Rule Protecting Scheme for Snort

Hyeong-Seo Son*, Sung-Woon Lee, Hyun-Sung Kim*****
*School of Computer Engineering, Kyungil Univ., hyeongseo@gmail.com
**Dept. of Information Security, Tongmyung Univ. of Info Tech, staroun@tit.ac.kr
***School of Computer Engineering, Kyungil Univ., kim@kiu.ac.kr

Abstract - This paper addresses the problem of protecting security policies in security mechanisms, such as the detection policy of an Intrusion Detection System. Unauthorized disclosure of such information might reveal the fundamental principles and methods for the protection of the whole network. In order to avoid this risk, we suggest two schemes for protecting security policies in Snort using the symmetric cryptosystem, Triple-DES.

Keywords: IDS, rule-based system, information security

1 Introduction

The protection of computers and information systems is vital for the success of electronic commerce over Internet. Traditionally, access-control services such as firewall, are used to control access to systems and services. However, a flaw in an access-control component could lead to loss of information or computer resources by allowing an attacker to circumvent existing security measures. Intrusion detection system (IDS) is a technology that attempts to detect unauthorized activities and suspicious events that violate the effective security policy for a certain domain. IDSs provide a second line of defense, allowing intrusions to be detected in the event of a breach in the perimeter defense. Additionally, IDSs allow misuse or suspicious behavior of users to be detected [1-3].

Snort is an open source network intrusion detection system (NIDS), capable of performing real-time traffic analysis and packet logging on IP networks [3-6]. However, most of the NIDS does not consider the security of the NIDS itself, especially their rules. This is very important because if an attacker succeeds in mounting an attack against the NIDS, it is no longer useful to detect attacks by passing attacks in rules. As well, Snort does not consider the security of the Rule itself.

To solve this problem, Kvarnstrom et al. suggested a protection scheme that uses the one-way function [3]. In their scheme, a rule is composed of the sequence of concatenated hashed results. Their scheme could support confidentiality and integrity for rules only for the header information not for the options information of the rules. However, the options information is also very important for rule-based NIDS.

Thereby, the purpose of this paper is to give two methods to protect rules in NIDS, especially for Snort. We use a symmetric cryptosystem instead of using the one-way function to protect rules. So, our schemes could support both the confidentiality and the integrity for rules in Snorts.

2 Background

This section describes rules in Snort, which is an open source NIDS, and a symmetric cryptosystem, Triple-DES, for the method of protecting rules.

2.1 Snort

Snort is an open source network intrusion detection system, capable of performing real-time traffic analysis and packet logging on IP networks. It can perform protocol analysis, content searching/matching and can be used to detect a variety of attacks and probes, such as buffer overflows, stealth port scans, CGI attacks, SMB probes, OS fingerprinting attempts, and much more [2].

Snort uses a simple, lightweight rules description language that is flexible and quite powerful. Snort rules are divided into two logical sections, the rule header and the rule options. Figure 1 illustrates a Snort rule [7].

```
Alert tcp any any -> 192.168.1.0/24 111 (content:"|00 01
86 a5|"; \
    Msg: "mounted access";)
```

Figure 1. Sample Snort Rule

The text up to the first parenthesis is the rule header and the section enclosed in parenthesis is the rule options. The words before the colons in the rule options section are called option keywords [7].

- **Rule Header :** Tue Rule header contains the information that defines the "who, where, and what" of a packet, as

well as what to do in the event that a packet with all the attributes indicated in the rule should show up.

- **Rule Option :** Rule options form the heart of Snort's intrusion detection engine, combining ease of use with power and flexibility. All Snort rule options are separated from each other using the semicolon ";" character. Rule option keywords are separated from their arguments with a colon ":" character.

2.2 Triple-DES

Given the potential vulnerability of DES to a brute-force attack, there has been considerable interest in finding an alternative. We look at the widely accepted Triple-DES approach. Tuchman proposed a triple encryption method that uses only two keys [8]. The function follows an encrypt-decrypt-encrypt (EDE) sequence.

$$C = E_{k1} [D_{k2} [E_{k1} [P]]]$$

There is no cryptographic significance to the use of decryption for the second stage. Its only advantage is that it allows users of Triple-DES to decrypt data encrypted by users of the older single DES

$$C = E_{k1} [D_{k2} [E_{k1} [P]]] = E_{k1} [P]$$

Triple-DES with two keys is a relatively popular alternative to DES and has been adopted for use in the key management standards ANS X9.17 and ISO 8732. Currently, there are no practical cryptanalytic attacks on Triple-DES. Coppersmith[COPP94] notes that the cost of a brute-force key search on Triple-DES is on the order of $2^{112} \approx (5 * 10^{33})$ and estimates that the cost of differential cryptanalysis suffers an exponential growth, compared to single DES, exceeding 10^{52}[8].

