International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:4, No:3, 2010

Proactive Detection of DDoS Attacks Utilizing k-NN Classifier in an Anti-DDos Framework

Hoai-Vu Nguyen and Yongsun Choi

***Abstract*—**Distributed denial-of-service (DDoS) attacks pose a serious threat to network security. There have been a lot of methodologies and tools devised to detect DDoS attacks and reduce the damage they cause. Still, most of the methods cannot simultaneously achieve (1) efficient detection with a small number of false alarms and (2) real-time transfer of packets. Here, we introduce a method for proactive detection of DDoS attacks, by classifying the network status, to be utilized in the detection stage of the proposed anti-DDoS framework. Initially, we analyse the DDoS architecture and obtain details of its phases. Then, we investigate the procedures of DDoS attacks and select variables based on these features. Finally, we apply the k-nearest neighbour (k-NN) method to classify the network status into each phase of DDoS attack. The simulation result showed that each phase of the attack scenario is classified well and we could detect DDoS attack in the early stage.

***Keywords*—**distributed denial-of-service (DDoS), k-nearest neighbor classifier (k-NN), anti-DDoS framework, DDoS detection.

I. Introduction

SECURITY technologies have to keep pace with the rapid development in information technology and network systems in order to protect the systems from attacks. Network security is one of the most important sections of the security domain. Distributed denial-of-service (DDoS) attacks first appeared in June 1998 and rapidly spread causing extensive damages. For instance, during the week of 7–11th of February in 2000, they emerged as the major attacks in the new category of attacks on the Internet. They attacked many well-known sites, including Yahoo, Buy, eBay, Amazon, Datek, E*Trade, and CNN (Todd, 2000). Since DDoS attacks are very powerful and pose a serious threat to the network security, it is important to understand how it works.

DDoS attack involves the combined effort of several machines in attacking a target system. In many cases, the attacker first selects some machines having security vulnerabilities as handlers and gains access to them. Then, the attacker continues to include more machines as zombies through the handlers. The zombies carry out the actual DDoS attacks by significantly increasing the malicious traffic to a target system. As a result, the victim machine loses all its computing and communication resources. Although the technique of DDoS attacks is relatively simple, it can attack both the Internet and system resources.

Since the extent of damage by DDoS attacks has increased, many studies on the detection mechanism have been carried out. However, the existing security mechanisms have failed to provide effective defence against these attacks or just can only provide defence against specific types of DDoS attacks. Some DDoS attack detection methods are based on traceback, while others are based on feature monitoring of a router or a server. However, existing methods have limited success because they cannot simultaneously achieve the objectives of (1) efficient detection with a small number of false alarms and (2) real-time transfer of all packets. For instance, some methods, which apply data mining techniques, can obtain a high correction rate in detecting the attacks. However, these methods usually can't be employed in real-time computing. Other methods, exploiting the abnormal increase in some types of packets, mitigate only some types of DDoS attacks. Furthermore, presently, there exist few effective and detailed model frameworks available for the detection and prevention of DDoS attacks.

In this paper, we first present a general anti-DDoS framework that contains two sequential stages—detection and prevention. Then, we present a method for proactive detection of DDoS attack by classifying the network status to be utilized in the detection stage of the general anti-DDoS framework. More specifically, we describe the two-stage view of DDoS architecture, the control stage and the attack stage. Then, we investigate the procedures of DDoS attacks to select feature variables that are important in recognizing DDoS attacks, since they are to be abnormally changed whenever the attack happens. Finally, we apply the k-nearest neighbor (k-NN) method to classify the status of networks for each phase of the DDoS attack. The simulation result has shown that the phases of the attack have been classified well and DDoS attacks could be detected in the early stage, with efficiency. In sum, we propose a general anti-DDoS framework and an automated method for the early detection of DDoS attacks. We apply the k-NN method for DDoS attack detection with flexible adjustment of feature variables. In addition, we provide a suitable method for mapping a document to an element that describes the period of packet transfer in a network.

