayake, 2019)

There are parameters for analyzing the security of a cryptographic algorithm, one of which is the avalanche effect, which is a value that indicates that small changes made to the plaintext or key will cause significant changes in the resulting ciphertext. The avalanche effect value, greater than 0, states a change between the plaintext sender and the plaintext recipient by looking at the number of different bits in the plaintext. Cryptographic algorithms that can have a significant avalanche effect value, then the level of security in an algorithm can be said to be good.

This paper will apply a signcryption scheme with the Dissanayake digital signature scheme algorithm and the Cramer-Shoup algorithm. The signcryption security scheme to be used is the sign-then-encrypt method. To measure the security level of the scheme developed, researchers will use the Avalanche effect

2 LITERATURE REVIEW

2.1 Cryptography

Cryptography is a technique that relies on mathematics to secure information such as confidentiality, integrity, and entity authentication (Vollala et al., 2021). Cryptographic techniques are used for encoding by hiding or encoding data (Rachmawati and Budiman, 2018). Encryption and decryption are important cryptography processes. Converting plaintext into ciphertext is called encryption and vice versa because decryption is converting ciphertext into plaintext.

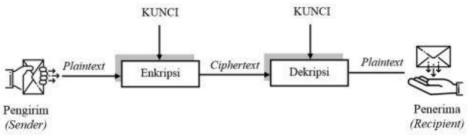

Figure 1: Working of Encryption and Decryption.

2.2 Asymmetric Cryptography

In 1976 Whitfield Diffie and Martin Hellman publicly introduced the concept of public key cryptography (Lydia et al., 2021). Asymmetric cryptography, or public key cryptography, has two keys (public key and private key) used in the implementation process (Vollala et al., 2021). Asymmetric cryptography distributes the public key by publishing, and the private key is stored by the owner or kept secret. Some examples of asymmetric algorithms are Cramer-Shoup, Elgamal, RSA, and others.

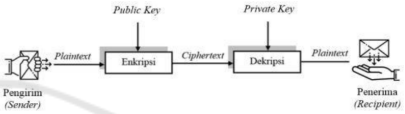

Figure 2: Working of Asymmetric Cryptography.

2.3 Digital Signature

The digital signature is one scheme with cryptographic value by relying on the message and the message's sender. Digital signatures guarantee data integrity, detect changes or modifications to messages sent, and perform authenticity. Creating a signature on a message can be done by performing a digital signature with a hash function.

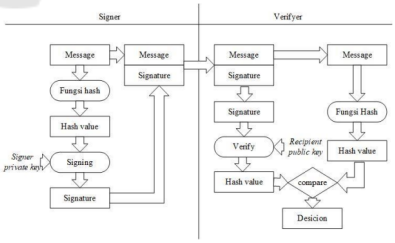

Figure 3: Digital Signature Scheme.

2.4 Hash Function and Data Integrity

A hash function is often referred to as a one-way function. The hash function has long been used in the world of computer science. A hash function is a mathematical calculation that takes the variable length of an input string and converts it to a fixed length. The

resulting output is commonly referred to as a hash value.

Using a hash algorithm makes it easy to calculate the hash value on a message and makes it impossible to modify the message without composing the resulting hash value. The hash function can detect any modifications to the sent message. The hash function can guarantee data integrity based on specific settings

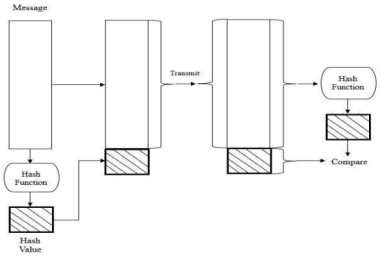

Figure 4: Message Integrity Checking.

2.5 Signcryption

This study used the sign-then-encrypt method. The message (plaintext) in the signature uses the Dissayanake Digital Signature algorithm to generate a signature against the message. After that, the message and signature are encrypted using the Cramer-Shoup algorithm. This method combines digital signatures and public key cryptography to enhance security. This process is called authentication and encryption.

