Rule-based Defense Mechanism against Distributed Denial-of-Service Attacks

Sung-ju Kim, Byung-chul Kim, Jae-yong Lee, Chan-kyou Hwang and Jae-jin Lee

Abstract — **Since the number of damage cases resulting from distributed denial-of-service (DDoS) attacks have recently been increasing, the need for agile detection and appropriate response mechanisms against DDoS attacks has also been increasing. The latest DDoS attack has the property of swift propagation speed and various attack patterns. There is therefore a need to create a lighter mechanism to detect and respond to such new and transformed types of attacks with greater speed.**

This paper proposes a rule-based defense mechanism against DDoS attacks. It is expected to improve the availability, confidentiality and integrity of service by blocking the propagation of DDoS attacks earlier.

Index Terms—**DDoS, Rule-based Traffic Control**

I. Introduction

Distributed denial-of-service (DDoS) is a very powerful attack, capable of depleting not only a system's resource but also network resources. Compared to the traditional DoS attack, more powerful effects on a target system can be felt when DDoS attacks are performed at the same time from distributed agents. Connection failures, network service speed deterioration and so on caused by abnormal traffic, such as worm viruses and DDoS attacks, are actually increasing in the real world. Also, as mentioned before, the damage has become more and more serious. In addition, DDoS attacks exploiting malicious Bots were listed in McAfee's top 10 security threats for 2007. [1] It predicted that the use of bots, computer programs that perform automated tasks, will increase as tools favoured by hackers. In addition, the amount of traffic that is suspected to be DDoS attacks is increasing year by year. We need to cope with such abnormal traffic to guarantee customers network QoS in this background. Also, it is necessary to analyze suspicious traffic and block abnormal traffic. The whole process should be performed in real time in order to have effectiveness. Thus, we present a rule based traffic control method against DDoS attacks. Following this introduction, the paper is organized as follows. Section 2 gives an overview of DDoS attacks, and Section 3 provides an overview of related works. Section 4 proposes a rule based defense mechanism. Section 5 shows tests and results. Finally, section 6 concludes the paper.

II. DDoS Attack Overview

A. DDoS Attacks

Intrusion is defined as "any set of actions that attempt to compromise the integrity, confidentiality, or availability of a computer resource." This definition disregards the success or failure of those actions, so it also corresponds to attacks against a computer system. [2] A denial-of-service attack is characterized by an explicit attempt by attackers to prevent legitimate users of a service from using that service [3]. Among those threats DDoS is an aggressive act that threatens the availability of service system resources as well as network resources. Since DDoS attacks use generally normal protocol packets, it is difficult to completely protect systems from DDoS with just intrusion protection methods, such as user authentication or cryptograph. Moreover, one of the things that make DDoS more threatening is its use of an automatic attack tool. It takes less than 5 seconds for an attacker to install an attack tool and put the attack into practice once he finds a vulnerable system.

B. Taxonomy of DDoS Attacks

Some examples of broadly known DDoS attack patterns are SYN Flooding, Smurf, Fraggle, ICMP Flooding, UDP Flooding and so forth. SYN Flooding makes use of the vulnerability of 3 way handshaking. The server is waiting for a reply from hosts while allocating the shared resources after sending SYN ACK packets. Since the attacker doesn't send the last ACK packets and continues to request a new connection, the server wastes its resources pending the reply because it doesn't disconnect the allocated resources. Smurf modifies a sender's IP address into a target IP address it intends to attack and transmits an ICMP Echo request to broadcast addresses, attaching ICMP messages to an IP header. All the hosts that receive the ICMP Echo requests send back ICMP Echo replies to the target IP address.
Fraggle is an attack tool that exploits UDP packets instead of the ICMP Echo. ICMP Flooding and UDP Flooding directly send massive ICMP Echo and UDP packets to a target without using a broadcast.

Manuscript received March 31, 2008. This work was supported in part by KT Network Technology Laboratory.

Sung-ju Kim is with Internet Research Department, KT Network Technology Laboratory, Daejeon, KOR (corresponding author to provide phone: 82-42-870-8277; fax: 82-42-870-8589; e-mail: be3sowon@ kt.com).

Byung-chul Kim is with the Information Communications Engineering Department, Chungnam National University, Daejeon, KOR. (e-mail: byckim@cnu.ac.kr).

