DDoS Attack Detection Using Flow Entropy and Packet Sampling on Huge Networks

Jae-Hyun Jun
School of Computer Science and Engineering
Kyungpook National University
Daegu, Republic of Korea
jhjun@mmlab.knu.ac.kr

Dongjoon Lee
School of Computer Science and Engineering
Kyungpook National University
Daegu, Republic of Korea
djlee@mmlab.knu.ac.kr

Cheol-Woong Ahn
Division of Digital Contents
Keimyung College University
Daegu, Republic of Korea
homepig@kmcu.ac.kr

Sung-Ho Kim
School of Computer Science and Engineering
Kyungpook National University
Daegu, Republic of Korea
shkim@knu.ac.kr

***Abstract*— While the increasing number of services available through computer networks is a source of great convenience for users, it raises several concerns, including the threat of hacking and the invasion of user privacy. Hackers can easily block network services by flooding traffic to servers or by breaking through network security, hence causing significant economic loss. It is well know that a Distributed Denial of Service (DDoS) attack, which robs the targeted server of valuable computational resources, is hard to defend against. In order to address and nullify the threat to computer networks from DDoS attacks, an effective detection method is required. Hence, huge networks need an intrusion detection system for real-time detection. In this paper, we propose the flow entropy- and packet sampling-based detection mechanism against DDoS attacks in order to guarantee normal network traffic and prevent DDoS attacks. Our approach is proved to be efficient via OPNET simulation results.**

Keywords-packet sampling; flow entropy; ddos detection; Network Security;

I. INTRODUCTION

Novel and ever-varying network services are being developed and launched as the rapid growth of the Internet and online users continues. A 2009 investigation by the German company Ipoque shows that Peer-to-Peer (P2P) traffic has constituted more than 60% of Internet traffic for the last few years, and will be responsible for a sizeable portion of it in the foreseeable future [1].

While the Internet provides numerous services through computer networks that make our lives easier, this convenience comes at the cost of ever-rising Internet crime, generally, in the form of hacking and similar invasions of privacy. These crimes cause significant economic damage by flooding network servers or hindering services by gaining access to the relevant computer systems [2].

The Distributed Denial of Service (DDoS) is not a new attack technology. While it first appeared in the late 1990s, the first well-publicized DDoS attack occurred in 2000 against major Internet corporations including Yahoo, Amazon, CNN and eBay. It has been more than ten years since a major DDoS attack has occurred. However, DDoS attacks are among the greatest threats for Internet infrastructure and for the information technology environment.

A DDoS attack occurs when the intruder, also called the *attacker*, invades one or more systems online. The initially compromised system is typically one with a large number of users and a high Internet bandwidth. The attacker then installs the attack programs on the initially compromised system, called the *DDoS master*. The master is then used to find other systems on the network that are vulnerable, and installs DDoS agents, called *daemons*, on these. Using the master system, the attacker then instructs the DDoS daemons to attack the intended target, or *victim*, of the DDoS attack. Hence, the conceptual node of a DDoS attack is comprised of attacker, master, daemon or zombie, and victim. Table 1 shows the explanation of each node. The structure of a DDoS attack is represented in Figure 1 [3].

Since it is not easy to distinguish between a DDoS attack and normal traffic, it is possible to misjudge a normal data packet as a DDoS attack packet. Thus, in order to protect a system from DDoS attacks, a method for an accurate analysis of incoming traffic and the detection of a DDoS attack therein takes pragmatic precedence.

TABLE I. ROLE OF DDOS ATTACK NODES [3]

NAME	ROLE
Attacker	Attacker who is leading all attack operates with an instrument by remote control and delivers commands directly.
Master	Master receives the commands from attacker and orders attack zombies managed by this master.

 ISBN: 978-1-61208-318-6

Zombie	They are controlled by master. Attack program operates the commands that came from each master, and finally performs their attack to the victims.
Victim	As for final victims, simultaneously they are attacked from several hosts.

