Simulation and Detection of LDDoS Attacks using Queuing Algorithms

Kamal Preet Kaur
M.tech Scholar
SBSSTC, FZR
Punjab, India

Navdeep Kaur
Asst. Prof. (ECE Dept.)
SBSSTC, FZR
Punjab, India

Gurjeevan Singh
DIC –ECE
SBSSTC, PW, FZR
Punjab, India

ABSTRACT

This study aims at the evaluation of queuing algorithms using NS2 simulator. The recent LDDoS attacks cause more severe damage to the TCP based applications than the traditional DDoS attacks. The congestion participation rate (CPR) approach is used for detection and prevention of LDDoS attacks. Earlier approaches can only detect the LDDoS attacks. The CPR approach using queuing management algorithms shows better results than the DFT approach. The simulations are done using various parameters such as throughput, delay and bandwidth. Drop tail and red software are also compared using CPR approach; the better performance is given by RED approach using CPR.

General Terms

Comparison between normal TCP flow and LDDoS attack flow by using CPR approach using the three queuing management algorithms named REM, RED and DROPTAIL.

Keywords

DROPTAIL, RED, REM, NS2

1. INTRODUCTION

Today public internet is a worldwide collection of computer networks which are accessible by different ways. A lot of data is transferred and received through this network. The network is accessed using particular set of protocols or rules called TCP/IP (Transmission control protocol/Internet protocol). The development of internet has created new opportunities for personal interactions and business ventures. The cost of communication has also fallen many folds. The internet with all its advantages is not free from cyber criminalities and attacks to the data. The information on the internet can be lost, eavesdropped, manipulated or misused. The attacks can also corrupt computers. On the internet environment, the attacks can be easy, inexpensive and hard to detect, trace and prevent. It is difficult to ensure security goals which are confidentiality, integrity and availability. If there is no security of internet then the users would utilize the internet. The four main security issues are confidentiality, integrity, privacy, and availability. Now days it has become an important part of everyone’s life. As the network is growing day by day, security has become a big issue for internet users. Security of internet and local network has become a priority for computer problems. Network security demands the approval of access to data, which is controlled by the network administrator. Everyone on the internet wants security of data but they may not know that someone else is an intruder and collecting the information from the network. For securing the information some technique is required which assures the protection of the internet.

The use of internet is increasing tremendously day by day. The number of attacks on the internet is also increasing in the same manner. The most vulnerable attack to the internet at the present time is DoS attacks. They can cause the permanent cut-off of the system from the internet services. Till date there have been many techniques invented and implemented to fight these attacks. One of these techniques is active queue management algorithms, which prevent the congestion in the queue of data. [11]

2. ATTACKS

The threats to the internet which cause temporary or permanent unavailability of internet services are reffered to as attacks to the network. Some attacks which we have studied in our work are explained below.

2.1 DoS attack

The denial of Service (DOS) attack continues to be the major threat and hardest security problem to the network. This attack tends to make a user incapable of using the machine or service by making it unavailable to them. It is the major problem to the today’s internet. In DoS attacks there is no major benefit to the attacker except for the user’s pain. The DoS attacks can detach the network from the internet. Thus prevents the information exchange.[2][3]

2.2 DDoS attacks

This attack can have proportionally severe effect than the DoS attacks. The attacking traffic consists of very large number of packets compared to the victim’s resource, which can cause downfall in the victim’s service performance and even stop delivering any service. Since there are number of hosts so it is difficult to distinguish the attacking hosts and to take the required action against them. The DDoS attacks are launched in two phases; first the attacker finds the computers which are less secure and are called compromised hosts and prepares them for attack. Then the attacking hosts flood a tremendous amount of traffic towards the victims either under the command of attacker or automatically. The compromised computers are called zombies, bots or attacking hosts. [4]

2.3 LDDoS attacks

A new kind of DoS attack, which causes further more damage to the legit flows on the internet, traditional DDoS flows used sledge-hammer technique, but TCP-targeted LDDoS attack sends continuous packets which are very difficult to detect, if either these attacks are detected, it is very difficult to distinguish whether it is an attack or original data flow, as it sends periodically pulsing data flow, which may dramatically reduce the average rate of attack flows. [1]

3. CONGESTION PARTICIPATION RATE (CPR)

The CPR based approach is used to identify the TCP targeted LDDoS attacks. It is a novel metric approach which denies the fact that TCP flows avoids network congestion and LDDOS induce network congestion. It means that TCP will send fewer packets during network congestion and LDDOS will not reduce the number of packets during network congestion. The CPR based approach can effectively identify and prevent the LDDoS attack flows.

