

Utilization of Asmuth-Bloom Algorithm In Data Security Using Secret Sharing Protocol

Heri Santoso¹, Aidil Halim Lubis², Ali Darta³

¹Departement Computer Science, Faculty of Science and Technology, UIN Sumatera Utara Medan, Indonesia

herisantoso@uinsu.ac.id, aidilhalimlubis@uinsu.ac.id, alidarta@uinsu.ac.id

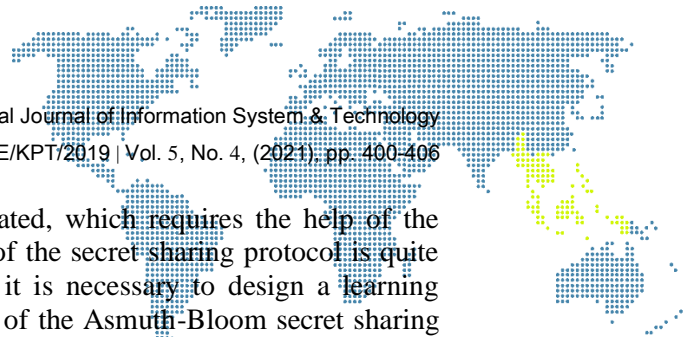
Abstract

Conventional cryptographic algorithms cannot be applied to solve a plaintext (message) into several ciphertexts because conventional cryptographic algorithms can only produce a ciphertext from a plaintext (message). In this case, a cryptographic protocol can be applied, namely the Asmuth-Bloom secret sharing protocol. The methodology used to analyze and design application software and learning the Asmuth-Bloom secret sharing protocol is the Waterfall method. This research is intended to produce a learning software that is able to display the working process of the Asmuth-Bloom secret sharing protocol and the application of the Asmuth-Bloom secret sharing protocol. The working procedure described by the software includes the key formation process, the shadow formation process and the shadow merge process. The software also provides an interface to do the process of splitting a text file into n shadow files and merging m shadow files into the original text file. In addition, the software also provides basic theories related to the two cryptographic protocols and reports the results of the calculation process are stored in a text file with *.txt extension.

Keywords: Cryptography, Secret Sharing, Asmuth-Bloom

1. Introduction

Cryptography is a science that studies how to keep data or messages safe when sent from sender to recipient without experiencing interference from third parties. Conventional cryptographic algorithms (such as DES, IDEA, RSA, LUC, and so on) can be used to secure data in the communication process. However, the conventional cryptographic algorithm can only generate a cipher text (encoded message) from a plaintext (original message). This causes the conventional algorithm cannot be applied if someone wants to break a message into several different cipher texts. The splitting of messages into several cipher texts is mainly applied when the confidential data is owned by a group of people or an organization, so to obtain the confidential data, the approval of a number of people is required. In this case, cryptographic protocols can be applied, such as the secret sharing protocol (secret sharing by applying the (m, n) -threshold scheme), secret splitting (secret sharing) and so on. Both secret sharing and secret splitting protocols can be used to break a message into n different cipher texts that can be shared with n people. The difference is that to get the original message, the secret splitting protocol requires the n pieces of the cipher text, while the secret sharing protocol applies the (m, n) -threshold scheme, which only requires m cipher texts from a total of n available cipher texts to form the message again, where $m < n$. In the cryptography literature, there are many algorithms that apply the concept of of the secret sharing protocol. One of the algorithms of the secret sharing protocol is the Asmuth-Bloom scheme. This algorithm uses prime numbers and random numbers to increase its security. In addition, this algorithm also requires n sequences of numbers in which must meet certain requirements. The process of forming the cipher text of the Asmuth-Bloom algorithm is relatively easy, that is, only by performing the modulo addition operation. Meanwhile, the process of



forming the original message is relatively complicated, which requires the help of the Chinese Remainder theorem. The working process of the secret sharing protocol is quite long and complicated if it is calculated manually, it is necessary to design a learning software that is able to display the working process of the Asmuth-Bloom secret sharing protocol. The problem in this research is how to design a software that is able to explain the work rules of Asmuth-Bloom algorithm and application of application of Secret algorithm Sharing Asmuth-Bloom.