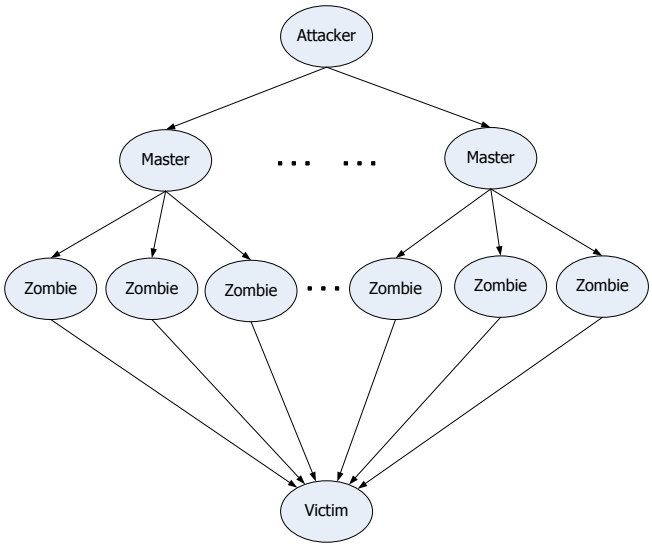

Figure 1. The structure of DDoS attack

Figure 2 shows the annual number of DDoS attacks on Korea Internet & Security Agency (KISA) [11]. As shown, in 2010, there were 6 small-scale 1Gbps DDoS attacks, 4 middle-scale attacks each within the 1~5Gbps and 5~10Gbps bandwidth ranges respectively, and 10 large-scale attacks of more than 10Gbps. By contrast, in 2012, 56 small-scale DDoS attacks occurred within the 1Gbps range, 21 and 25 middle-scale attacks within 1~5Gbps and 5~10Gbps respectively, and 36 large-scale attacks of bandwidth over 10Gbps. It is evident, then, that DDoS attacks are increasing in number by the year.

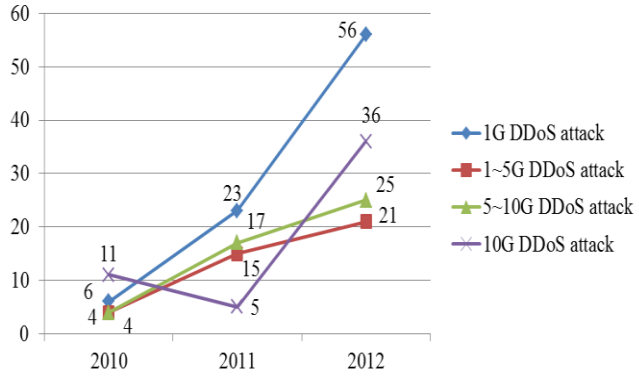

Figure 2. Number of annual DDoS attack on KISA [11]

This paper is structured as follows. In Section 2, we introduce DDoS attack and detection methods. In Section 3, we explain the DDoS attack detection method that uses using flow entropy and packet sampling on a large network, and the results of the testing and evaluation of this method are presented in Section 4. In the final section, we will reflect on our findings.

II. RELATED WORK

A. DDoS background

DDoS attacks first emerged as a kind of massive traffic-generation attack. In some of the first ones of this sort, attackers were able to harness a large amount of network traffic and transmit them to the target systems. In 2000, Yahoo and Amazon's web sites were targeted with such DDoS attacks. Several tools, such as Tribe Flood Network (TFN), TFN 2000 (TFN2K), Trinoo, Stacheldracht, etc., were employed for this type of attack. At the time, researchers were developing traffic anomaly detection techniques for huge networks with a lot of traffic. While it was possible to detect DDoS-type attacks with network traffic anomaly detection techniques on account of higher-than-usual bandwidth usage, it was very difficult to effectively block them. This was because even if a DDoS attack was detected, there was no way to accurately identify the specific attack packets due to IP address spoofing techniques.

Internet worms attack vulnerable systems and take them over automatically. In one such attack, the now-infamous 'Slammer Worm' infected more than 75,000 machines in 10 minutes, causing several network servers worldwide to crash. Figure 3 depicts the global scale of the outbreak [4].

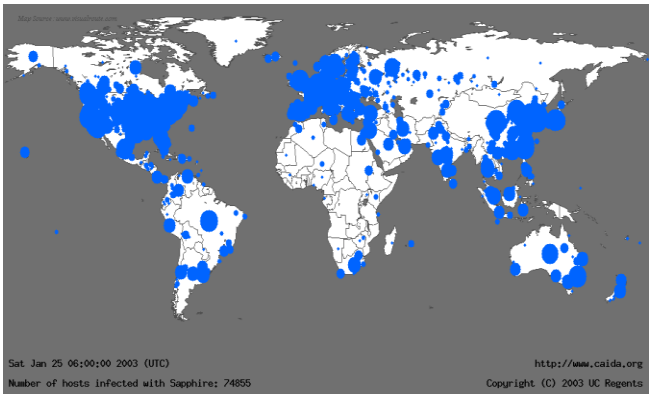

Figure 3. Geographic propagation of Slammer worm 30 minutes after release [4].

