Smile Mask to Capsulation MOLAZ Method

Moceheb Lazam Shuwandy

TU College of Computer Sciences,
Tikrit University Iraq, 0043 TU, Tikrit, Iraq

Abstract

Concealment of information is the most important things of interest to scientists and users alike. The work of many researchers to find new ways and methods for building specialized systems to protect the information from hackers. The method of those techniques AES and an adopted by the U.S. Department of Defense and launched in the eighties to the world. Even so, it parallels the evolution of these methods to penetrate systems. Researchers were developed this method for the protection of this algorithm. In the end of 2010, the researcher Engineer Moceheb Lazam during his studies at the Masters in the Universiti Utara Malaysia, develop this algorithm in order to keep the encryption and decoding. It was called MOLAZ. It used two algorithms AES 128 and AES 256 bits, and switching between them using special key (Ks). In addition, it uses two keys to encryption and decryption. However, this method needs to be develops and supports the protection of information. Therefore, in 2011 appeared MOLAZ-SM. It presents a study is the development of this system by adding the mask technique to prevent the use of the style of repeated attempts to enter the key. The system depends on the base "If you enter a true key, you obtain to the truth information, but if you enter the false key; you obtains to the false information."

Key words:
Switching, AES-128, AES-256, MOLAZ, MOLAZ-SM.

1. Introduction

MOLAZ is the method published in International Journal of Computer Science and Network Security (IJCSNS) in September 2010 entitled (Switching between the AES-128 and AES-256 Using Ks * & Two Keys). It is referred to the author's name, Moceheb Lazam. It uses three keys (two private keys and one key) and two algorithms AES 128 bits and AES 256 bits. The system divides the plain text into four parts. Each part uses one of the algorithms and the choice operation selected according to a key. The main part of the system is called the System Determine Algorithm (SDA). It deals with the text before and after the encryption and decryption process. It fulfilled where all operations such as dividing the text and then send it to the appropriate algorithm for it, see below Figure 1.1 .

The main purpose of using this technique is to increase the number of times using the 128 to be three in the key generation (Ks). In addition, it aims to increase the speed versus complexity in the data processing. This is the reason for naming the system "Smile Mask (MOLAZ_SM)", because it does not give error messages if used wrongly and cannot open messages in case of not using the program. The system gives intruders simile saying "If you want us. You will find us if you are authorized but if you are not, we give you, but not what you want".

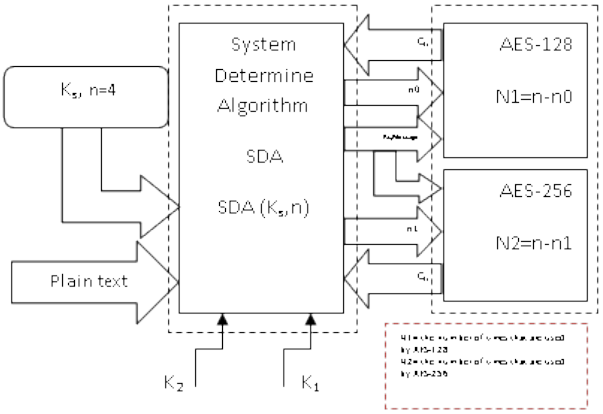

Figure 1.1: the SDA using Ks and the plain text.

The issue is that if the information is exposed especially if the documents in the administration have multiple attempts of attack; it will lead to an increased complexity in the mechanism of protection of data and this will slow processing. In order to encrypt the information, the user must provide two private keys (K1, K2) of the real level to the recipient as well as the message to be encrypted. This method can save the time especially in saving the hardware resource in implementing the AES 256 bit module and AES 128 bit module. Most of the designed modules can be used for both AES encryption and decryption. The use of AES 128 bit method of encryption saves cost and time which leads to a fast encryption. However, this method can be attacked and decoded more than other, like AES-192 and AES-256. The AES 256 bit method is strongest and difficult in attack when decoded, but in 2010, Seagate Corporation and Moceheb in same year had said, it is complex and slow during the process of encryption and decryption. Usually using keys are short and easy to remember, which leads to use of method of analysis and repetition to recover those keys.

This study aims to achieve the following research objectives:

Manuscript received December 5, 2013
Manuscript revised December 20, 2013

• To improve a system security by increasing complexity using AES-256 Method.
• To increase the speed and reduce the complexity by using 128 bits to give more speed to the system.
• To develop exchange (switching) between the cases due to the use of center processing (SDA), Smile Mask System (SMS) to check if the keys are correct or not and Fake part. (Figure 1.2) shows the main parts of the system.