3 Rule protection scheme

This section suggests a rule protection scheme for Snort by applying a symmetric cryptosystem. Rules are stored with the form of plain text in the Snort, therefore rule disclosure is one of the serious problem for NIDS. This section gives a solution for that. This section describes cryptographic requirements for the rule protection and then proposes a rule protection mechanism.

3.1 Security requirements

Confidentiality and integrity should be supported to protect rules in NIDS[8,9]. They are defined as follows :

> **Confidentiality** is the protection of transmitted data from passive attacks. With respect to the content of a data transmission, several levels of protection can be identified. The other aspect of confidentiality is the protection of traffic flow from analysis. This requires that an attacker not be able to observe the source and destination, frequency, length, or other characteristics of the traffic on a communications facility.
>
> **Integrity** is the assurance that data received are exactly as sent by an authorized entity, i.e., contain no modification, insertion, deletion, or replay.

To protect NIDS properly, these requirements should be satisfied for rules in Snort. Following sub-sections propose two solutions for the requirements.

3.2 Rule protection scheme – 1

This sub-section proposes a rule protection scheme, which makes a protected rule with a consequence of concatenated encrypted data to each field, using Triple-DES for Snorts.

Rules are classified into three categories: a fixed value for the header information, a fixed value for the option information, and a variable. We use a flag to distinguish them. From now on, we will focus on the way to protect rules depends on the classifications.

3.2.1 Fixed value

Most of header information and option information are classified into fixed value. Fixed value is so easy to apply protection. It is just protected by applying encryption by using a symmetric cryptosystem. However, the detection process for these two cases, header information and option information, are different. For the header information, information from inputted packet is first encrypted then compared with the protected rule.

In the option case, the length of string in the option field is different with each rule. So, the option information is decrypted first then compared with plain text information from inputted packet.

3.2.2 Variable

To get a better performance, we apply the commitment scheme and the de-commitment scheme proposed by Kvarnstrom et al. in [3]. However, the algorithm is very different with their scheme.

Commitment is applied to make a protected rule, whereas de-commitment to detect attacks. Here is an example for it. In the commitment phase, a representative value is made then this value is used to detect attacks after applying de-commitment phase to inputted packets. Figure

2 shows the relationship between two phases. Thereby, we can get a better efficiency than just applying encryption to the rule.

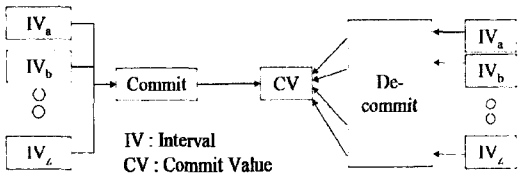

Figure 2. The relationship between commitment and decommitment

3.2.3 Rule creation

As described in the previous sub-sections, fixed values are encrypted first then concatenated to form a rule. However, commitment is first applied for variables, they are encrypted, and then concatenated to form a rule. These processes are shown in Figure 3.

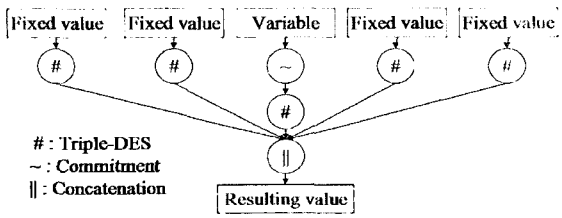

Figure 3. Step to make a protected rule.

3.2.4 Intrusion detection

Figure 4 shows overall processing steps for our proposed system. First of all, required information is filtered from network packets. The system decides whether applying encryption or applying decryption to each data. For the fixed value, the system takes encryption in the case of the header information but decryption in the case of the option information. However, filtered data should be de-committed first then encrypted for the case of variables.