The rest of the paper is organized as follows. Section II summarizes previous studies in the area of DDoS attack detection. In section III, we analyze the DDoS architecture

Hoai-Vu Nguyen is with the Department of Systems Management Engineering, Inje University, *621-749, South Korea* (e-mail: nguyenhoaivu_it@yahoo.com).

Yongsun Choi is with the Department of Systems Management Engineering, Inje University, *621-749, South Korea* (phone: +82-55-320-3117; fax: +82-55-320-3632; e-mail: yschoi@ inje.ac.kr).

and introduce a general anti-DDoS framework. Next, in section IV, we describe the proposed method for the early detection of DDoS attacks in detail. Section V presents the data collected and the simulated results. Finally, in section VI, we conclude our works with directions of further studies.

II. RELATED STUDIES

Thus far, many results related to DDoS attack defence have been reported. We can classify DDoS attack defence methods into congestion-based, anomaly-based, source-based methods, and others. Existing methods either belong to any one of these categories or are a combination of them [1]-[9]. Mahajan et al. (2001) [11] and Ioannidis and Bellovin (2002) [6] have proposed an aggregate-based congestion control (ACC) that decreases DDOS attack traffic on the basis of the congestion level. The detection algorithm in ACC determines the destination addresses of the victim machines on the basis of the destination of a network prefix of packets dropped at the observed router during a very short period. ACC will set the destination address to list if the number of dropped packets with a specific destination address is greater than the average number. If the arrival rate of a network prefix exceeds the threshold, ACC marks all traffic to this network prefix as DDoS attack traffic and responds to all incoming traffic sent to this network prefix. Cabrera et al. (2001) [1] and May et al. (2001) [12] have proposed a method based on network management information to detect DDoS attacks. The local Simple Network Management Protocol (SNMP) agents update the variables in a management information base (MIB) periodically. Hence, the network management system analyses the MIB variable correlations during the attack preparation, attack, and normal state to detect DDoS attacks. This method is efficient only if the victim host and attacker are on the same network. It is unable to solve the problem when the victim and attacker are on different networks.

Yaar et al. (2003) [20] have presented a method based on IP traceback and packet filtering to mitigate DDoS attack traffic. Packet marking identifies the paths followed by the attack traffic by inserting marks in packets. They introduced intelligent packet filtering method to filter out the ongoing attack traffic. However, the length of IP identification field is limited to only 16 bits in this method, which is not sufficient for storing the entire path. In addition, certain coding schemes have to be applied to shorten the length of marks. Gavrilis and Dermatas (2005) [3] have presented a DDoS attack detector in public networks utilizing the radial basis function neural network (RBFNN), which is originally introduced by Haykin (1994) [5]. Their method is based on the statistical features estimated in short time window analysis of the incoming data packets. This method is supported by three modules as: a data collector, features estimator, and DDoS detector. The DDoS detector is a two-layer neural network with nine feature vectors that are used to activate a two-output RBF network at each time frame. The most active output neuron detects the presence of a DDoS attack or characterizes the time frame as normal traffic. This approach is highly efficient, but has some weak points such as long computing time. Lee (2006) [9] has presented an improved marking technique that identifies DDoS traffic with time to live (TTL) information at the routers by applying the support vector machine (SVM) module to control malicious traffic and manage DDoS attack packets efficiently. This method can filter malicious traffic with the SVM congestion signature and improves the bandwidth of the entire network. Hence, it is possible to restructure the path to the source of DDoS attacks with a small number of marking packets. A disadvantage of this method is the requirement of additional memory at the routers for the DDoS-related identification performed by the SVM-based filtering module. Xu et al. (2007) [19] have proposed a novel DDoS detection method based on hidden Markov models (HMMs) and cooperative reinforcement learning in which a distributed cooperation detection scheme using source IP address monitoring is employed. To realize earlier detection of DDoS attacks, the detectors are distributed at intermediate network nodes or near the sources of DDoS attacks and HMMs are used to establish a profile for the normal traffic on the basis of the frequencies of the new IP addresses. A cooperative reinforcement learning algorithm computes the optimized strategies for information exchange among the distributed multiple detectors so that the detection accuracies can be improved without high load on information communication among the detectors. However, the evaluation of the HMM-based approach in the real-time DDoS detection cases is not included in this paper, requiring additional algorithms to be applied to realize a better balance between detection accuracy and communication load.