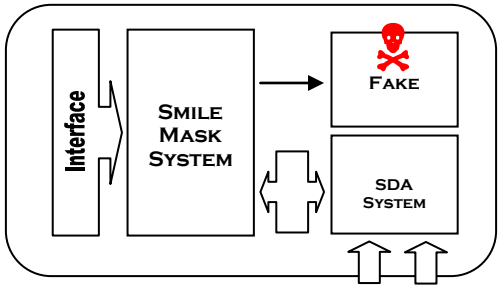

Figure 1.1: The main component of MOLAZ-SM.

2. MOLAZ-SM System

This system is contains five main parts like shown in Figure 1.2 above, SMS, fake operation, key generation, SDA and AES algorithms (AES 128 bits and AES 256 bits). Each part of this system connects to each other like the SMS when check the keys, it must send a request to SDA to get LS as shown below in Figure 2.1.

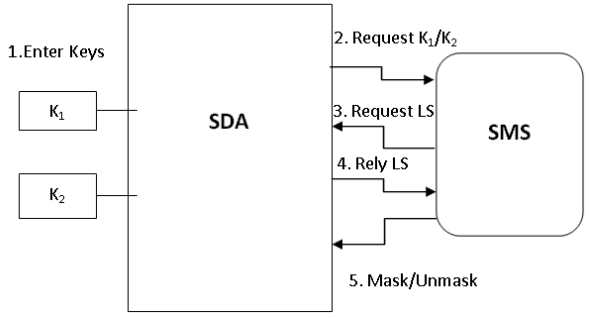

Figure 2.1: Relation between SMS and SDA by get LS.

Besides that, the security system in SMS follows what the SDA sent before. However, SMS is that of giving the order to SDA to carry out the mask or not, because it has a decision about that. In the next section, each part of MOLAZ-SM and the relations between those parts will be discussed and explained.

3. Smile Mask System (SMS)

The main tasks for SMS during the encryption process are two operations. The first is the process of making sure the keys K1 and K2 are accepted or not; the SDA sends a request to it with the keys and the ciphertext. However, the SMS does not work mask only after it answers SDA and only after the request is send from SDA for the second process (the mask).
The second operation of SMS is waiting for any request from SDA to make the Mask operation. At this time, SMS links with another system called fake system. It does one operation after get the request from SMS and SDA, called fake operation. Process of reading the data stop whether the plaintext or ciphertext. This happens when the SDA does request the process of the mask during the process of decoding. It is worthy to mention that the mask was effective only in the decoding process because it is important to protect the encrypted text.

4. Fake Operation

Fake operation is a virtual process of the decoder without using the ciphertext because of the request SMS to SDA to do the mask, which it is part of this process. This process consists of three inputs and one output using a function within the SMS (SMILEMASK_OOPS). The three inputs sent from SDA during the application process are two keys (K1 and K2) and fake text.
On the other side, there is only one output which is the fake text. The selection process of many texts through the assistance of another key is called sub_F. The sub_F is a random key; the aim of it is shifting text entered for completing the process of fake. Add sub_F to the Fake system by using to obtain shift operation. Figure 4.1 shows the operation of the shift to the fake text that entered before. In addition to the length of the text, which is chosen depends on the size of the ciphertext (ct) and Luck System (LS).

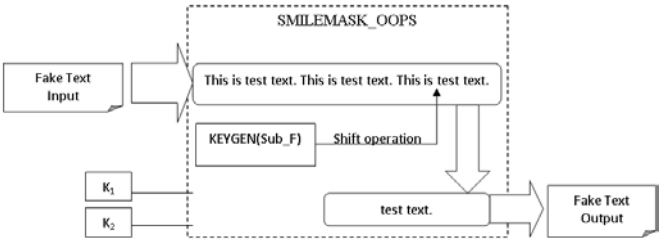

Figure 4.1: The Shift operation of the Fake text by using sub_F key.

5. Key Generation (K_s)

Ks is the key that content four numbers generate randomly to determine the sequence of algorithm that used in this system. Meaningful time period must using 128 bit keys since the increasing the number of bits of keys, increase period of time leads to wasting time and loss speed during ciphertext (Moceheb, 2010). The system uses the AES-128 encryption to obtain to ciphertext when the switch (Ks) has the even numbers 0,2,4,6 and 8, and AES-256 has the odd numbers 1,3,5,7 and 9. The difference between MOLAZ and MOLAZ-SM in Ks is the condition of generating the key that should be the number content just one odd number to select AES-256 algorithm.

Let Ks=4618, i=0, (see Table 5.1).

Table 5.1: Show the sequence of algorithms in MOLAZ-SM.

i	Ksi	The sequence of operations
0	4	AES-128 bit
1	6	AES-128 bit
2	1	AES-256 bit
3	8	AES-128 bit

If Ks =0000…or 9999, nothing to do if all numbers odd or even; since the benefit of this operation to using the two algorithms in the message.
Besides, the ASIC solution of AES is also required, because it can be more secure and consumes less power than that implemented by software.