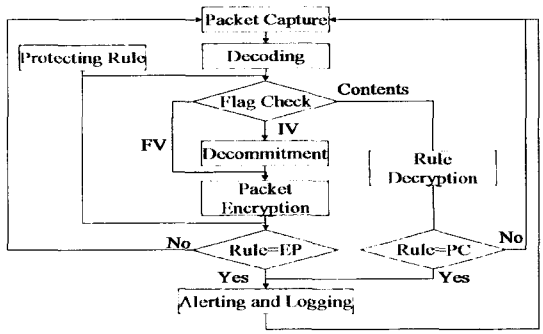

Figure 4. Intrusion detection steps.

3.3 Rule protection scheme – 2

This sub-section proposes an alternative from the previous scheme, which simply apply an encryption to a rule not apply to each field separately.

3.3.1 Rule creation

This sub-section proposes a rule protection scheme, which makes a protected rule with encrypted data to a rule, using Triple-DES for Snorts. Figure 5 shows the rule format.

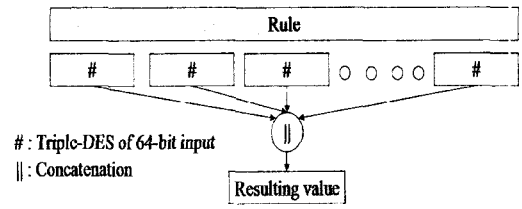

Figure 5. Steps to make a protected rule.

If the size of input is larger than Triple-DES input, repeated encryption is processed.

3.3.2 Intrusion detection

Figure 6 shows overall processing steps for our proposed system. The difference with the first case is that the encrypted rule is decrypted first then all the processing is same with the original Snort.

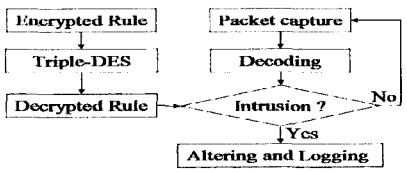

Figure 6. Intrusion detection steps.

4 Comparison and analysis

This section gives comparisons and analysis between our proposed schemes and previous scheme. There is only one research that provides the protection scheme for the NIDS, which is proposed by Kvarnstrom et al. in [3]. Table 1 shows the comparisons.

Table 1. Comparison of protecting schemes.

Property \ Scheme	Kvarnstrom et al[3]	3.2	3.3
Header Protection	Yes	Yes	Yes
Option Protection	No	Yes	Yes
Performance	High	Middle	Low

Proposed two schemes could be used to protect rules in Snorts. These two schemes have trade off between the processing structure and the performance. The first solution has processing efficiency than the second one. However, it requires source modifications in Snort. Contrast with that, the second solution has a simple structure, which does not require the source modification. But, it has somewhat performance degradation depends on the size of plain text rules.

5 Conclusion

This paper proposed two schemes for rule protection in Snort using symmetric cryptosystem. The problem of protecting security policies in security mechanisms, such as the detection policy of an Intrusion Detection System is considered in this paper.

Unauthorized disclosure of such information might reveal the fundamental principles and methods for the protection of the whole network. In order to avoid this risk, we first suggested a scheme, which makes a protected rule with a consequence of concatenated encrypted data to each field. Additionally, to simplify the processing, an additional scheme was proposed, which simply apply an encryption to a rule not apply to each field separately. These two schemes have trade off between the processing structure and the performance as described in the previous section. By using our schemes, we could efficiently protect NIDS.

References

[1] P.E. Proctor, "Practical intrusion Detection Handbook," Prentice Hall Inc, 2001.

[2] B. B. Coauthor, and A. Friend, "Boris's favorite journal paper," *A really great journal*, Vol 37, No. 3, pp. 433-448, Feb. 2000.

[3] S.Northcutt, J.Novak, and D.McLachlan, "Network Intrusion Detection - An Analyst's Handbook Second Edition," New Riders Publishing, 2001.

[4] H.Kvarnstrom, H.Hedbom, and E,Jonsson, "Protecting Security Policies in Ubiquitous Environments Using One-Way Functions," Pervacive'03, April 2003.

[5] Snort, http://www.snort.org

[6] M.Roesch, "Snort – lightweight intrusion detection for network," 13th Administration Conference, LISA '99, Nov. 1999

[7] S.Kim and Y.Kim, "A fast multiple string pattern matching algorithm," In proceedings of 17th AoM/IAoM Conference on Computer Science, August 1999.

[8] M.Roesch, Snort Users Manual Snort Release: 1.9.X, 26th, April 2002.

[9] W.Stallings, Cryptography and Network Security, Prentice Hall Inc., 2003.

[10] B.Schneier, Applied Cryptography, John Wiley & Sons, Inc., 1996