Jae-yong Lee is with the Information Communications Engineering Department, Chungnam National University, Daejeon, KOR. (e-mail: jyl@cnu.ac.kr).

Chan-kyou Hwang is with Internet Research Department, KT Network Technology Laboratory, Daejeon, KOR. (e-mail: ckhwang@kt.com)

Jae-jin Lee is with KT Network Technology Laboratory, Daejeon, KOR (e-mail: jaejin@kt.com)

There are also other DDoS attack tools, such as TFN (Tribe Flood Network), Trin00, TFN2K, Stacheldraht and the like. Most of these attack tools take advantage of SYN Flooding. The procedure for these attacks is to select hosts to be used as masters and agents and then to install DDoS attack tools. Agents are subordinate to a master and send massive abnormal packets at the same time by the master's direction. Depending on the master's control command, each agent uses an attack method such as SYN Flooding, ICMP Flooding or UDP Flooding. Thus, a destination host receives a lot of packets all at once and is exhausted by dealing with them. The table below compares representative attack tools.

Table1. The characteristics of attack tools

	Trin00	TFN2K	Stacheldraht
Attack Type	UDP Flood	UDP/SYN/ICMP Flood, Smurf	UDP/SYN/ICMP Flood, Smurf
Source IP	No spoofing		Auto spoofing
Source Port	No appointment	Auto selection (random /sequential)	Auto selection (random /sequential)
Target Port	No appointment	Appointment	Range appointment
etc		Data encryption between masters and agents	Data encryption among attackers, masters and agents

C. DDoS Attack Components

The topology of a DDoS attack is composed of 4 parts. The compromised systems are broken down into handlers and agents. The agents are where the disabling network traffic is generated. One or more handlers control these agents. The handlers maintain a list of all responding agents. The handlers signal the agents when to begin an attack and specify the method of attack. The attacker, or client, controls one or more handlers and each agent can respond to more than one handler. [4]

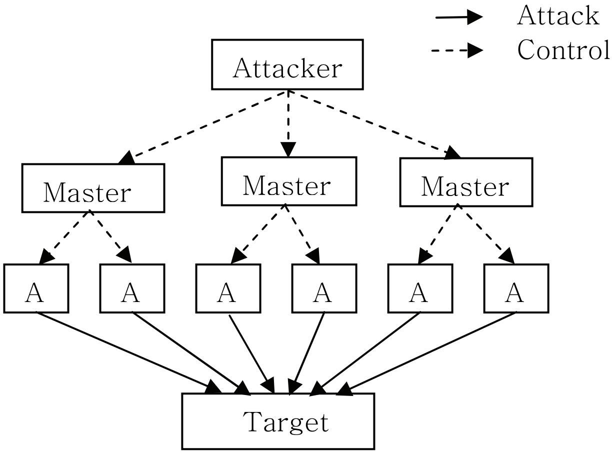


Fig.1. The topology of DDoS attack

D. The Patterns of DDoS Attacks

The patterns of DDos attacks can be grouped into three major categories: to completely deplete bandwidth in networks, to entirely waste the resources of a target system, to deny the service by attacking the vulnerability of applications or programming weaknesses. [5]

III. Related Works

A. Traditional Intrusion Detection and Response Methods

A detector's main goal is to detect and distinguish malicious packet traffic from legitimate packet traffic. Legitimate user activity can be easily confused with a flooding attack, and vice versa. [6]. Hence, employing statistical methods to detect DDoS attacks is the most efficient method. There are statistical detection algorithms, such as traffic volume evaluation, the entropy of packet property [7] and chi-square test [7].

a. Volume Evaluation

Among these statistical detection algorithms, volume evaluation is used to count the arrival time of packets to the Ethernet card

$$T = \sum_{i=1}^{n} (PAT[i-1] - PAT[i]) \qquad (1)$$

The above formula evaluates the arrival time of each packet and adds it to the arrival time of the packets as a group. That is to say, if a group is composed of 100 packets, the total arrival time of 100 packets to the Ethernet card is measured. The less traffic volume, the more packets arrive at the Ethernet card. This means that if traffic volume increases suddenly, we can conclude that the possibility of the occurrence of abnormal traffic is high.

b. Entropy Evaluation

Entropy evaluation calculates the randomness of some network property values and monitors the average variation value.

$$H = -\sum_{i=1}^{n} P_i \times \log_2 P_i \qquad (2)$$

The above formula shows how to calculate entropy H according to the property value of n. Pi indicates the probability that the property value of i_{th} is made a choice. A Chi-square test is a method to measure the variance of property values. Abnormal property values can be detected by calculating the variance of averages. The following is the concrete formula of the above method.

$$X^2 = \sum_{i=1}^{B} \frac{(N_i - n_i)^2}{n_i}, \quad n_i = \frac{n}{B}, \quad n = total\,samplesize \quad (3)$$

.

