Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 00 (2014) 000–000

Procedia
Computer Science

www.elsevier.com/locate/procedia

International Conference on Graph Theory and Information Security

Information Concealment Through Noise Addition

Budi Rahardjo[a,*], Kuspriyanto[a], Intan Muchtadi-Alamsyah[b], Marisa W. Paryasto[c]

[a]*School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Jalan Ganesha no.10, Bandung 40132, Indonesia*
[b]*Faculty of Mathematics and Natural Sciences, Institut Teknologi Bandung, Jalan Ganesha no. 10 Banding 40132, Indonesia*
[c]*School of Economics and Business, Telkom University, Jalan Telekomunikasi Terusan Buah Batu, Bandung 40237, Indonesia*

Abstract

A novel method of concealing information is proposed. Encrypted data is mixed with noise to add the security. The cost of an attack increases linearly with the size of the added noise. Since generating noise is cheap, the size of noise can be as large as possible. However, there are issues in automatically detecting and separating noise in the intended recipient end. This stegocrypto method also provides protection from traffic analysis attack.

Keywords: security; cryptography; steganography; information; noise

1. Introduction

Protecting information from unauthorized access can be done through various mechanisms, such as applying cryptography. In cryptography, information is encrypted into intelligible data using an algorithm and a key. The strength of the protection depends on the algorithm and the length of the key being used[1]. Usually, the stronger the algorithm, the more computation and time are needed. Balancing the cost of protecting the information and the value of the information is also of our concern. Note that encrypting data is still susceptible to traffic analysis attack.

Another approach to conceal information is through steganography. In steganography[5], the information is concealed or hidden in a cover media. Usually, the media is an image, but it is not limitted to that. Concealing information can also hide communication traffic patterns to avoid traffic analysis attack.

In this paper, we propose a method of combining steganography and cryptography by mixing noise with encrypted data to increase the security.

2. Stegocrypto

An attacker will start the process of an attack after he sees an encrypted data. He would not attack data that he knows that the data is not part of the message he is trying to gain. For example, he would not try to attack noise. The

[*] Corresponding author
E-mail address: rahard@gmail.com

idea we propose is to mix encypted data with noise so that the attacker does not know that there is a message. To explain this, we borrow the idea from steganography. Thus, the stegocrypto idea.

In a conventional steganography, the stream of plain message is embedded into a steam of cover message. The cover message could be text, image, or in this case is noise. An attacker must sift through the stream of noise to find the message. If the message mixed with the noise is in encrypted format, the attacker has an additional difficulty. He has no option but to attack both the encrypted message and noise. Thus adding another level of security.

Picture an empty and clear channel. In this channel we send a message (information) in clear text. An attacker can see the message easily. To protect this message from unintended recipients, we can encrypt the message and send the encrypted message through the same channel.

As an example, we can encrypt ”Hello, World!” using 256-bit AES using ”secret” (without quotes) as the password. The result is the following string.

```
U2FsdGVkX19mSFJ5nvUvYk+FSWJCjIDuBaAfunocZLc=
```

Watching the channel, an attacker can see that there is a message. While the message is in encrypted form, an attacker can capture it. The attacker cannot gain the original information without knowing the password. However, the attacker can perform various attacks (such as brute force) in an attempt to gain the original information or the password used to encrypt the information. The cost of such attack depends on the length of password and the strength of algorithm being used.

We can increase the difficulties by mixing the encrypted information with noise. An example of such a mix is shown below.

```
6RBYXFD7C8TTYEQ3TEQL34FB6NGV1IM6JIVK3BTQ
U0F9V4UA8WR48X76C640HQEJIOZ80WTAG8RKTL9Y
H9HEX4EKQZ5E4T9FZ4V8SMVAS7RYLN7PI2NRM2RZ
HPFYGKKZPJ1ANEJT4LM8IW4R0R3W88CWW1Q3DZPV
BA7SU2FsdGVkX19mSFJ5nvUvYk+FSWJCjIDuBaAf
unocZLc=EYA5FVCM694MU7BXJVUPFHXGI411HKJV
L2YMNYP4KH9BOFMBEBMC4LRO2DUN85WBASXJGV4C
EQMPGXB41TMWTPYN3KCV9X1DPFR8TRUTX8XDI1D2
75DW
```

An attacker does not know where to start the attack. The attacker can try to (brute force) attack the mixed message, ie. trying to decrypt noise.