2.6 Avalanche Effect

Avalance Effect is one of the mathematical properties in cryptography that can determine whether or not a cryptographic algorithm is good. A cryptographic algorithm can be said to satisfy the criteria avalanche effect if the input bits change, and then it is likely that all bits can change by at least half

$$Avalancheeffect = \frac{differentnumberofbits}{totalnumberofbits}100 \quad (1)$$

Cryptographic algorithms that can have a significant avalanche effect value, then the level of security in an algorithm can be said to be good.

3 METHOD

3.1 Dissayanake Digital Signature Algorithm

Maheshika W.D.M.G Dissanayake developed the Dissanayake digital signature scheme in 2019. The Dissanayake digital signature is proposed based on the factorization of prime numbers and simple mathematical properties. This scheme has a mathematical property: the sum of 2 different odd numbers is a multiple of 4.

This digital signature scheme uses a public key, which is a (e, n) A large prime number less than e, and n is a variable chosen by the signer r.

In Dissayanake Digital Signature Algorithm, there are three essential processes: key generation, signing, and verifying.

1. Key generation
   a Choose two different primes for the values p and q.
   b Counting n=p×q.
   c Counting $\phi(n) = (p-1)(q-1)$.
   d Select a prime number d so that gcd $(d, \phi(n) = 1$, is used as the sender's private key.
   e Calculate the public key e so that e.d $mod\phi(n) \equiv 1$
   f m is a message
   g generate the hash value of message m using the hash h function
   h h(m)=M, M is a hash value
   i Select an integer r so that $(M)+r)mod4 \equiv 0$
   j Find the odd number (a) so that (a)=$\frac{M+r-2}{2}$
2. Signing
   Count $S \equiv a^d modn$
3. Verifying
   a Calculate $S^e modn \equiv a'$
   b Find the odd number (a) so that $(a) = \frac{M+r-2}{2}$
   c If, then $a' = a$, the signature is valid if then the $a' \neq a$ the signature is invalid.

3.2 Cramer-Shoup Algorithm

The Cramer-Shoup cryptosystem is a symmetric cryptographic algorithm invented by two scientists, Ronald Cramer, and Victor Shoup, in 1998. The Cramer-Shoup algorithm proved to be the first effective scheme that resisted adaptive chosen ciphertext attacks compared to cryptographic systems that existed at the time (Elkamchouchi et al., 2018). The Cramer-Shoup algorithm is an extension algorithm of the Elgamal algorithm. There are three essential processes in Cramer-Shoup Algorithm: key generation, encryption, and decryption.

1. Key Generation
   a At the key generation stage, the process occurs as follows: Generated g1, g2 where the two

numbers are the subset q. It is a group of very large primes.

b Six random numbers with a range of 0 to q-1 were selected. x1, x2, y1, y2, z

c Then the variables c, d, and h are calculated by the following equation:

$$c = g1x^1 g2^{x2} \quad (2)$$

$$d = g1^{y1} g2^{y2} \quad (3)$$

$$h = g1^z \quad (4)$$

From the above calculations.
public key (g1,g2,c,d,h), private key (x1,x2,y1,y2,z)

2. Encryption
The message m is converted to a G element. Selected random numbers in the range 0 to q. Then the value is calculated u1, u2, e, a, and v using the public key. Here is:

$$u1 = g1^r \quad (5)$$

$$u2 = g2^r \quad (6)$$

$$e = h^r m \quad (7)$$

$$v = c^r d^{rxa} \quad (8)$$

3. Decryption
Calculate for decryption $m = \frac{e}{u1^z}$

4 SIMULATION AND RESULTS

4.1 Simulation

The test that will be carried out on the Dissanayake digital signature algorithm is how the algorithm is in the key generation, signing process, and verifying process. Key generation, encryption, and decryption will be performed in the Cramer-Shoup algorithm testing.

4.1.1 Key Generation Process Results Dissanayake Digital Signature

The result of the key generation process by the sender using the Dissanayake digital signature algorithm produces a public key and a private key

4.1.2 Results of Testing the Signing Process of the Digital Signature Algorithm

After generating the key in the Dissanayake digital signature algorithm, the following process is to generate the signature.

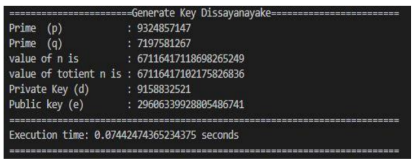

Figure 5: Key Generation Results on The Simulation.