The abovementioned methods and others focus much on the change in traffic flow. Methods based on data mining are suitable for detection, but they still can't ensure frequent transfer of packets. As like the methods based on neural networks, it is not easy to apply HMMs in the real world due to long computation time. On the other hands, some methods can be applied only to specific types of DDoS attacks, that is they are of limited usefulness and efficiency. To overcome these limitations of existing methods, we apply the k-NN classifier which is proved to be useful and very efficient when to classify documents [4]. The way of applying the k-NN method, for the early detection of DDoS attacks, is to be described in detail in section IV.

III. DDOS ATTACH ARCHITECTURE AND ANTI-DDOS FRAMEWORK

In this section, we describe the characteristics of DDoS attacks with two-stage view of DDoS architecture, the control stage and the attack stage. And then, we introduce a simple anti-DDoS framework that comprises stages of detection and prevention of DDoS attacks

A. DDoS Attack Architecture

DDoS attacks first appeared in June 1998. The attacks start by breaking into hundreds or thousands of machines (handlers) over the Internet. Then, the attacker installs DDoS software on the machines, allowing them to control all the

attacked machines (zombies or agents) to launch coordinated attacks on target sites. These attacks typically exhaust the network bandwidth, router processing capacity, or network stack resources, and disrupt the network connectivity to the victims. Different types of DDoS attacks have been developed, which can be classified as TCP flood, UDP flood, ICMP flood, and smurf [18]. The general architecture of DDoS attacks determined by Lin and Tseng (2004) [10] is shown in Fig. 1.

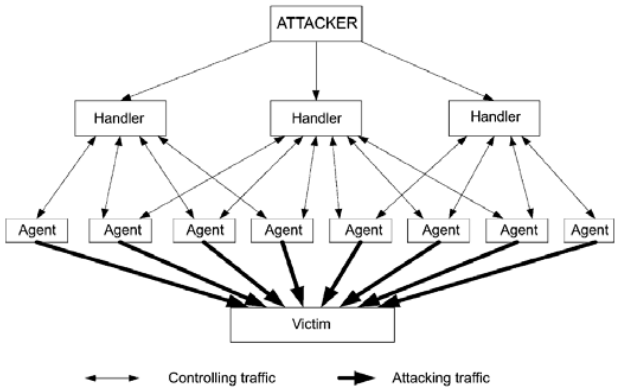

Fig. 1 General architecture of DDoS attacks

The general architecture of the DDoS attack shown in Fig. 1 can be divided into two stages:

- Control stage
- Attack stage

In control stage, a scan is performed on a large scale on the network to find a list of vulnerable hosts. Generally, the vulnerable hosts consist of handlers and agents, where the handlers (the first level vulnerable hosts) are controlled by the attackers and the agents (the second level vulnerable hosts) are controlled by attackers through handlers. The traffic of communication in the control stage takes place through signal transmission from an attacker to a handler; however, the communication between the handlers and agents is bidirectional. The two levels of topology in the locations of attackers can be hidden. At the end of the control stage, the vulnerable hosts are used to launch distributed attacking traffic in the attack stage. The attacking traffic including UDP flood, ICMP flood, Smurf, TCP SYN, TCP ACK, TCP RST, and TCP SYN/ACK can overwhelm the victim [18]. There are two different types of attack techniques followed by DDoS attacks: bandwidth consumption and resource consumption. In bandwidth consumption, the attacking traffic launched by the compromised hosts, which are controlled by the attackers, is aggregated to a single large flood that overwhelms the victim. In resource consumption, the attackers can use the leak of the network protocol or the system security, such as the techniques of SYN flood, land, and Teardrop. This results in the starvation of system resources (CERT/CC, 2003). As the DDoS attack tools have become more complicated in the recent years, it is becoming more difficult to encounter the up-to-date characteristics of DDoS attacks.