2. Research Methodology

2.1. Cryptography

The most influential development in the history of cryptography occurred in 1976 when Diffie and Hellman published the newspaper Directions in Cryptography. This paper introduces the revolutionary concept of public-key cryptography and provides a new and highly ingenious method of key exchange, the security of which is based on the difficulty of dealing with discrete logarithms. Although the authors were not aware of the practical utility of public-key encryption schemes then, the idea was clear and generated great interest and activity in the cryptographic community. In 1978 Rivest, Shamir, and Adleman invented the first public-key encryption and signature scheme, now known as RSA. The RSA scheme is based on another difficult problem in mathematics, the difficulty of factoring large integers. The application of difficult problems in mathematics to cryptography stimulated efforts to find more efficient methods of factoring. Cryptography can be defined as the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, authentication of data senders / recipients, and data authentication. Cryptography is the science and art of storing messages, data, or information securely. Cryptography studies how to keep important information secret into a form that cannot be read by anyone and return it to its original information using various existing techniques so that the information cannot be known by any unauthorized party. There are two important processes in cryptography that play a role in keeping information secret, namely encryption (encryption) and decryption (decryption). Encryption is the transformation of data (plaintext) into an almost unreadable form (ciphertext) without sufficient knowledge. The purpose of encryption is to make sure confidentiality by keeping information hidden from anyone who is not the owner or interested in the information, even those who have access to encrypted data. While decryption is the opposite of encryption, namely the transformation of data that has been encrypted (ciphertext) back to its original form (plaintext). The process of encryption and decryption generally requires the use of a secret amount of information, which is often called a key.

2.2. Cryptography system

There are two types of cryptographic systems, namely symmetric key cryptography and asymmetric key cryptography.

Symmetric key cryptography is often called secret-key cryptography. This system is also known as conventional encryption or single-key encryption. In this system, the same algorithm is used for the encryption/decryption process using the same key.

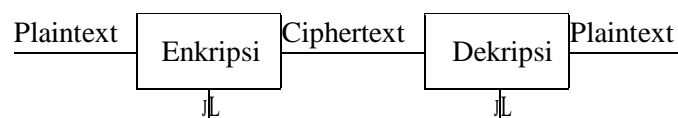
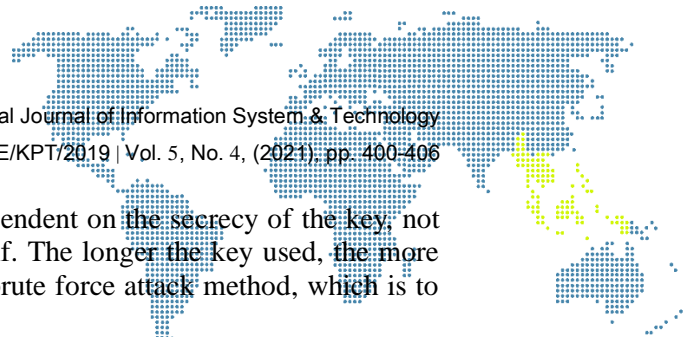


Figure 1. Symmetric Key Cryptography

The security of this system depends on several factors, namely:

- a) The encryption algorithm must be strong enough that it is impractical to decrypt a message with only the ciphertext.



- b) The security of symmetric encryption is dependent on the secrecy of the key, not the secrecy of the encryption algorithm itself. The longer the key used, the more difficult it is to guess the key by using the brute force attack method, which is to try all possible keys.

Asymmetric cryptography system (asymmetric key cryptography) is often called public-key cryptography (public-key cryptography). This means that the keywords used for encryption and decryption are different. These keys are interconnected with each other. With the public key one can encrypt the message but cannot decrypt it, only the person who has the private key can decrypt the message. Asymmetric algorithms can make sending messages more secure than symmetric algorithms. According to Stalling [1], the simple public-key encryption process involves the following four stages:

- Each user on the network creates a pair of keys to be used as encryption and decryption keys for messages to be received.
- The user publishes his encryption key by placing his public key in a public place. The other key pairs are kept secret.
- If user A wants to send a message to user B, he will encrypt the message using user B's public key.
- When user B wants to send a message to user B, he will use his own private key. No one else can decrypt the message because only B knows B's private key.