Experiments are done on NS2 simulator, the results shows that CPR based approach is better than the previously used DFT (Discrete Fourier Transform) approach as it effective for all LDDoS attacks rather than DFT which is effective for only a small set of LDDoS flows. [5]

4. ACTIVE QUEUE MANAGEMENT ALGORITHMS

The active queue management algorithms allows to manage the access to fixed amount of bandwidth by distinguishing which packet should be transferred and which one should be dropped when queue limit is fully occupied. There are many queue management algorithms which can be used for the balance between complexity, control and fairness. The main reason for the complexity is when more number of packets arrive then the capacity. The main motive of queue management algorithms is to minimize the congestion and provide the required bandwidth to the traffic. In our simulation we are using REM, RED and DROPTAIL. [5]

4.1 RED

Random Early Detection works by randomly (based on certain probability) discarding packets at the nodes of the network, before the occurrence of congestion, when the average queue length exceeds the predefined minimum threshold. When the average queue length exceeds the maximum threshold, the probability of rejection becomes equal to 1. RED monitors the average length of the queues by discarding or ECN-marking packets based on statistical probability. If the buffer is nearly vacant, all incoming packets are received. As there is increase in use, the probability of discarding recently arrived packet also increases. When the buffer is occupied, all incoming packets are deleted. RED has no QoS differentiation in the basic version. The versions WRED (Weighted RED) and RIO (RED with In and Out), which consider the QoS into account. [4][5]

4.2 DROPTAIL

Drop Tail is a simple queue management algorithm: it sets a predefined value for the maximum length of the queue and when this value is reached, new packets are discarded, until the next vacant buffer space to accept new packets .When using the Drop Tail mechanism, all the packets in the traffic are treated identically, regardless of the type of traffic which it belongs to. Packet loss will cause the transmitter to reduce the number of TCP packets sent before receiving the acknowledgment. The throughput of the given TCP session will then reduce, until the transmitter start again to receive acknowledgments and begin increasing the size of its congestion window. [4][5]

4.3 REM

REM differs from RED only in the first two design questions; it uses a different definition of congestion measure and a different marking probability function. The first design of REM is to stabilize both the queue around a small target and the input rate around link capacity, regardless of the number of users sharing the link. Each productivity queue that implements REM maintains a variable which is called ‘price’ as a congestion evaluation measure. The second idea of REM is to use the addition of the link prices along a path as a measure of congestion in the path, and to implant it into the end-to-end marking probability that can be observed at the source. [9]

5. SIMULATION SETUP

Dumbbell network topologies are commonly used in congestion control studies. Network topology consists of two routers (R0, R1, 30 users (User1-----User30), 20 attackers (Attacker1------Attacker20), 30 servers (Server1-----Server30), and a victim server (Victim Server). The link between two routers is the bottleneck link with a bandwidth of 5 Mbps and one-way propagation delay of 6 ms. All the other links have a bandwidth of 10Mbps and a one-way propagation delay of 2 ms. In this topology, User i communicates with Server i (i = 1------30) using FTP, and 20 attackers send UDP packets to attack the Victim Server. The queue size of the bottleneck link is 50. A RED based on packet count is deployed at router R0 on the queues of the bottleneck link. Other links use Drop Tail queues. A CPR-based detection module is installed at router R0 where most normal TCP packets are dropped when an LDDoS attack is present. For comparison, we also install a module based on Cumulative Amplitude Spectrum (CAS) at R0; CAS uses Discrete Fourier Transform (DFT) to locate disturbances caused by LDDoS flows. Simulation time period is 240s and the LDDoS traffic begins at 120s and ends at 220s. And the frequency is 1000 Hz.