B is the binding value with which the values that the sample packets can have are packed. N_i is the number of packets included in the binding range among N sample packets and n_i is the expected value included for the binding in the general distribution.

c. Data Mining Approach

Data mining is defined as the process of discovering useful interrelationships, patterns, or trends from large amounts of data through statistical, mathematical or pattern-recognition technologies. [8] Data mining techniques have recently emerged as a means of identifying patterns and trends from large quantities of data. Among them, association rule mining is a popular summarization and pattern extraction algorithm to identify correlations between items in transactional databases. [9, 10, 11] Since the algorithms of association rules are expressed in a format for rules that analyze all the associations among various data, they can be applied to find significant data in the network domain. Also, there is often the need to study the frequent sequential patterns of audit data in order to understand the temporal and statistical nature of many attacks, as well as the normal behavior of users and programs. We use frequent episodes to represent the sequential audit record patterns. [12] But these data mining approaches have the problem that it is almost impossible to analyze massive amounts of data while online.

d. Rule Based Approach

Two association analyses performed by predefined rules are the CIDF (Common Intrusion Detection Framework) Correlator developed by Stanford University and the Planning Process Model. The Stanford CIDF Correlator [13] was developed with the intention of decreasing false intrusion alarms and detecting massive attacks like DDoS. It has a structure integrating CIDF with CEP (Complex Event Processing). Association rules detect intrusion information through whether the patterns of intrusion information accord with predefined scenarios. Although these rule-based approaches can guarantee superiority to other methods, it is hard to define all the attack scenarios.

B. The Problem of Traditional Analysis Methods

Since the above methods for analyzing information, such as pattern matching [14], traffic transition methods, etc are exposed too much, attacks having higher program complexity, for example spoofing, tend to increase to deceive those methods. Also, it's hard to cope with abnormal traffic in time, because it is too complex a process to judge abnormal traffic in real time. In order to overcome these problems, a rule based analysis system is suggested to detect abnormal traffic correctly and rapidly in real time. Furthermore, the analysis of payload can make the decision much more accurate. The latest trend is that malicious traffic attacks are becoming more complicated. It is necessary to consider various things when we judge abnormal traffic. Consequently, it is now essential that the system thoroughly analyzes various factors to judge abnormal traffic.

IV. The Proposed Mechanism

This paper focuses on the creation of a much more efficient and accurate algorithm compared to traditional ones by complementing the traditional methods. The proposed algorithm has the following characteristics:

1) Flexibility: Since DDoS attack methods tend to become more and more advanced and diverse, a function to adjust the critical value in accordance with the type of attack is required.
2) Agility: One of the properties of DDoS attacks is that they are very speedy. The system can be called a practical one if it detects and controls abnormal traffic in real time. The system should complete the entire process from detection to control within several minutes.
3) Conciseness: The ability to focus on detecting and blocking a swiftly spreading DDoS is necessary.
   A complete countermeasure for bandwidth-consuming attacks involves five stages – prevention, detection, first response, trace back and second response. [15]

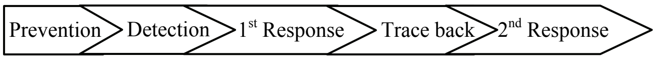


Fig.2. Present DDoS attack response steps

   It is not practical to try to detect and block DDoS attacks through a reverse trace or a second response. So, the function should concentrate on detecting abnormal traffic and blocking a problematic source IP so as to reduce the spread of damage with speed rather than having heavy functions.
4) Automation: If critical values to determine abnormal traffic are set and the process to block traffic over critical values can be automated, this procedure can save time in detecting and blocking abnormal traffic and decrease the spread of attack traffic.
5) Patterns: The judging validation can be increased to analyze the traffic patterns of attacks.

A. The Analysis of DDoS Attack Property

The definition of a denial of service attack is an attack on one computer or a network of computers that consumes resources only to decrease or even eliminate the availability of those systems. [16] DDoS attacks the victim with a lot of connection request packets from various sources.