B. *Anti-DDoS Framework*

In the security domain, the intrusion detection system (IDS) and intrusion prevention system (IPS) are well known [2]-[17]. In similar way, we construct our anti-DDoS framework containing two sequential stages of *DDoS attack detection* and *DDoS attack prevention*. However, we need to differentiate DDoS attacks from intrusive activities. The definitions of detection and prevention in the context of DDoS attacks are different from those in the context of intrusive activities. Fig. 2 shows all the components in each stage of the anti-DDoS framework in detail.

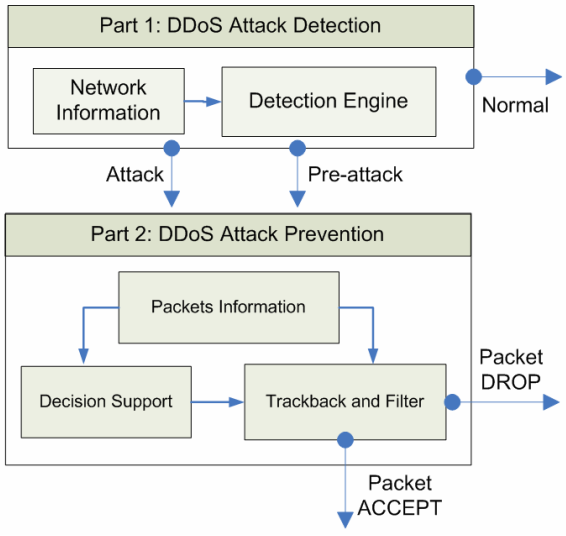

Fig. 2 A simple anti-DDoS framework

In case of DDoS attacks, it is difficult to protect a system some time after the attacker initiated attacks as shown in figure 1. Hence, we should detect the DDoS attacks in the early stage. The first stage of the anti-DDoS framework carries out the early detection of DDoS attacks, which will be described in detail in section IV. In case 'pre-attack' or 'real-attack' of network status is detected, detailed network status information is transferred to the prevention stage to mitigate DDoS attacks. In this paper, we just focus on the early detection of DDoS attacks without digging into detailed mechanism of preventing DDoS attacks. That is our anti-DDoS attack framework only 'mitigates' the DDoS attack without determining the exact attacker host. There already exist many methods of preventing DDoS attacks [15].

IV. K-NN METHOD FOR EARLY DETECTION OF DDOS ATTACKS

Lee (2007) [8] has proposed an efficient method for proactive detection of DDoS attacks using cluster analysis. In his study, he analysed in detail the characteristics of the selected variables, which are used for clustering by using the cubic clustering criterion (CCC). We will use these features in our method. Lee divided DDoS attacks to three phases so the status of network will be four types. However, as mentioned before, we have decided to classify the status of network into three classes:

- Pre-attack: includes the first two phases.
  - Phase 1 of DDoS attack—selection of handlers and agents
  - Phase 2 of DDoS attack—communication and compromise
- Attack: includes phase 3 of DDoS attack—attack
- Normal status of network

Moreover, since these classes are well divided, we decide to apply a classifying method for the early detection of DDoS attacks. By employing the classifying method, the detection will becomes more accurate and take shorter time for computing than the case when the clustering method is applied. In the classifying module, we choose the k-NN method to classify because it achieves the two objectives: accurate detection rate and short time computing. The k-NN method had been devises a long time back and is still useful. For instance, the k-NN method is used to classify documents by Reuters News, which is one of the most famous news agencies across the world [4].