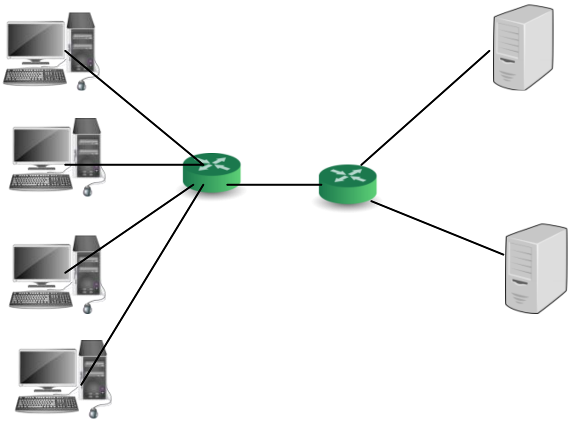

Fig 1: Network topology of experiment

6. RESULTS AND DISCUSSIONS

In this section we have discussed results, the results are divided into two scenarios on the basis of CPR approach used and the LDDoS attack flow. The two respective scenarios are discussed below.

6.1 Scenario 1: The comparison is done between normal and CPR approach for the three algorithms.

6.1.1 Number of packet sent

The total number of packets successfully sent from the source [4].

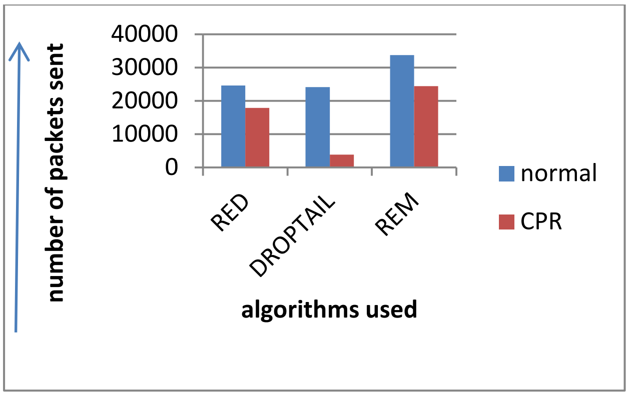

Fig 2: Number of packets sent for normal and CPR under REM, RED and DROPTAIL

In the figure 2, it is clear from the chart that number of sent packets for normal REM, RED and DROPTAIL is better than algorithms using CPR approach, and if we compare REM, RED and DROPTAIL individually, REM is better algorithm in case of number of sent packets in all of them. Further by using CPR approach if we compare REM, RED and DROPTAIL, the performance of REM is better than both RED and DROPTAIL. So, in case of number of sent packets REM is better in either way than RED or DROPTAIL.

6.1.2 Number of packets received

The total number of packets successfully received at the destination node[4].

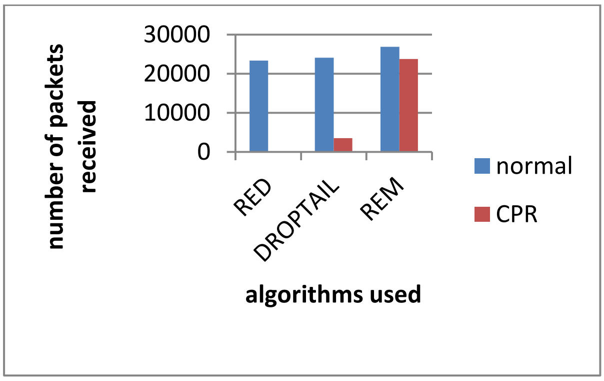

Fig 3: Number of packets received for normal and CPR under RED and DROPTAIL.

In the figure 3, the number of received packets by using algorithms REM, RED and DROPTAIL as well as REM, RED and DROPTAIL using CPR approach is shown. For normal REM, RED and DROPTAIL, number of packets received using RED is less as compared to number of packets received using REM and DROPTAIL, as well as in other case using CPR also, RED drops more number of packets than REM and DROPTAIL, so the number of packets received using RED is less as compared to other two algorithms. In both the cases RED drops more number of packets than REM and DROPTAIL. More packets are received using REM. If we compare individually number of packets received in RED is more when using normally than using CPR. In case of DROPTAIL, number of packets received when using normally is more than in case of using CPR. Similarly, same is the case with REM but with very small difference.

6.1.3 Number of packets lost

The total number of packets which are sent by the source, but are not received at the destination node [4].

