Network Layer DDoS Mitigation Model Using Hidden Semi-Markov Model

L. Kavisankar, C. Chellappan, and R. Vaishnavi

***Abstract*—Distributed Denial of Service (DDoS) remains a serious problem in cyber security. Some recent DDoS incidents show that such attacks continue to cause serious threats to the Internet. It does not allow the legitimate users to access the resources provided by the servers. With the growth in technology, the DDoS attackers have improved their sophistication, by automating the attacks. The attackers exploit the protocol vulnerabilities to create these kinds of DDoS attacks. The detection of DDoS attack is complicated, since they mix with the legitimate packet traffic. Later separation of DDoS attack packets from legitimate packet is highly difficult, since false DDoS alarm may lead to blocking a legitimate packet. The rate of arrival of the packets is very high in the case of DDoS attack; it's the same in the case of the flash crowd. This makes the detection of DDoS even more difficult. The proposed model uses the Hidden Semi-Markov model (HSMM) which is an extension of the Hidden Markov model (HMM) deals with explicit state duration. In this model using HSMM observations are performed in milliseconds for the analysis of network traffic flow packets, this result in optimal detection and mitigation of DDoS attack.**

***Index Terms*—DDoS, flooding attack, TCP SYN flooding, HSMM, TCP retransmission, stochastic finite state machine.**

I. Introduction

Every layer of communication has its own unique security challenges. The concentration of this work is on the transport Layer (Layer 4 in the OSI model) which is vulnerable for the Denial of Service (DoS) attack or Distributed Denial of Service (DDoS) attack. Distributed denial of service attack (DDoS) is one of the most common network attacks. Denial of service attack refers to a devastating attack, which blocks or denies legitimate users' access to a server. It uses mass packet data beyond processing capabilities of the target, consuming the available system resources, bandwidth resources, resulting in paralysis of network services. Any action, which can stop the legitimate users from service and cannot engage in normal behavior of network services, can be called a denial of service attack. Two most popular protocols used in the transport layer are TCP (Transmission Control Protocol) and UDP (User Datagram Protocol). One of the key security risks at the Transport Layer associated with TCP is TCP SYN attack.

A SYN flood is a form of denial-of-service attack in which an attacker sends a succession of SYN requests to a target's system in an attempt to consume enough server resources to make the system unresponsive to legitimate traffic. Normally when a client attempts to start a TCP connection to a server, the client and server exchange a series of messages which normally has the following state transitions:

The client requests a connection by sending a SYN (synchronize) message to the server which acknowledges with SYN+ACK flag packet. After receiving SYN+ACK from the server, if the client does not respond with ACK for connection establishment, the SYN packets get accumulated in the server side with number of packets from the client. This phenomenon is known to be SYN flooding attack. This chaos may be a result of network congestion or flash crowd. In this case the risk is minimal which can be solved by balancing the load in the network. In contrary, if SYN flooding occurs due to spoofing or falsified IP address it is a critical issue [1]. Some systems may also malfunction badly or even crash if other operating system functions are starved because of resources in this way. TCP "SYN" attack is also known as SYN Flooding. It takes advantage of these flaws in the design and implementation of TCP three-way handshake. The host's listen queue is quickly filled up, and it will stop accepting new connections, until a partially opened connection in the queue is completed or times out. This inability of removing a host from the network for at least 75 seconds can be used for denial-of-service attack.

Both end-host and network-based solutions to the SYN flooding attack have merits. Both types of defense are frequently employed, and they generally do not interfere when used in combination [2]. Because SYN flooding targets end hosts rather than attempting to exhaust the network capacity, it seems logical that all end hosts should implement defenses and those network-based techniques is an optional second line of defense that a site can employ. End-host mechanisms are present in current versions of most common operating systems. Some implement SYN caches, others use SYN cookies after a threshold of backlog usage is crossed, still others adapt the YN-RECEIVED imer and number of retransmission attempts for YN-ACKs. Since, some techniques are known to be ineffective they increase the backlogs and reduce the timer of SYN-RECEIVED; these techniques should definitely not be relied upon. Based on experimentation and analysis, a SYN cache seems like the best, end-host mechanism available. This choice is motivated by the facts that they are capable of withstanding heavy attacks, they are free from the negative effects of SYN cookies, and they do not need any heuristics for threshold setting as in many hybrid approaches.