A. Selection Features for Detecting DDoS

We studied the procedures of DDoS attacks to primarily select the packets and traffic parameters that change unusually in each phase of the attack. Lee (2007) [8] has mentioned some parameters such as source/destination IP addresses, port numbers, and packet types (ICMP, TCP SYN, UDP) that will used as features to detect DDoS attacks.

In the pre-attack phase, the attacker spreads packets to find the machines that have security vulnerabilities to intrude them and gain access to them. During this period, the destination IP address will be distributed randomly. However, in the last phase of attack—launching DDoS attack—the destination IP address will remain fixed or rarely change. To measure this change, Lee (2007) [8] had suggested using the concept of entropy.

If the information source has n independent symbols each with a probability of choice Pi, the entropy H is defined as follows:

$$H = \sum_{i=1}^{n} P_i \log_2 P_i \quad (1)$$

The other characteristic is the occurrence rate of a type of packet. These characteristics have been exploited and various methods have been developed to detect DDoS attacks. During the launch of DDoS attacks, there are some types of packets (DDoS attacks using a specific packet type) that change abnormally.

Finally, we use the following features of packet transfer, which Lee (2007) [8] had presented:

- Entropy of source IP address and port number
- Entropy of destination IP address and port number
- Entropy of packet type
- Occurrence rate of packet type (ICMP, UDP, and TCP SYN)
- Number of packets

We use these features as the gradients of the vector describing a period of network status. Next, we discuss the method that is used for classification.

B. K-NN Classifier

First, we select the features for detecting DDoS attacks and classify the network status to three classes. Next, we consider the classification of the current network status to one of the classes. There are many well-known methods for classifying documents such as SVM, NN, fuzzy logic, and rough set [14]. We choose the k-NN method because this method has features that are suitable for our goals. These features are: easy implementation, short time computation, and high accuracy.

The k-NN algorithm is a similarity-based learning algorithm and is known to be highly effective in various problem domains, including classification problems. Given a test element *dt*, the k-NN algorithm finds its k nearest neighbors among the training elements, which form the neighborhood of *dt*. Majority voting among the elements in the neighborhood is used to decide the class for *dt*. For the example shown in Fig. 3, we first find k elements that are nearest to the element to be classified. From the k nearest elements, we determine the most suitable class for the test element [4]-[16].

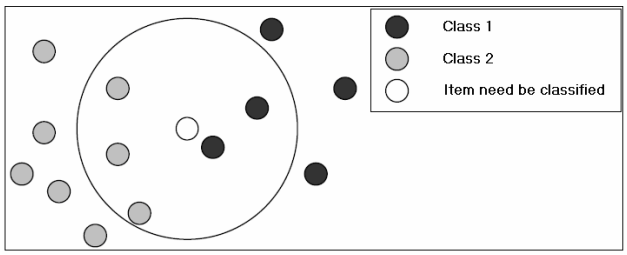

Fig. 3 Finding k elements that are nearest to the test element, k = 5

The term ‘near’ can be defined as the degree of similarity between two elements. There are several techniques to compute the similarity degree between two elements. However, the algorithm based on the cosine formula is most popular method used for estimating the similarity degree. In this study, we use this algorithm to compute the similarity degree. Besides, we also use the vector space model (VSM) to describe each element. Hence, each element is expressed as a vector that has n components. The example is given below.

For the 2 elements $X = \{x_1, x_2, ..., x_n\}$ and $Y = \{y_1, y_2, ..., y_n\}$, $W = \{w_1, w_2, ..., w_n\}$ is the weighted vector and w_i is the weight of the component i in the general vector. Then, we compute the similarity between two elements X and Y as follows:

$$Similarity(X, Y) = Cosine(X, Y, W) = \frac{\sum_{i=1}^{n}(x_i \times w_i) \times (y_i \times w_i)}{\sqrt{\sum_{i=1}^{n}(x_i \times w_i)^2}\sqrt{\sum_{i=1}^{n}(y_i \times w_i)^2}} \quad (2)$$

Using the abovementioned cosine formula, we can find the

k nearest elements. Next, we have to determine the most suitable class for these elements. We count the rate of each class types to determine the class that has the highest rate. This is the class in which the test element can be placed.