Ks=kygengenerat(10000);
Except if Ks =0000 ,….or 9999
The flowchart in Figure 5.1 shows the code below:

```
if(checkallodd (ks,4)==true || checkalleven (ks,4)==true)
        ks=kygengenerat(10000);
```

In each function checkallodd and checkalleven applies the following condition:

```
If (odd>1 || even==4)
        Return (Ks generator);
 Else
        Return;
```

6. System Determinate Algorithm (SDA)

SDA in MOLAZ-SM is almost the same system in MOLAZ, but it differs in several things:

1 - Ensure the acceptance over the keys by sending a request to the SMS.
2 - Send a request to the SMS to carry out the mask.
3 - Providing LS and sent it to the SMS.

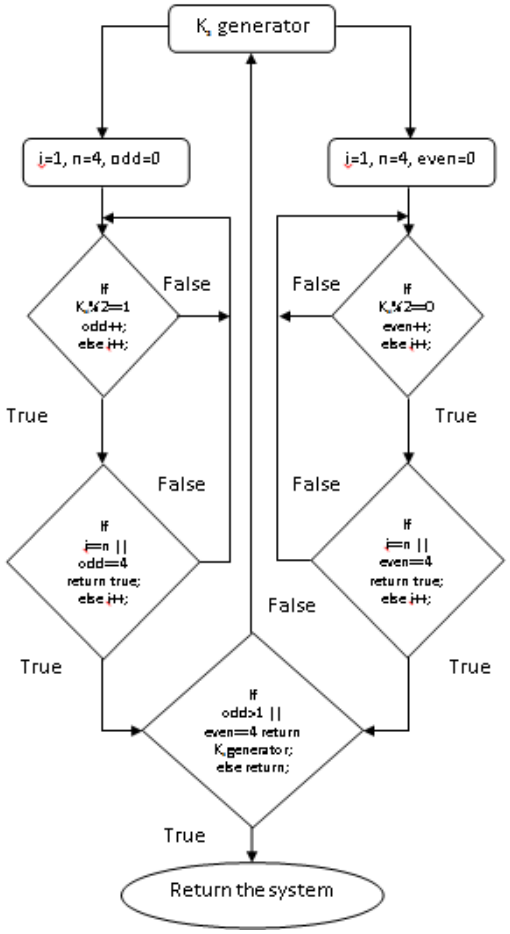

Figure 5.1: Flowchart show checks Ks elements.

7. AES Algorithms

AES-128 and AES-256 are used in MOLAZ-SM system like MOLAZ, but should check the two keys before decrypt the ciphertext. Besides that, all operations in encryption and decryption is similar to MOLAZ.

8. Input/output Data Operations

The system is consists of three inputs and one output. The three inputs are two keys (K1 and K2) and plain text. The plain text has two ways to enter the data when using the system. The first one is entering the data immediate by using the keyboard or copy-paste in the text field. The second way is reading-writing the data from or to file; even access to the file to get the data should notice the type of the file is (.txt).
Besides that, the output gets the cipher text or plain text, depends upon choice in the input (file or the keyboard). The result views in the text field of the file, especially the file deals with extension (.txt) and extension (.mlz)

encryption. On the other hands, the decryption operation should choose the (.mlz) file as input and the (.txt) file will become immediately the output.

9. Luck System (LS)

Luck System is the key that applies to safe the system to verify the keys entered, and cipher text is original or not. It is representing as a shield to the fighter in the battle field and like the fingerprint to verify the identity of the person. Therefore, the important strategy in this system keeps some information secret.

The LS sends to SMS when SDA request to SMS to verify that the keys are original or not. When the SMS received the request from the SDA, the system does nothing without LS, so SMS should send the request to SDA in order to get LS. As soon as, SMS gets the LS make the process on the keys and the cipher text like shown in the Figure 2.1 above.

10. Brute –Force attack

Brute force attack is the most famous password cracker known method. This attack tries to use every possible combination of characters a password. In order to restores the password of one character is enough to try 26 combinations ('a' to 'z'). This ensures that the system find password, but when two characters password requires 26 * 26 = 676 sets. The number of possible combinations (and thus, time required) is growing rapidly with the length of the password and more than this method quickly becomes useless. Are you ready to wait for two months while the cracked your password 9 characters? What about 100 years to get the password of 11 characters? Along the length of most of the character set, you must also specify the character set of any list of characters that will be included in groups. The character set requires a longer period of time. Usually, if no idea, what the letters contained in the password, here is the problem.