Manuscript received May 6, 2013; revised August 10, 2013. The experimental work was carried out at SSE Lab Anna University Chennai, established with funding supported by the NTRO Government of India for collaborative project on "Collaborative Directed Basic Research on Smart and Secure Environment" and this paper was modeled with the help of this project. Authors would like to thank the project coordinators and the NTRO officials.

L. Kavisankar, C. Chellappan, and R. Vaishnavi are with the Department of Computer Science and Engineering, Anna University, Chennai 600025, India (e-mail: kavisankaar@gmail.com, drcc@annauniv.edu, vaishnavigpl35@gmail.com).

HSMM is applied to describe the node behavior. It has been successfully applied in many fields, such as machine recognition, sequences clustering, mobility tracking in wireless networks, and inference for structured video sequences. The most notable characteristic of HSMM is that it can be used to describe most physical signals without many hypotheses. Furthermore, the non-stationary and the Non-Markovian properties of HSMM can best describe the self-similarity or long-range dependence of network traffic that has been proved by vast observations on the network traffic.

After the initiation of a TCP/IP connection by a SYN packet, a SYN flood if occurred, consumes all available slots in a server's TCP connections table and by doing so, prevents other users from establishing new TCP/IP connections. The main aim is to detect the SYN flood attack. Parameters such as overall traffic, arrival time and frequency are calculated**,** threshold value is fed to each normal nodes in this module by HSMM. The SYN flood attack is then implemented, network performance is analyzed to find or detect attacker. The flash crowd can also be discriminated using this HSMM [3]. HSMM uses ower spectral density to detect attacker where normal traffic is compared with attack traffic in run time.

The proposed approach examines the details of a SYN flood attack and describes the methods of detecting such attacks. Both the high rate and low rate flooding attack are detected by the proposed model. The model predicts the attack using the observations and the sequence of observations. The attack predictions are updated regularly by calculating the Bayesian update using the probability value of each observation. Based on the mean deviation of the entropy value, abnormality is noticed and defended from attacking packets. The method also mitigates the low rate attack by randomizing the RTO value.

II. Existing Methods

Distributed Denial-of-Service (DDoS) attacks are large-scale cooperative attacks typically launched from a large number of compromised hosts. DDoS attacks are bringing growing threats to businesses around the world. Many methods have been proposed to counter such attacks; they are not efficient enough. Some recent DDoS incidents show that such attacks continue to cause serious threats to the Internet.

The challenge of detecting App-DDoS attacks is due to the following aspects as utilizing high layer protocols to pass through most of the current anomaly detection systems designed for low layers and arrive at the victim web server. Flooding is not the unique way for the App-DDoS attack. There are many other forms, such as consuming the resources of the server (CPU, memory, hard disk, or database), arranging malicious traffic to mimic the average request rate of the normal users or utilizing the large-scale botnet to produce the low rate attack flows [4]. This makes detection more difficult.

An extended Hidden Semi-Markov Model was proposed to describe browsing behaviors of web surfers. In order to reduce the computational overhead introduced by the model's large state space, the algorithm is derived for the online implementation of the model based on the M-algorithm [4]. Entropy of the user's HTTP request sequence fitting to the model is used as a criterion to measure the user's normality. In the proposed work, Hidden Semi Markov Model is used to describe the behavior of the nodes during TCP connections. The parameters such as arrival rate, numbers of bytes received, service rate, response rate are calculated and compared with normal nodes using HSMM.

A trust based approach whereby the peer nodes interact amongst themselves to find the trustworthiness of other participating nodes in the neighborhood based on their reply [5]. Further the packets originating from a node are monitored and using the features of the packet header, a centroid is found for each node using clustering technique and the result of this is combined with the trust.