C. Early Detection of DDoS Attacks Using k-NN Classifier

We use the nine features that have been discussed in part A to classify the network status. Each variable is normalized to eliminate the effect of difference between the scales of the variables, as proposed by Lee et al. (2007) [8]. With normalization, variables become

$$z = \frac{x - \bar{x}}{\sigma} \tag{3}$$

where x, $\bar{x}$, σ, denotes the value of each feature, the mean of the sample dataset, and the standard deviation, respectively.

To classify the current network status, we use the k-NN classifier, which has been explained previously. Firstly, we train three datasets—normal, pre-attack, and attack datasets. Each element in each dataset has nine components that are computed from the data log for the period τ. We compute the current network status as an element with nine components in τ period. Finally, we apply the distance formula (3) to find the k nearest neighbors of the current network status. We set a label for the current network status based on the majority of the elements belonging to a class, in which most elements among the k elements are found. Hence, this aids in the recognition of the current network status and early detection of DDoS attacks. The details of the detection of DDoS attacks are shown in Fig. 4.

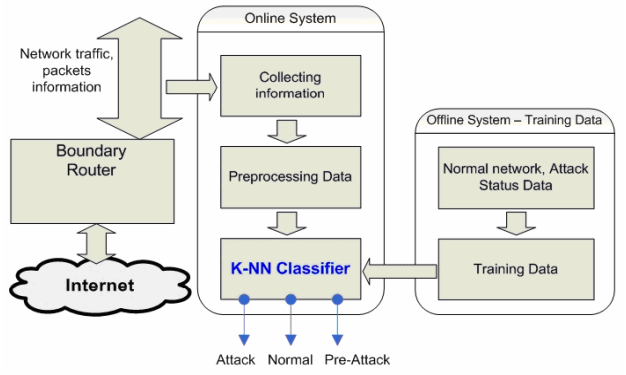

Fig. 4 General model for detecting precursor of DDoS attacks

V. SIMULATION RESULTS

Using the 2000 DARPA intrusion detection scenario specific data set (MIT Lincoln Lab, 2000) [13], we employ the proposed method for early detection of DDoS attacks. This dataset includes a DDoS attack launched by a novice attacker. This attack is carried out over multiple networks and audit sessions. These sessions have been grouped into 5 attack phases over the course of which the adversary probes break in, install Trojan mstream DDoS software, and launch a DDoS attack on an off-site server.

The five phases of the attack scenario are:

1- IPsweep of the AFB from a remote site
2- Probe of live IP's to look for the sadmind daemon running on Solaris hosts
3- break-ins via the sadmind vulnerability, both successful and unsuccessful on those hosts
4- Installation of the Trojan mstream DDoS software on three hosts at AFB
5- Launch of DDoS attacks

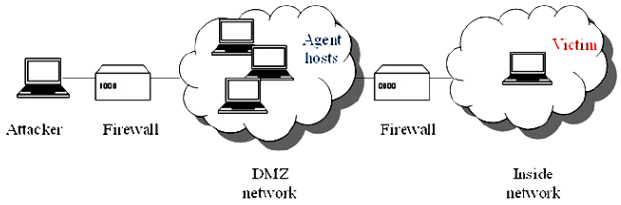

Fig. 5 Architecture of network used to obtain dataset

An attack has five phases. However, in this study, we slightly regroup these phases to two groups:

- Pre-attack phase: includes the first four abovementioned phases
- Attack phase: includes the last phase

We also have the dataset for the normal network status. Hence, we have three groups of datasets for training and testing. All elements in a group are trained as mentioned in section IV, part C. The following elements are obtained.