The public key is the key that is not hidden (may be known by others) and is used in the encryption process. The private key is a secret key that should not be known by others and is used in the decryption process.

2.3. Cryptography Mathematical Concept

a. Primes

Public keys in many cryptographic methods require prime numbers. A prime number is an integer greater than one which has a factor of one and the number itself [2]. The wrong way to get prime numbers is to generate random numbers and then try to factor them. The correct way is to generate a random number and then test if it is a prime number.

b. Modular Arithmetic

Modular arithmetic plays an important role in integer computing, especially in cryptographic applications. The operator used in modulo arithmetic is mod. The mod operator gives the remainder of the division [2].

c. Chinese Remainder Theorem

In the first century, a Chinese mathematician named Sun Tzu asked the following question: "Find an integer which when divided by 5 leaves 3, when divided by 7 leaves 5, and when divided by 11 leaves 7". Sun Tzu's question can be formulated into a linear congruent system:

$$x = 3 \pmod{5} \quad (1)$$

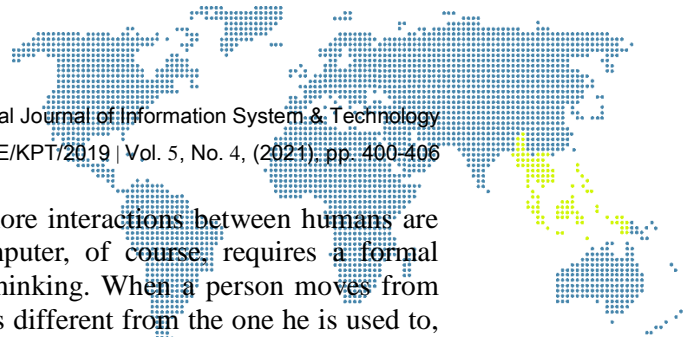
$$x = 5 \pmod{7} \quad (2)$$

$$x = 7 \pmod{11} \quad (3)$$

The following Chinese Remainder theorem will be used to find solutions for linear congruent systems as above [2].

2.4. Cryptography Protocol

A protocol is a series of steps involving two or more parties and designed to complete an activity [2]. Cryptographic protocol is a protocol that uses cryptography. This protocol involves a number of cryptographic algorithms, but in general the purpose of the protocol is more than just confidentiality. The participating parties may wish to share some of their secrets to calculate a value, generate random sequences, or sign contracts simultaneously. The use of cryptography in a protocol is primarily intended to prevent or detect



eavesdropping and cheating. Currently, more and more interactions between humans are carried out through computer networks. This computer, of course, requires a formal protocol to do what humans normally do without thinking. When a person moves from one area to another and knows that his voting card is different from the one he is used to, that person can adapt easily. However, this ability is not yet owned by computers, so a protocol is needed. Some examples of simple protocols include secret splitting protocols, secret sharing protocols, and so on.

a) Secret Splitting Protocol

If Anto has a secret, he can give 'half' the secret to Badu and the 'half' to Chandra. Badu, who received the first half of Anto's secret, could not find out what the secret contained. Likewise with Chandra. However, if Badu and Chandra combine the pieces of the secret, then Anto's secret will be revealed. Secret splitting can be done by:

1. Anto generates a random string R which is the same length as the secret message M.
2. Anto performs an XOR operation between M and R, resulting in S.
3. Anto gave R to Badu and S to Chandra
4. If Badu and Chandra meet, they can get M's secret message by performing an XOR operation between S and R.

b) Secret Sharing Protocol

Another cryptographic protocol is secret sharing, which allows the distribution of a secret among a group of people who trust each other. This secret sharing protocol applies the (m,n)-threshold scheme, ie information about the secret is distributed in such a way that any m of n people ($m < n$) have enough information to find the secret, but any set of $m-1$ person can't do it. In any secret sharing scheme, there is a select group of people whose cumulative information is enough to solve the secret.