Password software calculator can be used to estimate the time required to brute-force attack. Table 3.4 below shows the time required to brute-force attack, depending on the length of the password and use the character set. It is supposed to be carrying out the attack speed on a single computer of brute force 500 000 password per second.

10.1 Brute force attack technique

According to Davidson B., Luiz C. and Raphael M. 2011, the system tries to enter password and then check the memory if the decryption gives error or data. If it replies the data the attempt stop, if not it generates another key.

Conclusion

In this paper, The study introduced an improved design for a random key generator (Ks) in cipher algorithm. A cipher generates successive elements of the key based on an internal state. There is no algorithm occupying an equivalent position in the field of ciphers. There are huge varieties of alternative cipher designs and cryptanalysis tends to be couched in very general terms. It introduced here a cipher as a new random key generator used to provide a key to be used for encryption. The system consists of three parts: the SDA part, the SMS part and Fake operation. The SDA part uses Ks to determine the sequence of four parts of the message in ciphering operation.

References

[1] Adams, C., Heys, H., Tavares, S. & Wiener M. (1999). An analysis of the CAST-256 cipher. IEEE .22(1)

[2] Beaver, K. (2006). Hacking For Dummies. Indianapolis, Indiana: Wiley.

[3] Biryukov A., & Khovratovich D. (2009). Related-key Cryptanalysis of the Full AES-192 and AES-256. Advances in Cryptology–ASIACRYPT , 1-18.

[4] Bruce S., John K., Doug W., David W., Chris H., Niels F., Tadayoshi K.& Mike S., (2000). "The Twofish Team's Final Comments on AES Selection". Retrieved (12/10/2011) from (http://www.schneier.com/paper-twofish-final.pdf). http://www.schneier.com/paper-twofishfinal.pdf .

[5] Davidson B., Luiz C. and Raphael M. (2011). Brute force attacks against reflection-based software integrity verification methods. Universidade Federal do Rio de Janeiro (UFRJ), 1-8, Brazil.

[6] EMC (2011), New Attack on AES. RSA Share Project, EMC Corporation , Retrieved(15/9/2011) from https://community.emc.com/community/edn/rsashare/blog/2011/08/31/new-attack-on-aes.

[7] Federal Information Processing Standards Publication 197. (2001). Advanced Encryption Standard (AES). Nov. 26.

[8] James N., Elaine B., Lawrence B., William B., Morris D., James F.,& Edward R. (2000). Report on the development of the Advanced Encryption Standard (AES). Journal of Research-National Institute of Standards and Technology , 3(106),511-576.

[9] Jason W. (2004). Java Cryptography Extensions: Practical Guide for Programmers (The Practical Guides). Morgan Kaufmann .

[10] Jim H., (2011). Explanation of AES, Retrieved (5/11/2011) from http://www.giac.org/cissp-papers/67.pdf .

[11] Jonathan B.,& Knud S. (1998). Java Cryptography. Oreilly,First Edition.

[12] Joost, K. (2011). Practical hacking AES using the S-box weakness. IN DEI NOMINE FELICITER. Russia. Retrieved (6/10/2011) from http://www.cs.ru.nl/bachelorscripties/2011/Joost_Kremers___0714402___Practical_hacking_AES_using_the_S-box_weakness.pdf

[13] Khadivi, P. ,& Momtazpour, M. (2009). Application of data mining in cryptanalysis. Communications and Information Technology, 9th International Symposium on , 28-30.

[14] Moceheb L., Ali K., Firas L., & Adib M. (2010). Switching between the AES-128 and AES-256 Using Ks * & Two Keys

(MOLAZ Method). IJCSNS International Journal of Computer Science and Network Security , 8(11).

[15] Srinivasan, S. (2006). Security and Privacy in the Computer Forensics Context. Communication Technology, 2006. ICCT '06. *International Conference* , 1 - 3 .

[16] William S. (2010). *Cryptography and network security-principles and Practices*, Prentice Hall of India, 3 rd Edition.

Moceheb Lazam Shuwandy received the B.Sc. degrees in Computer and Software Engineering from Al-Mustansiria University Baghdad – College of Engineering - Department of Computer Engineering and Software 1998-2003 and M.Sc. degrees in Information Technology of Utara University of Malaysia – College of Arts and Sciences in 2013. During 2005-2008, he worked as supervisor of the Internet networks in the Tikrit University, Iraq (TUI). In 2005 he served as director of the website of the University of Tikrit, Iraq. In 2006-2009 he worked as a lecturer in the Faculty of Education / Department of Mathematics - University of Tikrit, Iraq in article Visual Studio® J # & C#. He is a member of the Iraqi Engineers Syndicate /Baghdad 2004. He works lecturer in College of Computer Science, TUI.