Therefore, we can say that it is a DDoS attack if packets make for one or several destination IP addresses in a short time in spite of having diverse source IP addresses. Also, DDoS storms a specific port or random ports depending on the property of attack. However, through investigation, we found that some of the payloads of abnormal traffic have patterns. We also realized that there are various patterns, depending on which protocols and ports are used.

B. Traffic Monitoring and Control Framework

The figure below shows the entire sequence of detection for DDoS attacks. At first, traffic data are collected with the method of mirroring through tapping to reduce burdens to network devices. Traffic data for a fixed time are analyzed through the rule engine in the next step. As the mechanism makes a group of rules to detect attack traffic, it performs swift judgments on the basis of this group of rules in real time.

Afterward the mechanism feeds back through detail investigation and adjusts the group of rules so that it does not alarm falsely. Depending on the circumstances, the rules can be classified as follows:

1) Inferred Rule: This means a rule to extract new facts from already known facts judging from the condition of affairs.
2) Action-Enabling Rule: This means a rule to perform new acts judging from the condition of affairs.
3) Computational Rule: This means a rule to obtain results

through calculation judging from the condition of affairs.

If the result of the analysis is not over the critical values, the process can be judged as normal. Otherwise, it is regarded as abnormal traffic and the source IP is automatically blocked. The mechanism also performs a precise analysis of whether the judgments were correct or not after recording to a database. The critical value is reconfigured according to the result of this analysis if it is required to be adjusted. The typical factors referring to rules include TCP octets, TCP flows, TCP packets, UDP octets, UDP flows, UDP packets, TCP traffic source port variation, destination port variation and source IP address variation. The system sets the rules based on the services. Furthermore, we added payload patterns to make the judgements more accurate. The table below shows malicious pattern examples.

Table 2. Malicious payload patterns by services

Services	Pattern recognition
DDoS/DoS	Pattern matching Ex)content: “aaaaaa”, “abcdefg..”, !@#$%^&*(.., “12345..”, “00000”…
Spam mail recognition	Harmful words, spam characters
P2P	Malicious execution code pattern