Since the mid-2000s, DDoS attack trends have changed. These days, DDoS attacks are primarily launched for economic gain. For instance, a hacker may demand payment from a company in exchange for not attacking its systems. For attacks of this sort, the hacker would prefer not to paralyze entire networks, but would only want the application-layer service to be unavailable to users. In order to do this, the attacker does not have to generate a massive amount of traffic. In fact, if the attack traffic is generated in a sophisticated manner, application-layer services could easily be brought down using only several kbps of attack traffic bandwidth. The attack data packet in this case resembles a

 ISBN: 978-1-61208-318-6

normal packet. It is thus, very difficult to detect an application-layer DDoS attack using only attack-traffic bandwidth analysis or packet-based attack detection methods, which, nowadays, are widely used to defend against DDoS attacks.

A pertinent instance of the above issue is a large DDoS attack that lasted from July 5 to July 10, 2009, launched against 48 websites in the United States and South Korea using tens of thousands of zombie PCs. The attackers in this case used numerous techniques, such as TCP SYN flooding, UDP/ICMP flooding, HTTP GET flooding, and CC attacks. While these attacks were being attempted, the ones on the HTTP service were not effectively preventable. This is because almost all DDoS detection techniques are based on bandwidth variation and the volume of traffic. Even though the aggregate volume of attack traffic was huge, these techniques failed to detect the exact attackers because the amount of traffic from each attack system was not sufficiently high for it to be located. For application-layer DDoS attack detection [5], it is necessary to develop an application-behavior-based attack detection technique [6].

B. DDoS Detection research

Machine learning can be roughly divided into two parts: Supervised and Unsupervised learning methods. Supervised methods use labeled data for training. A supervised learning approach uses labeled 'training data' to classify traffic as normal or otherwise [9]. Unsupervised learning methods use unlabeled data samples. A typical example is clustering. When the data flows in, it clusters the data into different groups [7]. With the incoming data so divided, the program can then inspect and detect abnormal data packets, such as those used for DDoS attacks, by any of a variety of detection methods.

In [8], the flow is formed using a quintuple, which consists of source/destination IP addresses, source/destination port numbers and the protocol. The entropy of four of the features -- source/destination IP address and port number -- is calculated to form clusters. The information is saved to the entropy cube, based on the destination IP address. If the entropy values of the source IP address and port number are higher than a certain preassigned value, or if the entropy value of the destination port number is lower, the entropy cube labels them as a DDoS attack.

A classification problem arises if the entropy of heavy traffic has a value similar to that of a DDoS attack. Hence, we propose that incoming traffic be classified by using flow entropy as well as packet sampling of data.

III. PROPOSED METHOD

In order to detect DDoS attack flows on huge networks, we classify flow using packet sampling, as well as considering measures of flow entropy, the average entropy, the entropy of the source port and the number of packets/second. The flowchart in Figure 4 depicts our proposed method.

From incoming traffic, we extract one of every five data packets for sampling. Figure 5 shows the packet sampling on a router. The sampled packets are collected during a 'time window'.