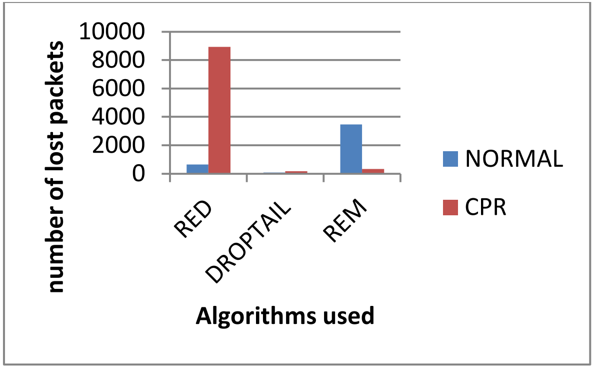

Fig 4: Number of lost packets for normal and CPR under RED and DROPTAIL

In the figure 4, number of packets lost by using REM, RED and DROPTAIL, as well REM, RED and DROPTAIL with CPR approach are compared. As shown in the chart, number of packets lost in RED is more compared to REM and DROPTAIL, similarly using CPR, number packets lost using RED is large as compared to REM and DROPTAIL using CPR approach. If we compare normal and CPR approach of each algorithm, in case of RED, number of packets lost using CPR is very large as compared to normal approach. In case of DROPTAIL very small amount of packets are lost, and in case of REM number of packets lost using normal approach is more an using CPR approach.

6.2 Scenario 2: the comparisons are done between normal TCP flow and LDDoS attack flow using RED and REM

In this section we present our experiment results obtained from ns-2. It tests the influence of the RED and REM mechanism on the approach. The performance evaluation of the CPR-based approach is done in the presence of different LDDoS attacks. [1]

The two LDDoS attacks used in our experiment are:

AFI (Attack Frequency Intensification)

The first category represents the LDDoS attacks whose aggregate attack period is equally distributed among n flows. The attack frequency of the aggregate flow is intensified by n times, compared to the frequency of each attack flow. [1]

AWI (Attack burst Width Intensification)

The second category corresponds to the case when the aggregate burst width of an LDDoS attack is equally distributed among n flows. An attack burst of a flow is immediately followed by a burst from another flow. In this case, the attack burst width of the aggregate attack flow is intensified by n times. [1]

6.2.1 AFI Attack experiment for average CPR in RED

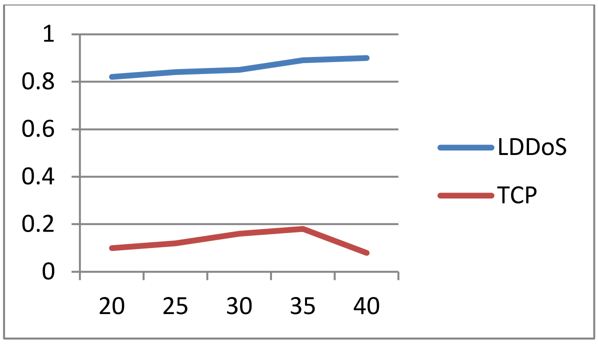

Fig 5: Average CPR for normal TCP flow and LDDoS using RED in AFI attack

In this simulation result the comparison is done between CPR approach for RED algorithm for TCP flows and LDDoS attack flow in AFI attacks. The time duration for both the simulations is same; the average CPR for LDDoS attack is more than that of TCP flow.

6.2.2 AFI attack experiment for average CPR in REM

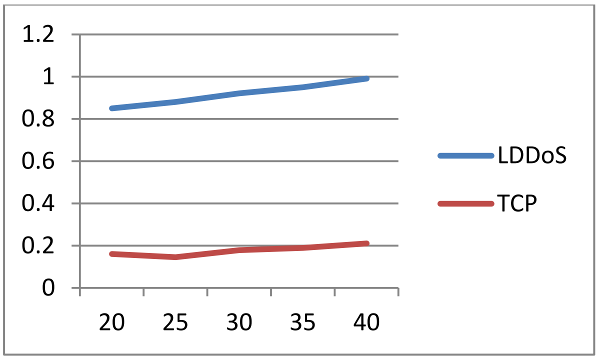

Fig 6: Average CPR for LDDoS attack flow and TCP flow using REM in AFI attacks

In this simulation result the comparison is done between CPR approach for the LDDoS attack flows in AFI attacks is more than that of TCP flow. The time duration for both the simulations is same.