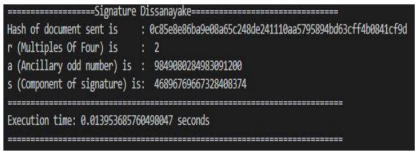

Figure 6: The Results of Generating Signatures in The Simulation.

4.1.3 Results of Verifying Process Testing Dissanayake Digital Signature

The results of the proses of verifying on the Dissanayake digital signature verifier algorithm take the value of the signature to be verified using a public key.

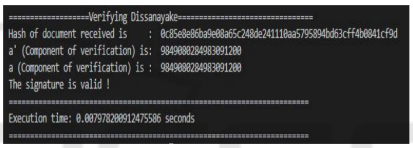

Figure 7: The Results of Verifying the Simulation.

4.1.4 Cramer-Shoup Algorithm Key Generation Process Test Results

The result of the key generation process by the sender using the Cramer-Shoup algorithm generates a public key and a private key.

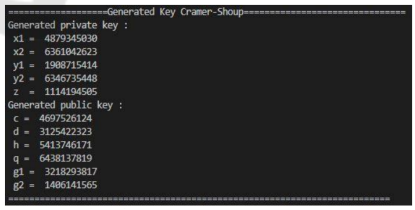

Figure 8: Key Generation Results on the Simulation.

4.1.5 Cramer-Shoup Algorithm Encryption Process Test Results

After generating the key in the Cramer-Shoup algorithm, the following process is that it can generate the ciphertext

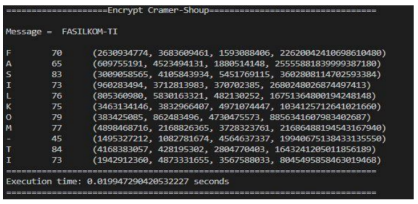

Figure 9: Results of Generating Ciphertext in The Simulation.

4.1.6 Results of Testing the Decryption Process of the Cramer-Shoup Algorithm

The result of the decryption process is that the recipient can use his private key to decrypt the ciphertext.

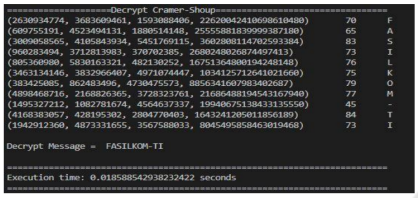

Figure 10: The Result of The Decryption in The Simulation.

The simulation results of the running time to analyze the signing and verifying process time and the encryption and descriptive processes are based on experiments with different plaintext lengths. Units of execution time use seconds. The hardware specifications used in this study to build and test the scheme are as follows.

Table 1: Hardware Specifications.

No	Specifications	Information
1	Processor	Intel® Core TM i3-7020U CPU 2.30GHz
2	RAM	8 GB DDR 4
3	Desktop	14 inch
4	OS	Windows 10 Home Single Language 64 Bit

4.2 Simulation of Signing Dissanayake Digital Signature

The signing process using the Dissanayake digital signature algorithm calculates the processing time based on the plaintext length, which is 50 characters, 100 characters, 200 characters, 300 characters, and 400 characters. The results of the signing process time of the Dissayanyake digital signature algorithm can be seen in Table 2.

Based on the processing time with character length, it shows the average result of the signing process of the Dissanayake digital signature algorithm,

Table 2: Simulation Results for Signature Generation.

Plaintext length	Execution time (second)			
	1	2	3	Avarage
100	0.0025	0.0049	0.0040	0.0038
200	0.0039	0.0051	0.0070	0.0053
300	0.0050	0.0061	0.0090	0.0067
400	0.0070	0.0091	0.0050	0.0070
500	0.0081	0.0060	0.0100	0.0080

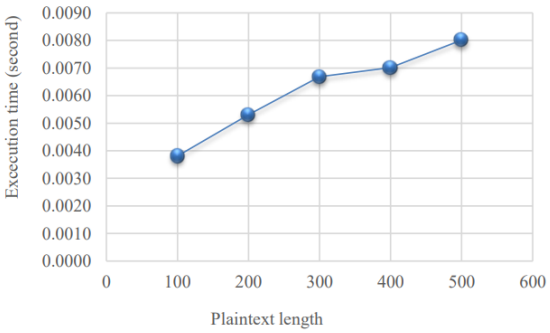

Figure 11: Graph of Digital Signature Generation Results.

where the average process time starts from 0.0038 s to 0.0080 s based on tests with different character lengths. Based on Figure 12 illustrates a graph of digital signature generation results. The process results show the length of the linear plaintext with execution time.