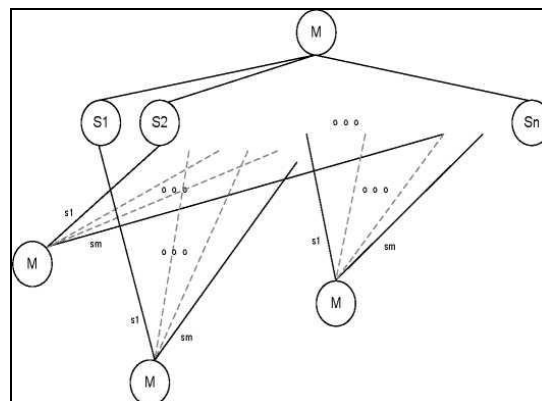


Figure 2. Protocol Secret Sharing

In some implementations of a secret sharing scheme, each participant receives a secret after the secret in question is generated. In other implementations, the real secret is never made visible to the participant, even if access is granted to get the secret (eg in access to the room or permission to process).

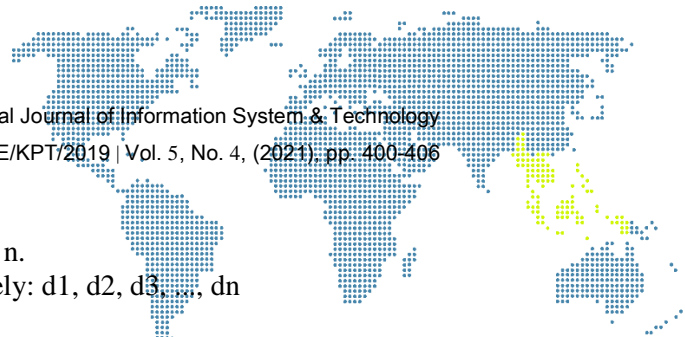
2.5. Asmuth-Bloom Algorithm

The Asmuth-Bloom algorithm uses modulo arithmetic, prime numbers and random numbers to increase its security. In addition, this algorithm also requires the help of the Chinese Remainder theorem when merging messages back [4].

Broadly speaking, the Asmuth-Bloom algorithm can be divided into 3 stages of the process, namely:

1. Key Formation Process

- a) Determine a prime number p , where p is greater than the ASCII Code value of



- Message M.
- b) Determine the values of m and n, where $m < n$.
 - c) Find n numbers that are smaller than p, namely: $d_1, d_2, d_3, \dots, d_n$ such that:
 - 1) A series of d values in ascending order, $i_n < d_{i+1}$.
 - 2) Each value in is prime relative to every value in the others.
 $d_1 * d_2 * \dots * d_m < p * d_{n-m+2} * d_{n-m+3} * \dots * d_n$.