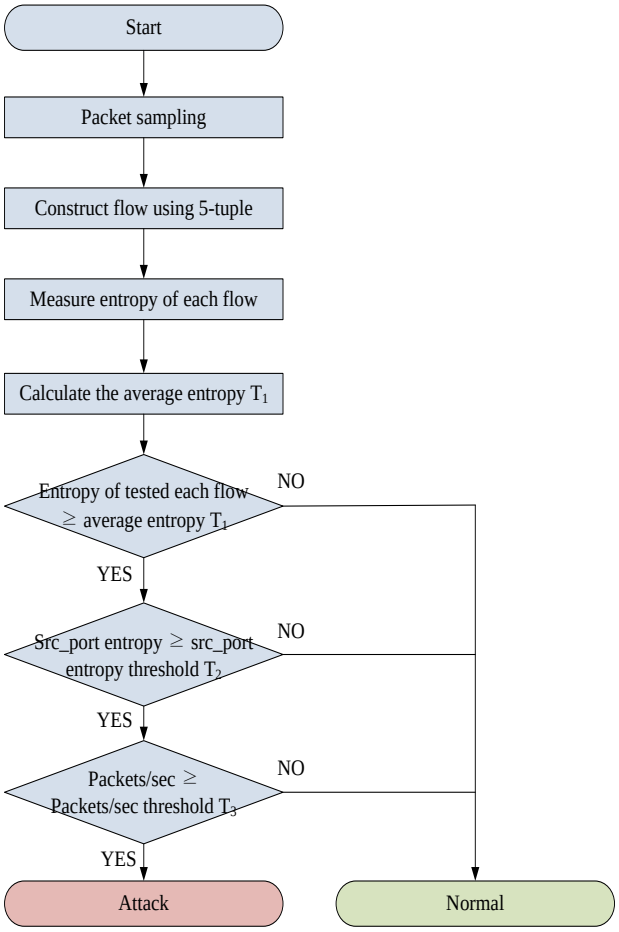

Figure 4. Proposed method

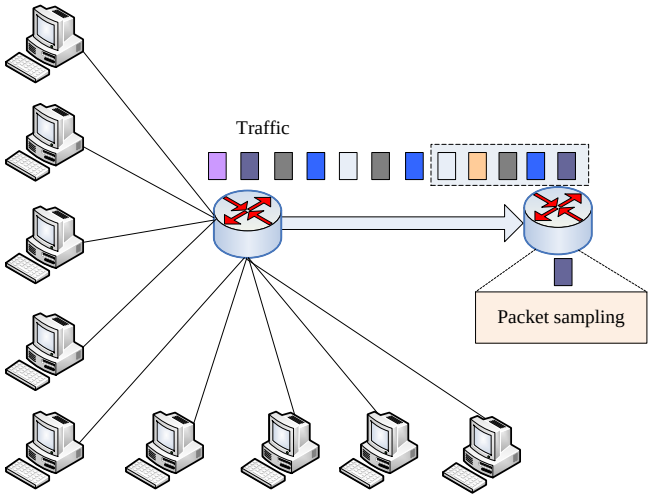

Figure 5. Packet sampling on router

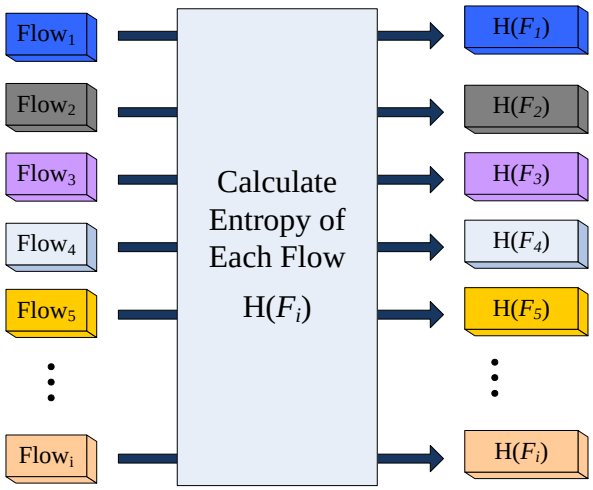

Figure 6. Measure entropy of each flow

The physical definition of entropy was offered by the German physicist Rudolf Clausius, in 1865 [10]. Since entropy causes uncertainty, it is impossible to accurately predict what happens next in an entropic situation. However, if entropy decreases, uncertainty decreases as well. In such cases, entropy may be calculated using the following equations:

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

In Equation (1), n is the number of features (packet number, source port, packets/sec), P_i is the probability of feature i.