The detection mechanism is based on the protocol behavior of TCP SYN–FIN (RST) pairs, and is an instance of the Sequential Change Point Detection. To make the detection mechanism insensitive to site and access pattern, a non-parametric Cumulative Sum (CUSUM) method [6]-[7] is applied.

The detection schemes for SYN Flooding attacks have been classified broadly into three categories of detection schemes.

1) Router data structure based
2) Statistical analysis of the packet flow based
3) Artificial intelligence based

The application layer resource attacks are characterized as request flooding, asymmetric, repeated one-shot, on the basis of the application workload parameters that they exploit. To protect servers from these attacks, a counter-mechanism that consists of a suspicious assignment mechanism and a DDoS-resilient scheduler, DDoS Shield was proposed [8].

The methods like Dynamic Detection, PAD and MAD Models, HAWK, At EDGE Router, Self- Similarity methods [9] suffer from processing and memory overhead which is over come by the RTO randomization by just modifying the congestion control mechanism. Most of the operating systems in use today have a common base TCP Retransmission Timeout (RTO) with RTO randomization; an attacker cannot predict the next TCP timeout and consequently cannot inject the burst at the exact instant [10]. Randomization on minRTO shifts and smoothes TCP's null frequencies and could mitigate the impact of attacks, but the fundamental tradeoff between TCP performance and vulnerability to low rate DoS attacks remains.

III. Methodology

The detection and defence of high rate flooding and low rate flooding are achieved in; the proposed model. It works in three parts given in Fig. 1:

1) Attack detection using HSMM
2) RTO Randomization to mitigate low rate attack
3) Blocking flood attacks

A. Attack Detection Using HSMM

In this part, to defend against DDoS attacks, the statistics of packet attributes in the headers of IP packets, such as IP address, time-to-live (TTL), protocol type, etc., are considered as some traffic characteristics that inherently distinguish the normal packets from the attack ones. We use the parameters

such as arrival rate, service rate, and response time rate to calculate and compare with normal nodes using HSMM. Therefore, "abnormal" traffic can be detected based on these traffic characteristics during a DDoS attack.

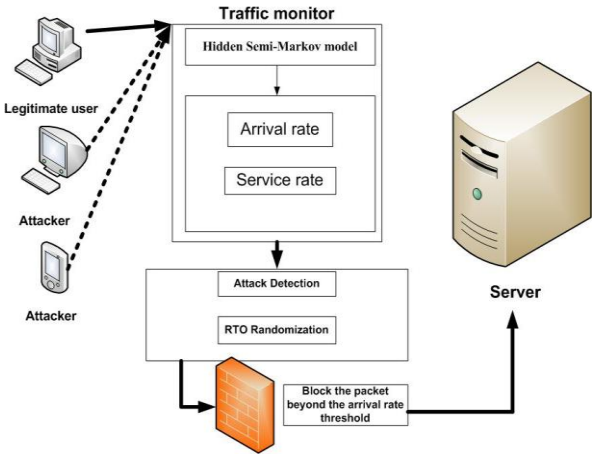

Fig. 1. Proposed architectural model

Hidden Semi-Markov Model is proposed to describe the network behavior while establishing the connection with the server using the TCP three way handshake. HSMM is an extension of the Hidden Markov Model (HMM) with explicit state duration. It is a stochastic finite state machine, specified by

$$(S, \pi, A, P) \tag{1}$$

where:

S is a discrete set of hidden states with cardinality N, i.e., $S= \{1,\ldots., N\}$

π is the probability distribution for the initial state

$$\pi_x \equiv \Pr [s_1=\mathbf{x}], s_t \tag{2}$$

The above denotes the state that the system takes at time t and $x \in s$. The initial state probability distribution satisfies

$$\Sigma_x \pi_x = 1 \tag{3}$$

A is the state transition matrix with probabilities:

$$a_{xy} \equiv \Pr [s_t = y|s_{t-1} = \mathrm{x}], x, \mathrm{y} \in \mathrm{s} \tag{4}$$