4.3 Simulation of Verifying Dissanayake Digital Signature

The verifying process uses the Dissanayake digital signature algorithm to calculate the processing time of the plaintext length, which is 50 characters, 100 characters, 200 characters, 300 characters, and 400 characters. The results of the verifying process execution time of the Dissayanyake digital signature algorithm can be seen in Table 3.

Table 3: Simulation Results for Verifying.

Plaintext length	Execution time (second)			
	1	2	3	Avarage
100	0.0021	0.0020	0.0020	0.0020
200	0.0030	0.0030	0.0010	0.0023
300	0.0050	0.0030	0.0020	0.0033
400	0.0060	0.0060	0.0040	0.0053
500	0.0071	0.0051	0.0061	0.0061

Based on the processing time with character length, it shows the average result of the verifying process of the Digital signature Dissanayake algorithm. On tests with different character lengths, the

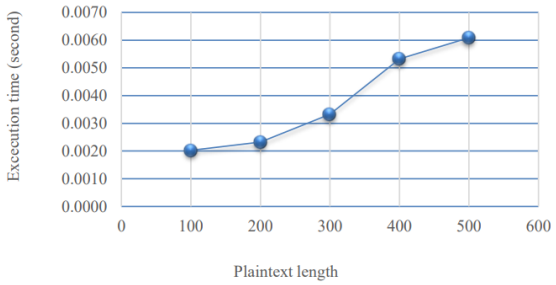

Figure 12: Graph of Verifying Results.

average process execution time starts from 0.0020 s to 0.0061 s. Figure 13 illustrates a graph of the results of the verifying process. The process results show the length of the linear plaintext with execution time.

4.4 Simulation of Encryption Cramer-Shoup Algorithm

The encryption process uses the Cramer-Shoup algorithm to calculate the processing time based on the plaintext length, which is 50 characters, 100 characters, 200 characters, 300 characters, and 400 characters. The results of the encryption process execution time of the Cramer-Shoup algorithm can be seen in Table 4.

Table 4: Simulation Results for Encryption.

PL	Execution Time (second)			
	ke-1	ke-2	ke-3	Avarage
100	0.1586	0.1576	0.1152	0.1438
200	0.2234	0.2514	0.3137	0.2628
300	0.3136	0.3518	0.4219	0.3624
400	0.4226	0.6499	0.5183	0.5303
500	0.6213	0.7639	0.6627	0.6827

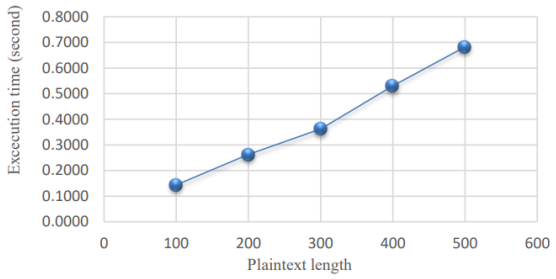

Figure 13: Graph of Results for Encryption.

Based on the processing time with character length, it shows the average result of the encryption process of the Cramer-Shoup algorithm, where the average process time starts from 0.1438 s to 0.6827 s based on tests with different character lengths. Figure 14 illustrates a graph of the results of the encryption process. The process results show the length of the linear plaintext with execution time.

4.5 Simulation of Decryption Cramer-Shoup Algorithm

The decryption process uses the Cramer-Shoup algorithm to calculate the processing time based on the plaintext length, which is 50 characters, 100 characters, 200 characters, 300 characters, and 400 characters. The results of the decryption process execution time of the Cramer-Shoup algorithm can be seen in Table 5.

Table 5: Simulation Results for Decryption.