$$H(F_{avg}) = \frac{-\sum_{i=1}^{n} H(F_i)}{N(H(F_i))} \quad (2)$$

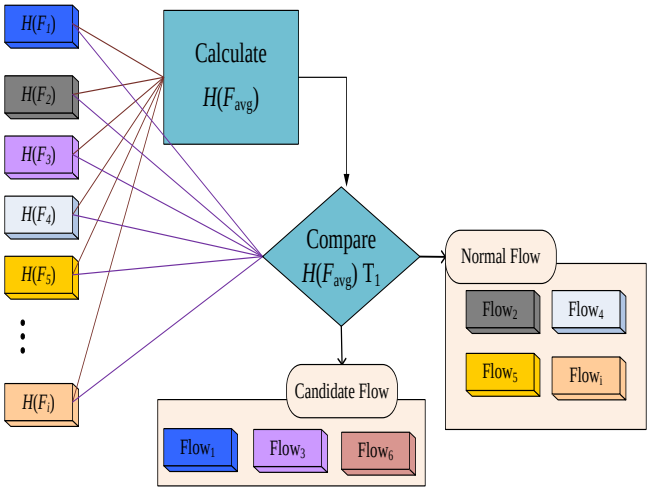

Figure 7. Calculate the average entropy T_1

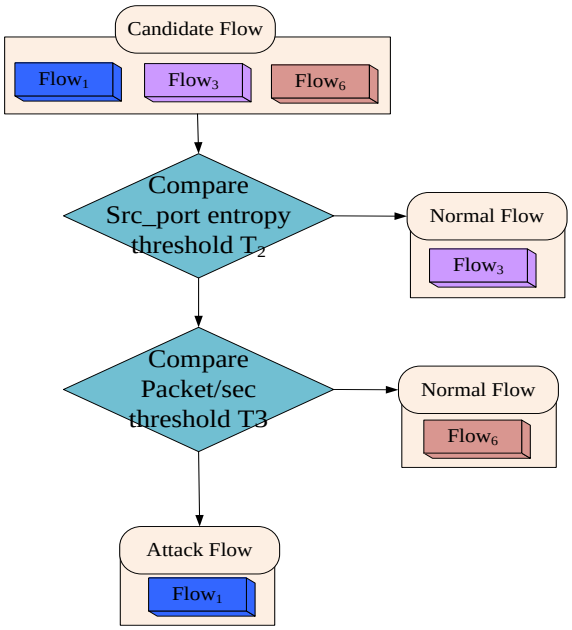

Figure 8. Compare source port entropy T_2 and packet/sec T_3 threshold

In Equation (2), $N(H(F_i))$ is the total flow. In general, DDoS attacks consist of several attack packets. In order to evaluate a flow with several sampled packets, we compare its entropy obtained from (1) with the average calculated entropy $H(F_{avg})$ of the system:

$$\begin{cases} H(F_i) \le H(F_{avg})\ T_1, & Normal\ Flow\ N(F_i) \\ H(F_i) > H(F_{avg})\ T_1, & Candidate\ Flow\ C(F_i) \end{cases} \quad (3)$$

Figure 7 shows the average entropy. If the entropy value of the flow $H(F_i)$ is larger than $H(F_{avg})$, we select it as a candidate flow, $C(F_i)$, for a DDoS attack. If $H(F_i)$ is less than or equal to $H(F_{avg})$, we classify it as a normal flow $N(F_i)$. As is evident, candidate flows have a higher probability of being DDoS attacks.

$$\begin{cases} Source\ port\ entropy\ of\ C(F_i) > Source\ port \\ entropy\ Threshold\ T_2, & Candidate\ Flow\ C(F_i) \\ Source\ port\ entropy\ of\ C(F_i) \le Source\ port \\ entropy\ Threshld\ T_2, & Normal\ Flow\ N(F_i) \end{cases} \quad (4)$$

The candidate flow $C(F_i)$ is used to calculate the entropy of the source port number. This entropy is then compared with the source port entropy threshold T_2. A higher entropy value means that a lot of ports are being used for transmission. A DDoS attack always involves the use of several ports to transmit a large number of packets. If the measured port entropy is higher than the entropy threshold (T_2), the corresponding traffic will be designated as a candidate flow $C(F_i)$ (see Equation (4) and Figure 8).

 ISBN: 978-1-61208-318-6

$$\begin{cases} Packet/second\ of\ C(F_i) > Packet/second\ threshold\ T_3,\ Attack\ Flow\ A(F_i) \\ Packet/second\ of\ C(F_i) \le Packet/second\ threshlod\ T_3,\ Normal\ Flow\ N(F_i) \end{cases} \quad (5)$$

In the final stage of the detection process, we calculate the rate of packet transmission (packets/sec) and compare it with the packet transmission threshold (T_3) to determine whether or not the corresponding traffic is part of a DDoS attack. If the packet transmission rate is higher than T_3, the flow in question will be classified as a DDoS attack. This process is consistent with Equation (5) and Figure 8.