The state transition coefficients satisfy

$$\Sigma_y a_{xy} \tag{5}$$

P is the state duration matrix with probabilities:

$$p_x(d) \equiv Pr[r_t = d \,|s_t = m], T_t \tag{6}$$

The above denote the remaining time of the current state

$$s_t, x \in s, d \in \{1,\ldots.,D\} \tag{7}$$

D is the maximum interval between any two consecutive state transitions, and the state duration coefficients satisfy

$$\Sigma_d p_x(d) = 1 \tag{8}$$

Then, if the pair process (s_t, T_t) takes on value (x, d), the semi-Markov chain will remain in the current state x until time $t+d-1$ and transits to another state at time $t+d$, where $d \geq 1$. The observations are represented by y and the unknown states are represented by x. The predication of the DDoS attack is done using the HSMM by observing the three flags SYN, SYN+ACK, ACK of the TCP protocol in the packet. The unknown states are legitimate and attack nodes.

The set of HSMM parameters λ consists of initial state distributions, the state transition probabilities, the output probabilities and state duration probabilities. For short notation, they can be denoted as

$$\lambda = (\{\pi_x\},\{ a_{xy}\},\{b_x(k)\},\{ p_x(d)\}) \tag{9}$$

The sequential observation plays the major role in the prediction of the attack. The prediction depends on the previous state and observation. For e.g. a node may send the attack packet for a period of time, as well as in the mere future and would be the same in the case of legitimate behavior of the node. The observation of behavior over a period is the key for prediction.

The observations $O=(o_1,\ldots.,o_t)$ where o_t denotes the observable output at time t and T is the number of samples in the observed sequences.

For every state an output distribution is given as

$$b_x(k) \equiv Pr[o_t = k| \, s_t = x] \tag{10}$$

where $x \in s$ and $k \in V = \{1, \ldots., K\}$. K is the size of the observable output set. The legitimate or attack node are updated based on the observed sequence. Each observation has its probability value, which are being updated from its initial state.

The output probabilities satisfy when the following conditional independence of output is assumed,

$$\Sigma\, b_x(k) = 1.b_x\,(o_{a|b}) = \prod_{t=a}^{b} \mathrm{b_x(o_t)} \tag{11}$$

where $o_{a|b} = \{o_{t:a\leq\, t\, \leq\, b}\}$ represents the observation sequence from time a to time b. The prediction and update detects the attack packet using the HSMM. Entropy of the node TCP flag sequence fitting to the model is used as a criterion to measure the nodes normality.

B. RTO Randomization to Mtigate Low Rate Attack

In RTO randomization the low rate DoS attack can be detected. The low rate DoS attack exploits the fact that most systems have RTO as 1sec (standard value). Therefore if the RTO is set to any random values it would be difficult for the attacker to find out the next RTO value and this ultimately would help in controlling the rate of attack. In such a way RTO randomization along with proper flow monitoring helps to detect the problems with the packet and back tracking will lead us to the attacker source.

C. Blocking Flood Attacks

The blocking of attack packets is done using the detection of both the HSMM and RTO randomization. High rate of arriving attack packets are detected using HSMM and RTO randomization detects the low rate of arriving attack packets. Based on the behavior of the node, the entropy is calculated and is used for the blocking of attack packets.

IV. Performance Measure

A. Experimental Setup

The experiment was done in Linux OS using Network Simulator 2 (NS-2.34). In Front end for Tool Command Language (version 8.4.1) is used. For the back end C++ is used and the Interface used is Network Animator Tool (Xgraph-12.1).

B. Results and Discussion

The behavior of each node is analyzed using HSMM. The parameters such as arrival rate, service rate, response time rate are calculated and compared with normal nodes using this model.

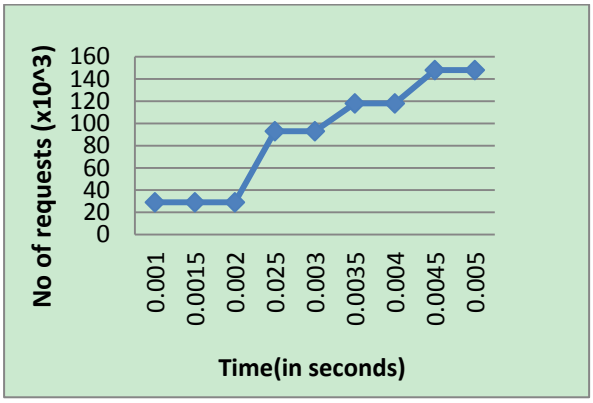

Fig. 2. Continuous attack from the attacker to the victim server.