6.2.3 AFI attack experiment for average CAS in RED

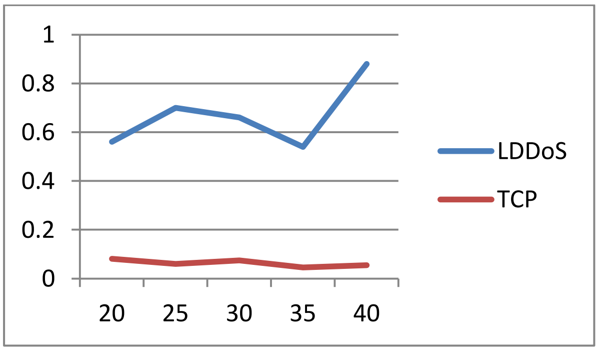

Fig 7: using CAS in RED for TCP flow and LDDoS attack flow in AFI attacks

In this simulation result the comparison is done between CAS approach for RED for TCP flows and LDDoS attack flow in AFI attacks. The time duration for both the simulations is same, the average CAS for LDDoS increases with time as compared to TCP flow.

6.2.4 AFI attack experiment for average CAS in REM

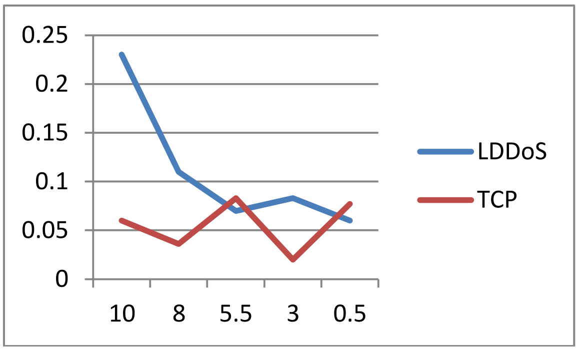

Fig 8: using CAS in REM for TCP flow and LDDoS attack flow in.

In this simulation result the comparison is done between CAS approach for RED and REM for LDDoS attack flows in AFI attacks. The time duration for both the simulations is same; the average CAS for RED is more than that of REM.

6.2.5 AWI attack experiment for average CPR in RED

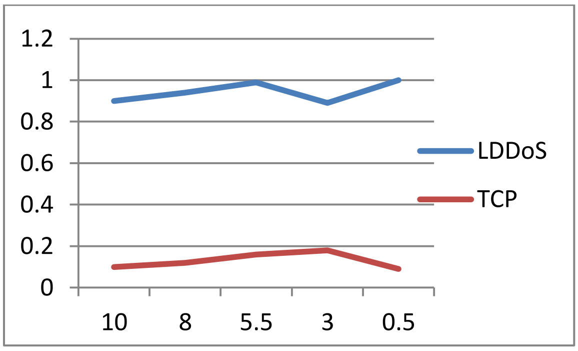

Fig 9: average CPR in RED for LDDoS attack flow and TCP flow for AWI attacks.

In this simulation result the comparison is done between CPR approach for RED for TCP flows and LDDoS attack flow in AWI attacks. The time duration for both the simulations is same; the average CPR for LDDoS attack flow is more than that of TCP flow.

6.2.6 AWI attack experiment for average CPR in REM

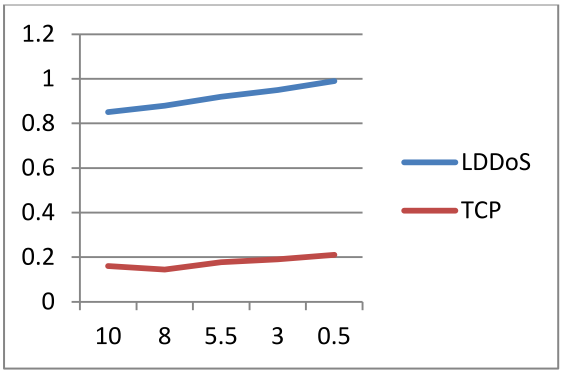

Fig 10: average CPR for REM in LDDoS flow and TCP flow for AWI attacks.

In this simulation result the comparison is done between CPR approach for REM in LDDoS attack flows and TCP flow in AWI attacks. The time duration for both the simulations is same; the average CPR for LDDoS attack flow is more than that of TCP flow.