IV. THE RESULT OF EXPERIMENT

In this section, we will evaluate the performance of our DDoS attack detection method. Our method is applied to a 'victim' router. We use OPNET [12] to simulate the network environment and evaluate our approach.

A. Experiment circumstance

We allowed web services and e-mail traffic as normal traffic on the network. In addition, we used attack traffic to simulate DDoS attacks. The topology of the experiment is a star, and uses 50 nodes, 1 server and 3 routers.

We allocate the nodes as follows: there are 25 nodes (node 1~25) which create the DDoS attack traffic and send it to the server, while 22 nodes (node 26~47) constitute normal traffic; three nodes (node 48~50) act as the server. We collect the traffic in the router using a 6-second time window. We also set appropriate thresholds for average entropy (T_1), the entropy of the source port (T_2) and packet transmission (T_3).

B. Experiment result and analysis

In this section, we use OPNET to evaluate the performance of the proposed method.

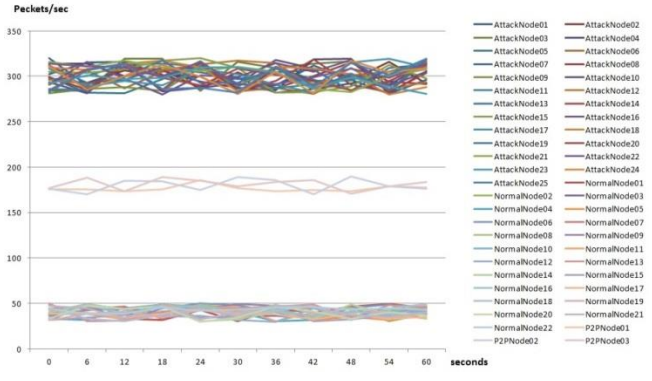

Figure 9. Creation rate each node packet

Figure 9 shows the relationship between packets transmitted (*y*-axis) and time (*x*-axis) for each node. Nodes 1 to 25, which simulated a DDoS attack, transmit approximately 300 packets per second, while nodes 26 to 47, used to imitate normal traffic, create around 40 packets per second. Nodes 48 to 50 -- our simulated P2P service -- create approximately 180 packets per second.

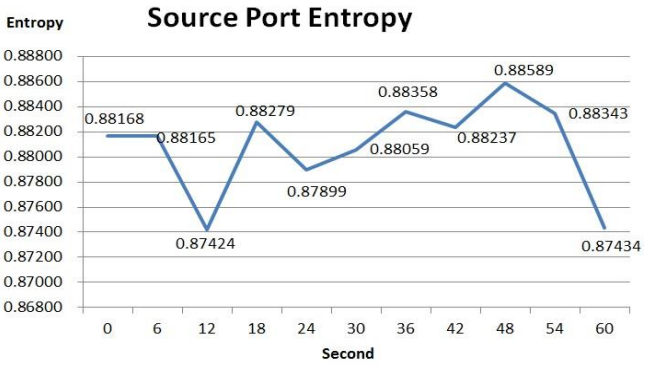

Figure 10. The entropy of source port number on candidate traffic

Figure 10 shows the entropy of the source port, which is used to determine whether traffic in question is a candidate for a DDoS attack. As we can see in Figure 10, the entropy of the source port is approximately 0.88, which is higher than the threshold value (T_2=0.8), as shown in Figure 10. Thus, it can thus be evaluated as candidate flow.

The last process involves checking the packets transmission rate of the candidate flow. As we can see in Figure 9, the rate for attack nodes is around 300 packets per second, far higher than the threshold (T_3=60). We can, thus, conclude that the flow is a DDoS attack.