The rate of arrival of the attack packet to a victim server increases with time. The constant increase of attack packets is due to the continuous flooding from the attacking nodes as seen in the Fig. 2.

The analysis is made in milliseconds so that the detection using the HSMM gives improved detection efficiency compared to the existing works in this field.

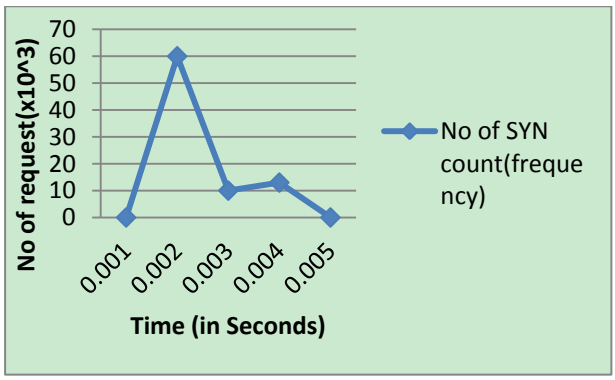

Fig. 3. Detection of continuous DDoS attack using HSMM.

Fig. 3 shows the continuous DDoS attack detection using HSMM; the drastic increase in the SYN request count in a given period of time close to 60,000 SYN requests is found. Based on this HSMM detection, packets from the black listed IP addresses are blocked.

TCP three way handshake protocol has three major flags such as SYN, SYN+ACK, ACK for connection establishment are taken as observations in the HSMM. With the unknown state of attack, legitimate nodes are predicted by the behavior of their observations. The prediction and update are done constantly to find the unknown state of the node. Fig. 4 gives the clear indication of SYN flooding attack occurrence, since there is SYN+ACK packets close to 12,00,000 packets minimum to maximum of 13,50,000 packets. This gives us the clear indication that there are number of half open connection in the server based on this detection of the attack packets can be found. Then the blocking of attack packets from attacking nodes takes place.

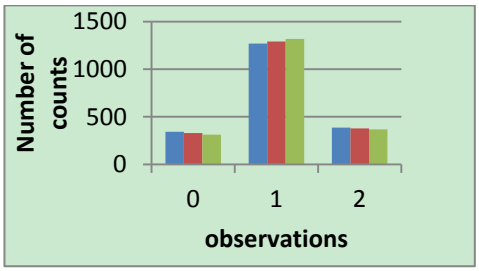

Fig. 4. TCP Three way handsake observation frequencies.

Comparison of graphs in Fig. 2 and Fig. 5 give us a clear indication that, RTO randomization reduces the effect of the low rate DDoS attack which happened, using the continuous and increasing rate of arrival of the attack packet.

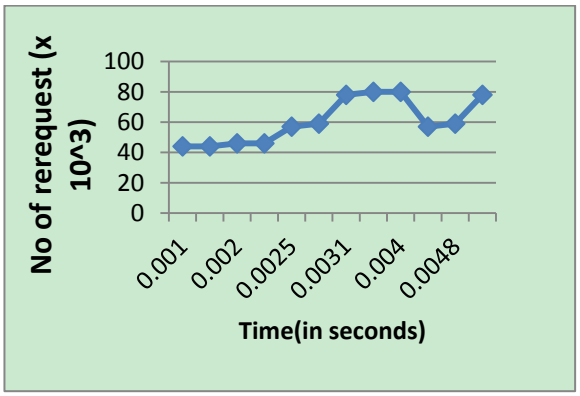

Fig. 5. Reduction in low rate attack packet using RTO randomization.

It can be found that nearly half of the request is bogus and it is mitigated using the RTO randomization.