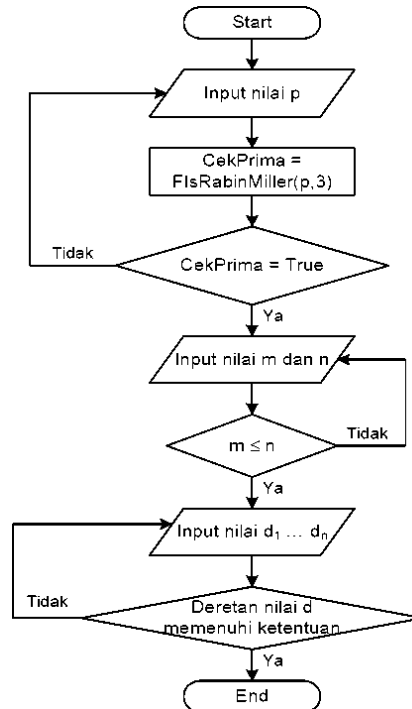


Figure 3. Asmuth-Bloom's Secret Sharing Algorithm Key Formation Process

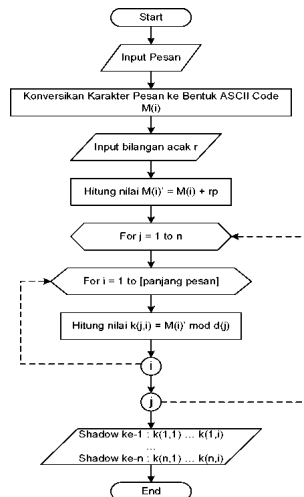


Figure 4. Shadow Formation Process Asmuth-Bloom Secret Sharing Algorithm

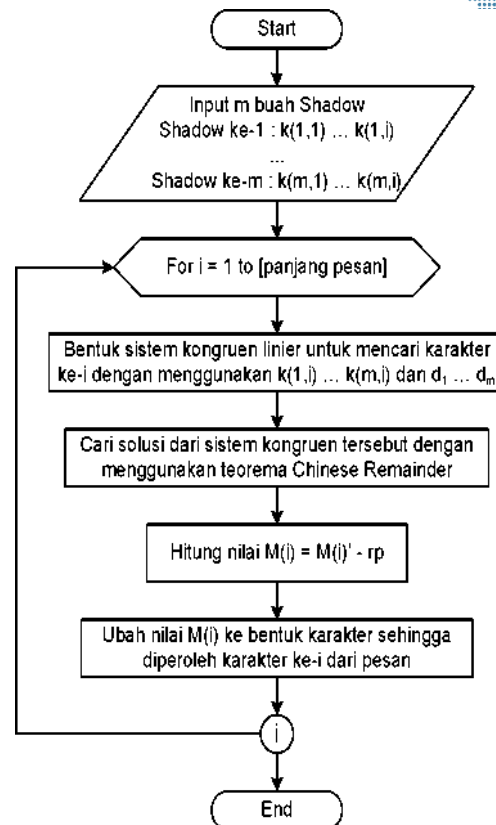


Figure 5. Asmuth-Bloom's Secret Sharing Algorithm Shadow Merger Process

3. Result and Discussion

Designed is able to explain the working process of this secret sharing algorithm in detail step by step and provides an interface for applying this algorithm that can be applied to text files.

1) Key Formation Process

As in the public key cryptography algorithm, the key generation process from this secret sharing algorithm generates a private key and a public key that will be used in the process of forming and merging shadows.

The private and public keys contained in the Asmuth-Bloom Secret Sharing algorithm can be detailed as follows:

a) The public key (public key) of all users, namely the prime number p .

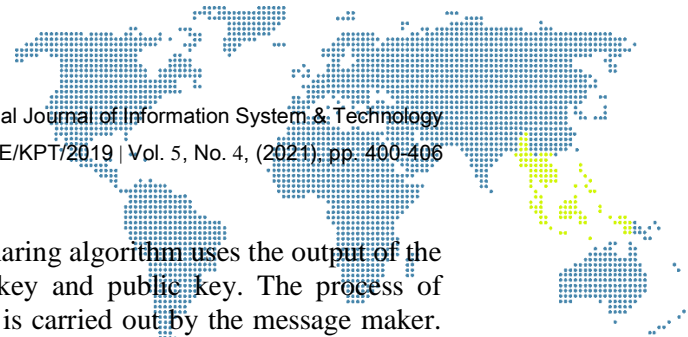
This prime p number can be generated by using the prime number generation algorithm from the Rabin-Miller method or manually inputted and tested using the prime number testing algorithm from the Rabin-Miller method. This prime p must be greater than the ASCII Code of the message. Since the largest ASCII Code value is 255, the prime p value must be greater than 255.

b) The private key of each user, which is a series of values $d_1 \dots d_n$.

This series of d values can be determined manually or generated randomly by fulfilling the following requirements:

1. A series of d values in ascending order, at $< at+1$.
2. Each value in is prime relative to every value in the others.
3. $d_1 * d_2 * \dots * d_m < p * d_{n-m+2} * d_{n-m+3} * \dots * d_n$.

In addition, the key formation process will also produce the values of m and n where the value of m is the number of shadows needed to form the message and the value of n is the number of shadows desired.