PL	Execution time (second)			
	1	2	3	Avarage
100	0.1267	0.0818	0.0973	0.1019
200	0.1476	0.1620	0.2389	0.1828
300	0.2254	0.2503	0.3391	0.2716
400	0.2952	0.5022	0.3821	0.3932
500	0.3381	0.5852	0.6074	0.5102

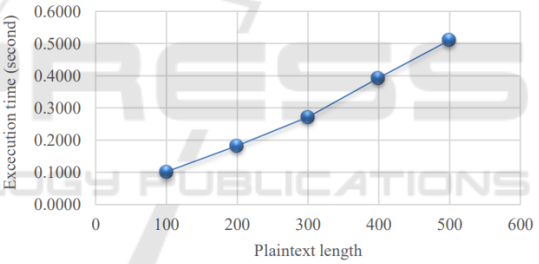

Figure 14: Graph of Results for Decryption.

Based on the processing time with character length, it shows the average result of the encryption process of the Cramer-Shoup algorithm, where the average process execution time starts from 0.1019 s to 0.5102 s based on tests with different character lengths. Based on Figure 15 illustrates a graph of the results of the decryption process. The process results show the length of the linear plaintext with execution time.

4.6 Simulation of Sign-then-Encrypt Scheme

The signcryption scheme process uses the Dissanayake digital signature algorithm and the Cramer-Shoup algorithm with sign-then-encrypt and decrypt-then-sign methods to calculate the processing execution time based on the length of the plaintext, namely 50 characters, 100 characters, 200 characters, 300

characters, and 400 characters. The sign-then-encrypt and Decrypt-then-sign execution can be seen in Table 6.

Table 6: Simulation Results for Sign-Then-Encrypt and Decrypt-Then-Verifying.

PL	Execution time (second)			
	sign-then-encrypt		Decrypt-then-verifying	
50	0.0738	0.0520	0.0973	0.1019
100	0.1341	0.0926	0.2389	0.1828
200	0.1846	0.1375	0.3391	0.2716
300	0.2687	0.1993	0.3821	0.3932
400	0.3454	0.2582	0.6074	0.5102

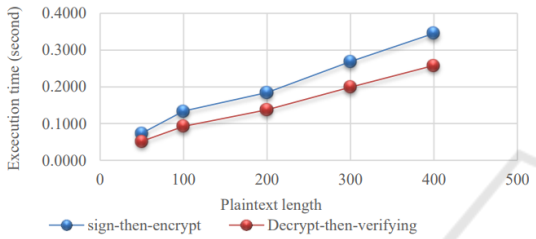

Figure 15: Graph of Results for Sign-Then-Enccrypt and Decrypt-Then-Verifying.

Based on fig.16 shows that the execution time in the sign-then-encrypt process is longer than the running time of the decrypt-then-verifying process because the sign-then-encrypt process takes execution time to generate prime numbers and generate keys. Based on Figure 16 illustrates a graph of the results of the sign-then-encrypt and decrypt-then-verifying processes. The process results show the length of the linear plaintext with execution time.

4.7 Simulation of Avalanche Effect

Simulation in this study on the signature results of the signing process Dissanayake digital signature and ciphertext from the encryption process of the Cramer-Shoup algorithm to obtain values on the avalanche effect. Simulation is conducted to obtain the average value of the avalanche Effect results.

Table 7: Simulation Results of Avalanche Effect Dissanayake Digital Signature.

	Algoritma Dissanayake Digital Signature		
	1	2	3
Plaintext	RAHMAD	BAHRI	FASILKOM
Signature	482387741	2058316320	2612119605
Plaintext modifications	RAHMA4	4AHRI	FASILKO8
Signature modifications	615076365	899842372	1597165780
Avalanche effect (%)	49.75	52.51	48.15
AVERAGE	50.14		

Dissanayake Digital Signature obtains an average value of 50.14% based on the avalanche effect simulations.

Table 8: Simulation Results of Avalanche Effect Cramer-Shoup Algorithm.

	Algoritma Cramer-Shoup		
	1	2	3
Plaintext	RAHMAD	BAHRI	Fasilkom
Ciphertext	2393999649	2692183725	27562015
Plaintext modifications	RAHMA4	4AHRI	Fasilkot
Ciphertext modifications	116590405	2347432728	12651684
Avalanche effect (%)	42.35	52.95	55.56
AVERAGE (%)	50.29		

Based on the simulations using the avalanche effect, the Cramer-Shoup algorithm obtains an average value of 50.29%.