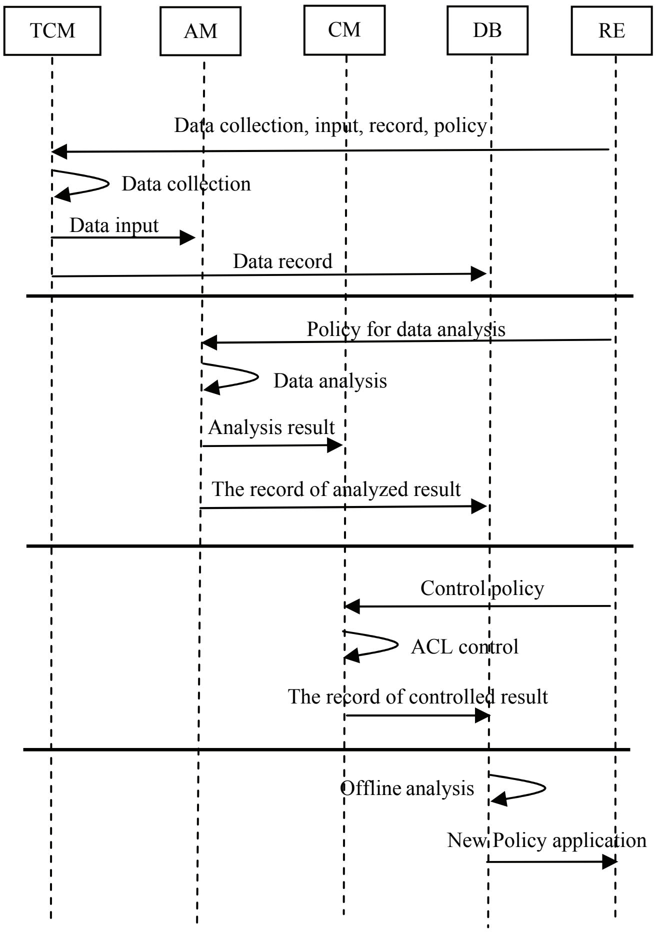


Fig.3. The traffic analysis and control sequence diagram

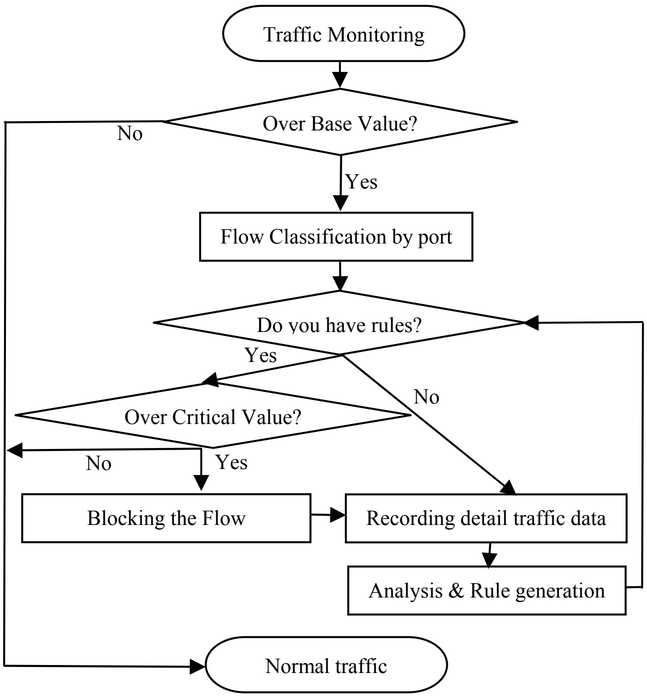


Fig.4. The traffic control flow chart

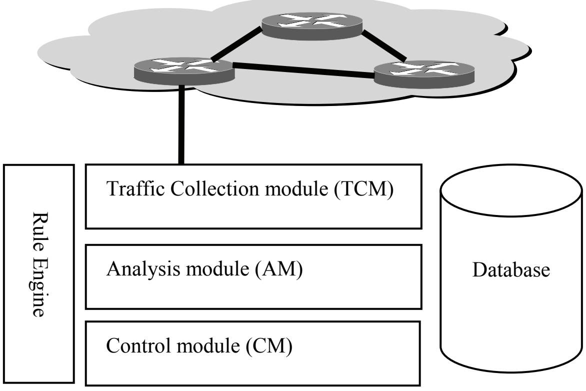


Fig.5. Traffic analysis and control functional block

V. Experiment and Result

A. Experiment Environment for Attack Tests

We collected network traffic data through NetFlow installed to manage real network. The raw traffic data came from a Cisco12416 router. This router had a 10Gbps interface and played the role of an access router that connected about 10,000 hosts and servers to the Internet. Well known DDoS attack tools, such as TFN2k, Stacheldraht and Synk4 were used for the test. The attack types we used were TCP SYN Flood, which takes on massive proportions, UDP Flood, ICMP Flood and a MIX attack which combining TCP SYN Flood with UDP Flood. And we varied the levels of spoofing attacks with cheating source IP addresses using the options of DDoS tools. Tests were performed to alter the spoofing options from complete random attacks to limited attacks that made the sub network addresses static and the user addresses change randomly. Since it is difficult to carry out serious attacks on real networks, we did this for only about 10 minutes using 3 to 5 agents to determine the performance of the mechanism. The reason we chose a real network was that it is was important to detect DDoS attacks from normal traffic to prove the performance of the system. So, it was assumed that real DDoS attacks are more powerful and destructive compared with the results of the tests.

B. *Performance Assessment*

We tested the following scenarios with well-known attack methods and tools to evaluate the performance of the mechanism.

Scenario #1: TCP SYN Attack

We carried out a TCP SYN attack using several agents. The objectives of this attack were not to increase the traffic volume and the load on the server. So, to detect an attack, the system should determine whether the PPS (Packets per second) increase and how many IP addresses are involved in attacking a victim as is shown below. The figure below shows the relationship between the attacking agents and an attacked target.