- Normal class: $N_1, N_2, ..., N_L$
- Pre-attack class: $P_1, P_2, ..., P_L$
- Attack class: $R_1, R_2, ..., R_L$

where $N_i = (x_1, x_2, ..., x_9)$, $P_i = (y_1, y_2, ..., y_9)$, and $R_i = (z_1, z_2, ..., z_9)$.

The testing dataset will be obtained independently from the training dataset to ensure the accuracy of the process. The steps involved in the experiment are illustrated detail in Fig. 6.

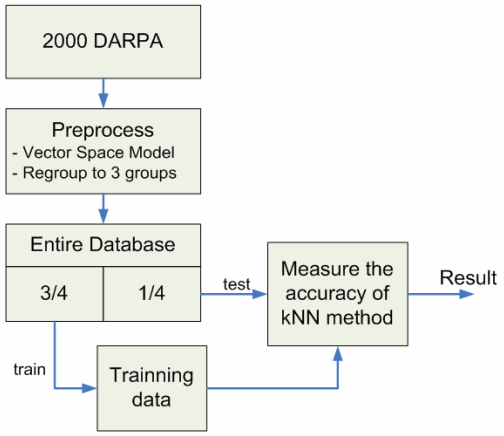

Fig. 6 Scheme of experiment

The result obtained from the experiment is presented in TABLE I.

TABLE I
CLASSIFICATION RESULT

Network status class	Number of test elements	Correct classification	Incorrect classification
Normal	3000	2790	210
Pre-attack	2500	2174	326
Attack	1500	1468	32
Sum	7000	6432(91.886%)	568 (8.114%)

The result of the experiment shows that our method is efficient enough for early detection of DDoS attacks. It can classify all elements well in a suitable computing time.

VI. CONCLUSIONS

In this study, we have introduced a general anti-DDoS framework, which can be applied and developed in the real world. We have also presented a suitable method for the early detection of DDoS attacks using the k-NN classifier. This method can also be applied to the first stage of our anti-DDoS framework. Many studies on DDoS attack detection have been carried out; however, they focus only on the change in network traffic. The methods based data mining are suitable for the detection; however, they do not ensure real-time transfer of packets. Our method first selects nine features of packet/traffic that are widely found in various phases of the attack. Then, the current network status is classified to determine the class to which it belongs to. Hence, our method can classify the current network status well to detect DDoS attacks early.

To evaluate this detection method, we analyzed the MIT Licon Lab Dataset (2000 DARPA: Scenario DDoS 1.0) [13] and the dataset for the normal network status. The result shows that our method can classify the DDoS phases correctly and efficiently detect DDoS attack early. Besides, the method being simple can be easily implemented. Short computing time and real-time transfer of packets can be achieved.

In the future, we will carry out a detailed analysis of the features of DDoS attacks using more advanced k-NN method or other methods and obtain a better result. Finally, we will apply the method in practical situations and study the behavior of DDoS attacks and make modifications if possible. Moreover, we will develop a suitable and efficient method for the second stage of the anti-DDoS framework.

ACKNOWLEDGMENT

This work was supported by the Korea Science and Engineering Foundation (KOSEF) grant funded by the Korea government (MOST) (No. R01-2007-000-21070-0).

REFERENCES

[1] J.B.D. Cabrera, et al. "Proactive detection of distributed denial of service attacks using MIB traffic variables—a feasibility study", Proceedings of the seventh IFIP/IEEE International Symposium on Integrated Network Management, Seattle, May, 2001, pp. 1–14.

[2] S. Chebrolu, A. Abraham, and P. J. Thomas, "Feature deduction and ensemble design of intrusion detection systems", Computers & Security, Vol. 24, issue 4, pp. 295–307. 2005.

[3] D. Gavrilis, and E. Dermatas, "Real-time detection of distributed denial-of-service attacks using RBF networks and statistical features", Computer Networks, Vol. 48, issue 2, pp. 235–245. 2005.

[4] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, "Using kNN model for automatic text categorization", Soft Computing - A Fusion of Foundations, Methodologies and Applications, Vol. 10, No. 5, pp. 423-430. 2006.