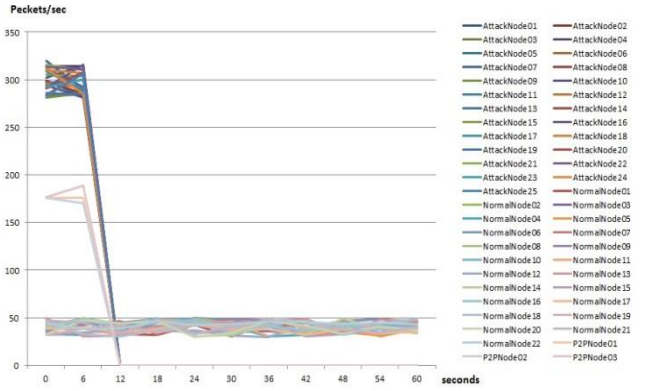

Figure 11. The previous method result

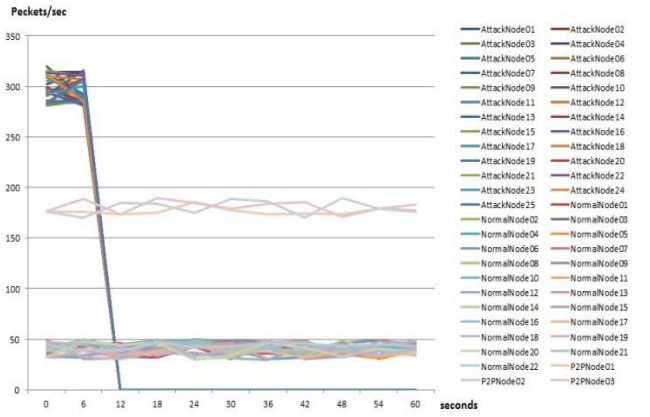

Figure 12. The Proposed method result

Figures 11 and 12 show the comparison between an existing DDoS attack detection method [8] and our proposed method. We can see that, unlike the existing ones, our proposed method can accurately detect DDoS attacks even in environments constituting small volumes of network traffic.

V. CONCLUSION

People gain much convenience from computer network because of the increasing services based on Internet. However, it is a "Double–edged Sword". It brings negative influence, such as Internet crime simultaneously. Every year, flooding attack causes a lot of economical loss.

In this paper, we proposed an effective DDoS attack detection method using flow entropy and packet sampling on a huge network. Once the DDoS attack is detected, we are able to control the attacker hosts. We have also demonstrated the superiority of our method to existing DDoS detection algorithms through experimental results.

REFERENCES

[1] G. Szabo, I. Szabo, and D. Orincsay, "Accurate Traffic Classification," IEEE Int. Symposium on World of Wireless Mobile and Multimedia Networks, 2007, pp. 1-8.

[2] Y. Xie and S. Z. Yu, "Monitoring the Application -Layer DDoS Attacks for Popular websites," IEEE/ACM Trans on Networking, vol. 17, no. 1, Feb. 2009, pp. 15-25.

[3] T. Peng, C. Leckie, and K. Ramamohanarao, "Survey of Network-based Defense Mechanisms Countering the DoS and DDoS Problems," ACM Computing Surveys, vol. 39, Iss. 1, Article 3, April 2007.

[4] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford, and N. Weaver, "The Spread of the Sapphire/Slammer Worm," http://www.caida.org/publications/papers/2003/sapphire/sapphire.html [retrieved; Dec 2013]

[5] S. Ranjan, R. Swaminathan, M. Uysal, A. Nucci, and E. Knightly, "DDoS-Shield: DDoS Resilient Scheduling to Counter Application Layer Atttacks," IEEE/ACM Transactions on Networking, vol. 7, no. 1, Feb. 2009, pp. 26-39.

[6] Arbor Networks ASERT Team, "July, 2009 South Korea and US DDoS Attacks," ARBOR Networks, July 2009.

[7] Y. Hong, S. Kwong, Y. Chang, and Q. Ren, "Unsupervised feature selection using clustering ensembles and population based incremental learning algorithm," ScienceDirect, vol. 41, no. 9, Sep. 2008, pp. 2742-2756.

[8] X. Kuai, Z. Zhi, and B. Suratol, "Profiling Internet Backbone Traffic Behavior Models and Application," ACM SIGCOMM, vol. 35, no. 4, Oct. 2005, pp. 169-180.

[9] T. Thapngam, S. Yu, and W. Zhou. " DDoS Discrimination by Linear Discriminant Analysis (LDA)," Computing, Networkng and Communications (ICNC) 2012, May 2012, pp. 532-536.

[10] R. Clausius, "The Mechanical Theory of Heat: With Its Applications to the Steam-engine and to the Physical Properties of Bodies," May 1867.

[11] Y. K. Park, "DDoS Attack Trend Analysis of 2012 year through the Cyber-Shelter," Internet and Security Focus, vol. 2, 2013.

[12] OPNET application and network performance, "http://www.opnet.com/"

 ISBN: 978-1-61208-318-6