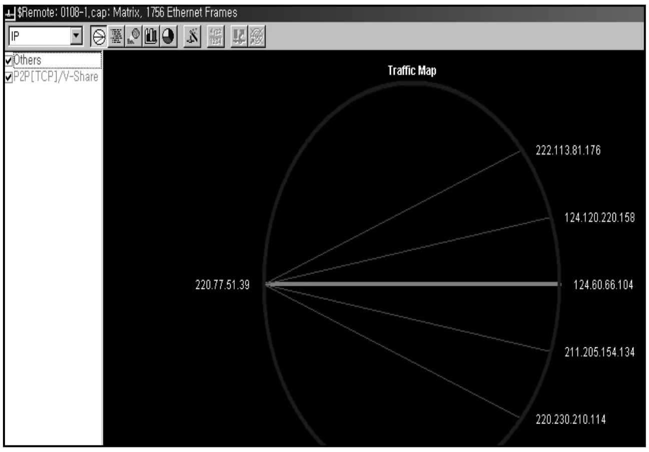
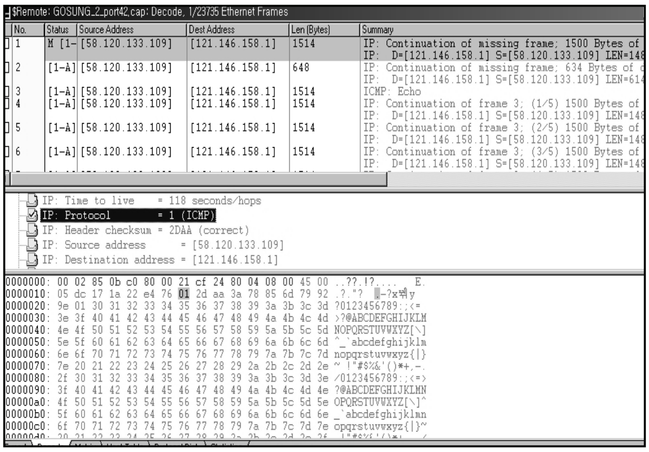


Fig.6. ICMP overflow traffic

Scenario #2: ICMP Attack

The second scenario was to perform DDoS to transfer massive packets over 3000Bytes to an attack target making use of ICMP Flooder to assess how early the mechanism detects a network attack. The figure below shows the features of the ICMP attacks, for which the PPS and BPS are over the critical values. Also, it makes sure that it is a real attack by the fact that the payload has regular patterns.



$Remote: GOSUNG_2_port42.cap: Decode, 1/23735 Ethernet Frames

No.	Status	Source Address	Dest Address	Len (Bytes)	Summary
1	M [1-	[58.120.133.109]	[121.146.158.1]	1514	IP: Continuation of missing frame; 1500 Bytes of IP: D=[121.146.158.1] S=[58.120.133.109] LEN=148
2	[1-A]	[58.120.133.109]	[121.146.158.1]	648	IP: Continuation of missing frame; 634 Bytes of d IP: D=[121.146.158.1] S=[58.120.133.109] LEN=614
3	[1-A]	[58.120.133.109]	[121.146.158.1]	1514	ICMP: Echo
4	[1-A]	[58.120.133.109]	[121.146.158.1]	1514	IP: Continuation of frame 3; (1/5) 1500 Bytes of IP: D=[121.146.158.1] S=[58.120.133.109] LEN=148
5	[1-A]	[58.120.133.109]	[121.146.158.1]	1514	IP: Continuation of frame 3; (2/5) 1500 Bytes of IP: D=[121.146.158.1] S=[58.120.133.109] LEN=148
6	[1-A]	[58.120.133.109]	[121.146.158.1]	1514	IP: Continuation of frame 3; (3/5) 1500 Bytes of IP: D=[121.146.158.1] S=[58.120.133.109] LEN=148