3. Security Metric

As we can see from previous section, the attack is more difficult. Can we measure the increase in security?

Let C be the cost of an attack (decrypting the message). Without the noise, that is the only cost of an attack. Let k be size of noise (say, in number of characters) compared to the size of encrypted message. An attacker must also attack the noise data. Thus, the cost of the attack becomes $C' = C + kC$.

Another attack scenario involves detecting and separating noise from encrypted data. Assuming that an attacker knows how to do this, and it costs less than trying to decrypt noise, this method is preferable. Let the cost of detecting is D. Thus, the cost of the attack becomes $C' = C + kD$.

While the cost of attack only increase linearly, generating noise and adding it with encrypted data is easy. We can make the k as big as possible.

Since the attacker does not know where to start, and perhaps create some kind of sliding window, then this adds to the cost of the attack. The attacker also must know when to stop when attacking noise.

4. Detection Issues

How can an intended recipient separate noise from (encrypted) data but an attacker cannot? There various ways to do this automatically. First, we can create disjoint sets of noise data and encrypted data. For example, in an Elliptic

Curve Cryptography (ECC)[2,3,4] setting we can set the encrypted data to be on curve points and noise to be off curve points. Another idea would be to create two disjoint or orthogonal codes and use error correcting codes to distinguish the data and noise. However, care must be taken so that an attacker cannot perform this easily.

Secondly, we can use markers to indicate the begining of (encrypted) data.

```
noise ... | marker | data | noise ...
```

Finding these markers, an intended recipient can thus extract the encrypted data from noise. The data can be in fixed length or there is an indicator of length of data. Another method is using header and trailer.

```
noise | header | data | trailer | noise
```

However, a new problem arises in this scheme. The noise generator may accidentaly generate unintended marker. For example, if we use "FF" (one byte data) as the marker then there is a probability of $1/256$ than the noise generator generates that particular number. We can increase the length of the marker to reduce the probability. For example, if we use two bytes data as the marker then the probability of accidentaly generating unintended marker becomes $1/256^2$. We can continue increasing the length of the marker. Unfortunately, increasing the length of the marker makes it easier for an attacker to identify it since it may be repeated in the stream of mixed data.

5. Conclusion

Concealing encrypted data through noise addition increase the cost of attack. Although the additional security is linear in fashion, generating noise is very easy. The cost of generating noise is low, thus the noise to data ratio can be very large. This method also provides protection for traffic analysis attack.

Automatic detection and separating (encrypted) data from noise in the recipient end is still an issue. Progress is being made in this area.

Acknowledgements

This research is supported by Hibah Riset dan Inovasi KK ITB 2013.

References

1. A. K. Lenstra and E. R. Verheul, Selecting cryptographic key sizes, PKC2000, page 446-465, January 2000.
2. V. S. Miller, Use of Elliptic Curve Cryptosystem in Cryptography, pages 417-426. CRYPTO '85. Springer-Verlag, 1986.
3. M.W. Paryasto, Composite Field Multiplier Unit Architecture combining MH-KOA for Elliptic Curve Cryptography. PhD Thesis, Institut Teknologi Bandung, 2012.
4. B. Rahardjo, M. Paryasto, Kuspriyanto, I. Muchtadi-Alamsyah, F. Yuliawan, Nopendri, Pengantar Kurva Eliptik dan Lapangan Hingga dan Aplikasinya untuk Kriptografi, Penerbit ITB 2015.
5. W. Stallings, Cryptography and Network Security: Principles and Practice. Prentice Hall, 2010.