6.2.7 AWI attack experiment for average CAS in RED

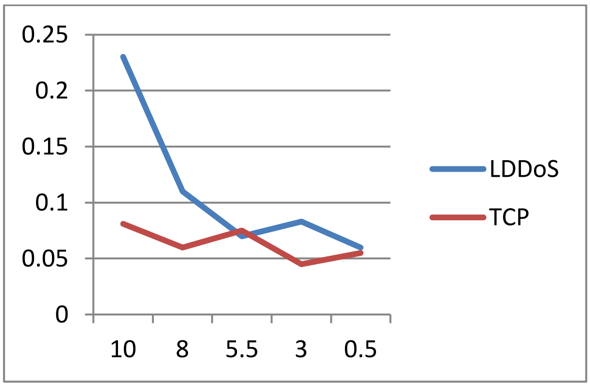

Fig 11: average CAS in TCP flow and LDDoS flow using RED in AWI attacks.

In this simulation result the comparison is done between CAS approach for RED for TCP flows and LDDoS attack flow in AWI attacks. The time duration for both the simulations is same; the average CAS for LDDoS attack flow is more than that of TCP flow.

6.2.8 AWI attack experiment for average CAS in REM

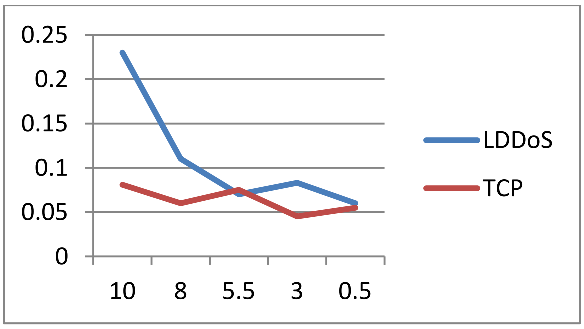

Fig 12: average CAS in LDDoS attack flow and TCP flow using REM in AWI attacks

In this simulation result the comparison is done between CAS approach using REM for the LDDoS flows and TCP flow in AWI attacks. The time duration for both the simulations is same; the average CAS for LDDoS attack flow is more than that of TCP flow.

7. CONCLUSION

As per the comparison between the algorithms REM, RED and DROPTAIL, the REM approach shows better results when used independently, but the value of packets lost and dropped is very large as compared to REM and DROPTAIL, in case when using independently as well as with CPR approach packets received is much more while using REM and DROPTAIL either independently or with CPR approach, number of packets sent is shown better in case of REM and RED rather than in DROPTAIL. These results will be beneficial for us in our further research work of the comparison of more queuing algorithms. In case of simulations in attacks, REM shows better average CPR value than RED, and in case of CAS approach there is mixed response, it is not clear which one is better, but as the time increase the average CAS for REM increase than average CAS in RED. In the future work, more number of algorithms will be compared to get the better knowledge of the queue management algorithms; the number of paremeters used will also be increased.

8. REFERENCES

[1] Zhang, C., Cai, Z., Chen,W., Luo, X., and Yin, J. 2012. Flow level detection and filtering of low-rate DDoS.

[2] Bhuyan, M.H., Kashyap. J., Bhattacharyya, D.K., and Kalita, J.K. .Detecting Distributed Denial of Service Attacks: Methods, Tools and Future Directions.

[3] Douligeris, C. and Mitrokotsa, A. 2004.DDoS attacks and defense mechanisms: classification and state-of-the-art.

[4] Agrawal, M., Tiwari, N., Chaurasia, L. A. and Saraf, J. 2009. Performance Analysis and QoS Assessment of Queues over Multi-Hop Networks.

[5] Kumar, S., Bhandari, A., Sangal, A.L. and Saluja, K.K. 2011. Queuing Algorithms Performance against Buffer Size and Attack Intensities.

[6] Afrasiabi, S. and Abazari, F. 2013. The evaluation of the behavior of computer networks by NS simulator and the effect of queuing systems in the performance of especial networks.

[7] Xiao, B., Chen, W. and He, Y. A novel approach to detecting DDoS attacks at an early stage.

[8] Hu, N., Ren, L. and Chang, J. Evaluation of Queue Management Algorithms.

[9] Athuraliya, S., Low, S.H., Li, V.H. and Yin, Q. 2001. REM: Active Queue Management.

[10] Zhi-jun, W., Hai-tao, Z., Ming-hua, W. and Bao-song, P. 2012. MSABMS-based approach of detecting LDoS attack.

[11] Ghansela, S. 2013. Network Security: Attacks, Tools and Technique.

[12] Lee, K., Kim, J., Kwon, K.H., Han, Y. and Kim, S. 2008. DDoS attack detection method using cluster analysis.