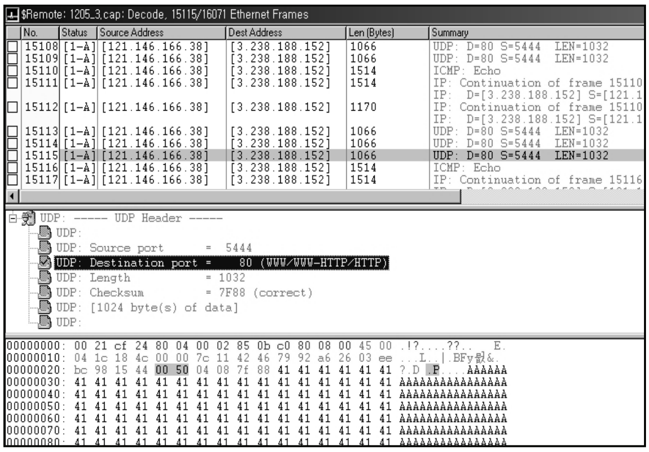
```
IP: Time to live    = 118 seconds/hops
IP: Protocol        = 1 (ICMP)
IP: Header checksum = 2DAA (correct)
IP: Source address      = [58.120.133.109]
IP: Destination address = [121.146.158.1]
```

```
0000000: 00 02 85 0b c0 80 00 21 cf 24 80 04 08 00 45 00 ..??.!?....   E.
0000010: 05 dc 17 1a 22 e4 76 01 2d aa 3a 78 85 6d 79 92 .?."?  .-?x퀵y
0000020: 9e 01 30 31 32 33 34 35 36 37 38 39 3a 3b 3c 3d ?0123456789:;<=
0000030: 3e 3f 40 41 42 43 44 45 46 47 48 49 4a 4b 4c 4d >?@ABCDEFGHIJKLM
0000040: 4e 4f 50 51 52 53 54 55 56 57 58 59 5a 5b 5c 5d NOPQRSTUVWXYZ[\]
0000050: 5e 5f 60 61 62 63 64 65 66 67 68 69 6a 6b 6c 6d ^_`abcdefghijklm
0000060: 6e 6f 70 71 72 73 74 75 76 77 78 79 7a 7b 7c 7d nopqrstuvwxyz{|}
0000070: 7e 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d 2e ~ !"#$%&'()*+,-.
0000080: 2f 30 31 32 33 34 35 36 37 38 39 3a 3b 3c 3d 3e /0123456789:;<=>
0000090: 3f 40 41 42 43 44 45 46 47 48 49 4a 4b 4c 4d 4e ?@ABCDEFGHIJKLMN
00000a0: 4f 50 51 52 53 54 55 56 57 58 59 5a 5b 5c 5d 5e OPQRSTUVWXYZ[\]^
00000b0: 5f 60 61 62 63 64 65 66 67 68 69 6a 6b 6c 6d 6e _`abcdefghijklmn
00000c0: 6f 70 71 72 73 74 75 76 77 78 79 7a 7b 7c 7d 7e opqrstuvwxyz{|}~
```

Fig.7. The result of TCP SYN Attack

Scenario #3: The CGI attack on Web application servers

There are a lot of attack methods to attack web servers. Most of them send endless abnormal packets through port 80 in order to make the resources of web application servers unavailable. We see the payload is filled with garbage data.



$Remote: 1205_3.cap: Decode, 15115/16071 Ethernet Frames

No.	Status	Source Address	Dest Address	Len (Bytes)	Summary
□ 15108	[1-A]	[121.146.166.38]	[3.238.188.152]	1066	UDP: D=80 S=5444 LEN=1032
□ 15109	[1-A]	[121.146.166.38]	[3.238.188.152]	1066	UDP: D=80 S=5444 LEN=1032
□ 15110	[1-A]	[121.146.166.38]	[3.238.188.152]	1514	ICMP: Echo
□ 15111	[1-A]	[121.146.166.38]	[3.238.188.152]	1514	IP: Continuation of frame 15110 IP: D=[3.238.188.152] S=[121.1
□ 15112	[1-A]	[121.146.166.38]	[3.238.188.152]	1170	IP: Continuation of frame 15110 IP: D=[3.238.188.152] S=[121.1
□ 15113	[1-A]	[121.146.166.38]	[3.238.188.152]	1066	UDP: D=80 S=5444 LEN=1032
□ 15114	[1-A]	[121.146.166.38]	[3.238.188.152]	1066	UDP: D=80 S=5444 LEN=1032
□ 15115	[1-A]	[121.146.166.38]	[3.238.188.152]	1066	UDP: D=80 S=5444 LEN=1032
□ 15116	[1-A]	[121.146.166.38]	[3.238.188.152]	1514	ICMP: Echo
□ 15117	[1-A]	[121.146.166.38]	[3.238.188.152]	1514	IP: Continuation of frame 15116

```
UDP: ----- UDP Header -----
  UDP:
  UDP: Source port      =  5444
  UDP: Destination port =    80 (WWW/WWW-HTTP/HTTP)
  UDP: Length           = 1032
  UDP: Checksum         = 7F88 (correct)
  UDP: [1024 byte(s) of data]
  UDP:
```

```
00000000: 00 21 cf 24 80 04 00 02 85 0b c0 80 08 00 45 00 .!?....??..   E.
00000010: 04 1c 18 4c 00 00 7c 11 42 46 79 92 a6 26 03 ee ...L..|.BFy嚴&.
00000020: bc 98 15 44 00 50 04 08 7f 88 41 41 41 41 41 41 ?.D .P....AAAAAA
00000030: 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAA
00000040: 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAA
00000050: 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAA
00000060: 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAA
00000070: 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAA
00000080: 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAA
```

Fig.8. The CGI attack to web servers

As we see from the above, we can conclude that the correctness of judgement on the basis of the rule engine is high.