[5] S. Haykin, *Neural Networks: A Comprehensive Foundation,* Upper Saddle River, Prentice Hall, New Jersey, 1994.

[6] J. Ioannidis, and S. M. Bellovin, "Implementing pushback: router-based defense against DDoS attacks", Presented at Network and Distributed System Security Symposium, 2002.

[7] M. Kim, H. Na, K. Chae, H. Bang, and J. Na, "A Combined Data Mining Approach for DDoS Attack Detection", ICOIN 2004, LNCS 3090, Springer-Verlag, Berlin Heidelberg, pp. 943–950.

[8] K. Lee, J. Kim, K. H. Kwon, Y. Han, and S. Kim, "DDoS attack detection method using cluster analysis", Expert Systems with Applications, 2007, Vol. 34, pp. 1659–1665.

[9] H. W. Lee, "SVM Based Packet Marking Technique for Traceback on Malicious DDoS Traffic", ICOIN 2006, LNCS 3961, Springer-Verlag, Berlin Heidelberg, pp. 754–763.

[10] S. C. Lin, and S. S. Tseng, "Constructing detection knowledge for DDoS intrusion tolerance", Expert Systems with Applications, 2004, Vol. 27, pp. 379–390.

[11] R. Mahajan, S. M. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker, "Controlling high bandwidth aggregate in the network", ACM SIGCOMM Computer Communication Review, 2002, Vol. 32, No. 3 pp. 62 - 73.

[12] J. May, J. Peterson, and J. Bauman, "Attack detection in large networks", Proceedings of the DARPA Information Survivability Conference & Exposition II (DISCEX '01), 2001, Vol. 1, pp.15–21.

[13] MIT Lincoln Lab, 2000, DARPA intrusion detection scenario specific datasets, http://www.ll.mit.edu/IST/ideval/data/2000/2000_data_index.html.

[14] T. M. Mitchell, *Machine Learning,* MacGraw Hill, New York, 1996.

[15] K. Park, and H. Lee, "A proactive approach to distributed DoS attack prevention using route-based packet filtering", Tech. Rep. CSD-00-017, Department of Computer Sciences, Purdue University, 2000.

[16] F. Sebastiani, "Machine learning in automated text categorization", ACM Computing Surveys, Vol. 34, issue 1, Consiglio Nazionale delle Ricerche, Italy, 2002, pp. 1–47.

[17] A. Sharma, A. K. Pujari, and K. K. Paliwal, "Intrusion detection using text processing techniques with a kernel based similarity measure", Computers & Security, 2007, Vol. 26, issue 7–8, 2007, pp. 488–495.

[18] B. Todd, "Distributed Denial of Service Attacks", 2000. http://www.linuxsecurity.com/resource_files/intrusion_detection/ddos-faq.html

[19] X. Xu, Y. Sun, and Z. Huang, "Defending DDoS Attacks Using Hidden Markov Models and Cooperative Reinforcement Learning", Yang C.C. et al. (Eds.): PAISI 2007, LNCS 4430, Springer-Verlag, Berlin Heidelberg, pp. 196–207.

[20] A. Yaar, A. Perrig, and D. Song, "Pi: a path identification mechanism to defend against DDos attack", Proceedings of the IEEE Symposium on Security and Privacy, 2003, pp. 93–107.

Hoai-Vu Nguyen is a graduate student in the Department of Systems Management & Engineering, Inje University, South Korea. He received his B.S. degree from the Department of Information Technology, Hanoi University of technology, Vietnam. His research interests are network security, software engineering, natural language processing, and business process management.

Yongsun Choi is the Director of BPM Laboratory and a Professor in the Department of Systems Management & Engineering at Inje University, South Korea. He received his B.S. degree in Industrial Engineering from Seoul National University and his M.S. and Ph.D. degrees in Industrial Engineering from Korea Advanced Institute of Science and Technology. His research interests include workflow & business process management, service oriented architecture, and multiple-criteria decision making.