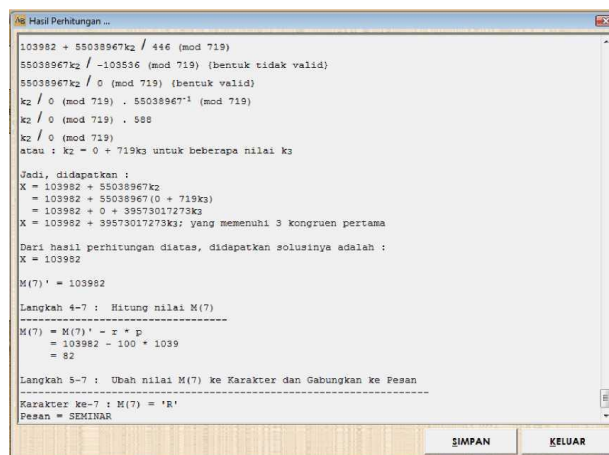


2) Shadow Forming Process

The shadow formation process from this secret sharing algorithm uses the output of the key formation process, namely the user's private key and public key. The process of forming the shadow of the secret sharing algorithm is carried out by the message maker. The result of this process is n shadows that will be distributed to n people, where each shadow has a different value.

3) Shadow Merging Process

The process of merging the shadow of this secret sharing algorithm uses the output of the key formation process, namely the user's private and public keys, as well as the user's private key m fruit shadow. The process of merging the shadow of the secret sharing algorithm is carried out by m people who want to get the original message. The result of this process is the original message which is hidden by the message creator. The shadow merging process uses the help of the Chinese Remainder theorem to find a solution of a linear congruent system formed from the combination of m shadows and m d_i values.



```

103982 + 55038967k2 / 446 (mod 719)
55038967k2 / -103536 (mod 719) (bentuk tidak valid)
55038967k2 / 0 (mod 719) (bentuk valid)
k2 / 0 (mod 719) . 55038967^-1 (mod 719)
k2 / 0 (mod 719) . 988
k2 / 0 (mod 719)
atau : k2 = 0 + 719k3 untuk beberapa nilai k3
Jadi, didapatkan :
X = 103982 + 55038967k2
  = 103982 + 55038967(0 + 719k3)
  = 103982 + 0 + 39573017273k3
X = 103982 + 39573017273k3; yang memenuhi 3 kongruen pertama
Dari hasil perhitungan diatas, didapatkan solusinya adalah :
X = 103982
M(7)' = 103982
Langkah 4-7 : Hitung nilai M(7)
-----
M(7) = M(7)' - r * p
      = 103982 - 100 * 1039
      = 82
Langkah 5-7 : Ubah nilai M(7) ke Karakter dan Gabungkan ke Pesan
-----
Karakter ke-7 : M(7) = 'R'
Pesan = SEMINAR
    
```

Figure 6. Form Results of Proses.

4. Conclusion

After conducting several experiments on the encryption and decryption process of a text file using the Secret Sharing protocol and the Asmuth-Bloom algorithm, the following conclusions can be drawn:

- The program developed in the research goes according to plan
- Problems that arise in research can be answered using research results and data analysis
- The results of the encryption of the processed sentence strings form several shadows that look just like a series of numbers that have no meaning.
- The result of the decryption of the cipher file shows that the cipher file can be substituted into its original form.

References

- [1] G. Blakley. Safeguarding cryptographic keys. In Proc. of AFIPS National Computer Conference, 1979.
- [2] Munir, R. 2015 "Teori Bilangan". Slide tersedia di: [https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2015-2016/Teori %20Bilangan%20\(2015\).pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2015-2016/Teori%20Bilangan%20(2015).pdf)
- [3] Asmuth, Charles and John Bloom. "A modular approach to key safeguarding." IEEE Trans. Inf. Theory 29 (1983): 208-210.
- [4] Kaya, Kamer, and Ali Aydın Selçuk. "Threshold cryptography based on Asmuth-Bloom secret sharing." Information sciences 177, no. 19 (2007): 4148-4160.