VI. Conclusion and Future Work

The number of damage cases resulting from distributed denial-of-service (DDoS) attacks has been recently increasing. The need for agile detection and appropriate response mechanisms against DDoS attacks has also been increasing. We need to cope with such abnormal traffic to guarantee customers network QoS in this background. Also, it is necessary to analyze suspicious traffic and block abnormal traffic. The whole process should be performed in real time in order to have effectiveness.

Through the aforementioned process, this paper suggested a rule based abnormal traffic defense mechanism that is capable of detecting massive attacks such as DDoS and Bot, and provides accurate critical values for the operators by analyzing intrusion data. According to the test results, the proposed system makes it possible to determine various attack types occurring in wired networks and provides necessary information for their management and analysis.

However, the assessment of the proposed system used well known attack scenarios. Because there are not objective criteria and information to evaluate defense systems precisely. As for future work, we are going to test the system with a greater number and variety of attack scenarios and compare it with typical rule-based detection systems.

References

[1] MCAFEE AVERT LABS UNVEILS PREDICTIONS FOR TOP TEN SECURITY THREATS IN 2007 AS HACKING COMES OF AGE. Available: http://www.mcafee.com/us/about/press/corporate/2006/20061129_080000_f.html

[2] USING INTERNAL SENSORS FOR COMPUTER INTRUSION DETECTION, Diego Zamboni. Available: https://www.cerias.purdue.edu/tools_and_resources/bibtex_archive/archive/2001-42.pdf, P17

[3] CERT Coordination Center, "Denial of Service Attacks," http://www.cert.org/tech_tips/denial_of_service.html

[4] Paul J. Criscuolo, "Distributed Denial of Service – Trin00, Tribe Flood Network, Tribe Flood Network 2000, and Stacheldraht", CIAC-2319, Feb. 2000.

[5] Chul-hyun Chung, Dae-young Byun, "DoS trace through traffic analysis". Available: http://www.superuser.co.kr/security/certcc/030115-DoS.pdf, 2003.1.

[6] Carl G, Kesidis G, Brooks R.R, Suresh Rai, "Denial-of-service attack-detection techniques", Internet Computing, IEEE, Jan-Feb.2006.

[7] Feinstein L, Schnackenberg D, Balupari R, Kindred D, "Statistical Approaches to DDoS Attack Detection and Response", DARPA Information Survivability Conference and Exposition (DISCEX 2003), April 2003.

[8] Sung-Ju Kim, Dong-Sik Yun, and Byung-Soo Chang, Association Analysis of Customer Services from the Enterprise Customer Management System, ICDM2006, LNAI4605,pp279,2006

[9] Chengqi Zhang and Shichao Zhang., Association Rule Mining: Models and Algorithms, LNAI 2307, Springer-Verlag, Germany 2002

[10] Agrawa R., Imielinski T., and Swanmi A., Database mining: A performance perspective. IEEE Transaction. Knowledge and Data Eng, 5(6), pp914~925, 1993

[11] Han J., Pei J. and Yin Y., Mining frequent patterns without candidate generation, Proceedings of the ACM SIGMOD International Conference on Management of Data, pp1~12, 2000

[12] Wenke Lee, "A Framework for Constructing Features and Models for Intrusion Detection System," PhD thesis, Columbia University, June 1999

[13] L. Perrochon, E. Jang, and D.C. Luckham, "Enlisting Event Patterns for Cyber Battlefield Awareness," DARPA Information Survivability Confirence & Exposition (DISCEX'00), Hilton Head, South Carolina, January 2000

[14] Brian Caswell, Jay Beale, Andrew R. Baker, "Snort Intrusion Detection and Prevention Toolkit", SYNGRESS, 2006.

[15] Eric Y. Chen, "An Active-Network-Powered Defense Mechanism against DDoS Attacks" IWAN'01, Sept, 2001

[16] Denial of Service Attacks, Birger Kuhnel. Available: http://wwwcs.upb.de/cs/ag-madh/WWW/Teaching/2004SS/AlgInternet/Submissions/07-denial-of-service-attacks.pdf