5 CONCLUSION

Based on research, the digital signature Dissayanake algorithm has a process to guarantee authenticity by using verifying methods and hash functions. Verifying is performed on formulas using the recipient's public key to generate a hash value used as input in the signing process and can provide entity authentication. The use of $S^e mod n \equiv a'$ hash functions $h(m) = M$ can guarantee the authenticity of the information. In the process, the recipient can re-run the hash function to generate a new hash value, which is then compared with the hash value received from the sender. Based on the simulation results on the signing and verifying process with Dissanayake digital signature and the decryption encryption process with the Cramer-Shoup algorithm, the characters' length affects the execution time. The process's result shows the linear character's length with execution time. From the results of the avalanche effect simulation, Dissanayake digital signature got the average value of the avalanche effect of 50.14%, and the Cramer-Shoup algorithm got the average value of the avalanche effect of 50.29%. Implementing the sign-then-encrypt scheme can maintain security by encrypting and guaranteeing authenticity by adding a digital signature.

REFERENCES

Abood, O. and Guirguis, S. (2018). A survey on cryptography algorithms. *Int. J. Sci. Res. Publ*, 8(7).

Dissanayake (2019). A novel scheme for digital signatures.

Elkamchouchi, H., Takieldeen, A., and Shawky, M. (2018). An advanced hybrid technique for digital signature scheme. *Conf. Electr. Electron. Eng. ICEEE*, pages 375–379,.

Gençoğlu, M. (2019). Importance of cryptography in information security. *IOSR J. Comput. Eng*, 21(1):65–68,.

Hossain, M., Hossain, M., Uddin, M., and Imtiaz, S. (2013). International journal of advanced research in performance analysis of test generation techniques.

Jain, D. (2017). Data security using cramer – shoup algorithm in cloud computing. *Ijarcce*, 6(3):930–933,.

Kasyoka, P., Kimwele, M., and Angolo, S. (2021). Cryptanalysis of a pairing-free certificateless signcryption scheme. *ICT Express*, 7(2):200–204,.

Khan, A., Basharat, S., and Riaz, M. (2018). Analysis of asymmetric cryptography in information security based on computational study to ensure confidentiality during information exchange. *Int. J. Sci. Eng. Res*, 9(11):992–999,.

Liu, Z., Yang, X., Zhong, W., and Han, Y. (2014). An efficient and practical public key cryptosystem with cca-security on standard model. *Tsinghua Sci. Technol*, 19(5):486–495,.

Lydia, M., Budiman, M., and Rachmawat, D. (2021). Factorization of small rprime rsa modulus using fermat's difference of squares and kraitchik's algorithms in python. *J. Theor. Appl. Inf. Technol*, 99(11):2770–2779,.

Manna, S., Prajapati, M., Sett, A., Banerjee, K., and Dutta, S. (2017). 3rd ieee int. conf. res. *Comput. Intell. Commun. Networks, ICRCICN*, 2017-Decem:327– 331,.

Maqsood, F., Ahmed, M., Mumtaz, M., and Ali, M. (2017). Cryptography: A comparative analysis for modern techniques. *Int. J. Adv. Comput. Sci. Appl*, 8(6):442–448,.

Matsuda, T., Matsuura, K., and Schuldt, J. (2009). Efficient constructions of signcryption schemes and signcryption composability.

Molk, A., Aref, M., and Khorshiddoust, R. (2021). Analysis of design goals of cryptography algorithms based on different components. *Indones. J. Electr. Eng. Comput. Sci*, 23(1):540– 548,.

Pandey, A. (2014). An efficient security protocol based on ecc with forward secrecy and public verification.

Panhwar, M., Khuhro, S., Panhwar, G., and Memon, K. (2019). Saca: A study of symmetric and asymmetric cryptographic algorithms. *IJCSNS Int. J. Comput. Sci. Netw. Secur*, 19(1):48,. Online]. Available:.

Rachmawati, D. and Budiman, M. (2018). Using the rsa as as an asymmetric non-public key encryption algorithm in the shamir three-pass protocol. *J. Theor. Appl. Inf. Technol*, 96(17):5663–5673,.

Stallings, W. and Bauer, M. (2012). Computer security.

Vollala, S., Ramasubramanian, N., and Tiwari, U. (2021). Energy-efficient modular exponential techniques for public-key cryptography.