V. Conclusion and Future Work

Every layer of communication has its own unique security challenges. The concentration of this work is on network and transport layer (layer 3 & 4 in the OSI model) that is vulnerabilities of TCP protocol and IP contributes to Denial of Service (DOS) attack or Distributed Denial of Service (DDOS) attack.

The HSMM detection is accurate, since it detects the arrival rate of the packets in milliseconds. Parameters like arrival rate, service rate are taken for the detection of DDoS attack. Based on the detection, IP addresses which goes beyond the threshold limit is blocked. The RTO randomization is used to mitigate the low rate attack to a greater extend.

Future work intends to improve the work on avoidance of attacker packets by using its variation in threshold level using Trust.

Acknowledgment

The researchers would like to thank the NTRO sponsored Collaborative Directed Basic Research on Smart and Secure Environment project lab for providing experimental setup and UGC for financial support as fellowship.

REFERENCES

[1] W. Chen and D. Y. Yeung, "Defending against TCP SYN flooding attacks under different types of IP spoofing networking," in *Proc. International Conference on Systems and International Conference on Mobile Communications and Learning Technologies*, 2006.

[2] N. A. Noureldien and M. O. Hussein, "Block spoofed packets at source (BSPS): A method for detecting and preventing all types of spoofed source IP packets and SYN flooding packets at source: A theoretical framework," *International Journal of Networks and Communications*, vol.2, no. 3, pp. 33-37, 2012.

[3] S. Yu, W. l. Zhou, W. J. Jia, S. Guo, Y. Xiang, and. F. L Tang, "Discriminating DDoS attacks from flash crowds using flow correlation coefficient," *IEEE/ACM Transactions on Parallel and Distributed System*, vol. 23, no. 6, pp. 1073-1080, June 2012.

[4] X. Yi and S. Z. Yu. "A large-scale hidden semi-Markov model for anomaly detection on user browsing behaviors," *IEEE/ACM Transactions on Networking*, vol. 17, pp. 54-65, 2009.

[5] J. Padamanabhan, K. S. Easwarakumar, B. Gokul and S. Harishankar , "Trust based traffic monitoring approach for preventing denial of service attacks," *ACM SIN'09*, pp. 200-206**,** October 2009.

[6] M. E. Manna and A. Amphawan, "Review of SYN-flooding attack detection mechanism," *International Journal of Distributed and Parallel Systems (IJDPS)*, vol. 3, no. 1, January 2012.

[7] H. N. Wang, D. L. Zhang, and K. G. Shin, "Detecting SYN flooding attacks," in *Proc. INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, IEEE*, vol.3, June 2002, pp. 1530-1539.

[8] S. Ranjan, R. Swaminathan, M. Uysal, and A. Nucci, "DDoS-Shield: DDoS-Resilient scheduling to counter application layer attack," *IEEE/ACM Transactions on Networking*, vol. 17, no. 1, pp. 26-39, February 2009.

[9] R. Mathew and V. Katkar, "Survey of low rate DoS attack detection mechanisms," in *Proc. ACM of the International Conference & Workshop on Emerging Trends in Technology ICWET'11*, pp. 955-958, 2011.

[10] Y. Guang, M. Gerla, and M. Y. Sanadidi, "Defense against low-rate TCP-targeted denial-of-service attacks," in *Proc. IEEE Ninth International Symposium on Computers and Communications*, vol. 1, 2004.

L. Kavisankar is a PhD student in the Department of Computer Science and Engineering at Anna University, Chennai, India. He received his B.E Computer Science and Engineering from Easwari Engineering College under Anna University in 2007 and M.E Computer Science and Engineering from SSN College of Engineering Under Anna University in 2009, Chennai, India. His current research is on mobile IPv6 and network security.

C. Chellappan is a professor in the Department of Computer Science and Engineering at Anna University, Chennai, India. He received his B.Sc. in Applied Sciences and M.Sc in Applied Science–Applied Mathematics from PSG College of Technology, Coimbatore under University of Madras in 1972 and 1977. He received his M.E and Ph.D in Computer Science and Engineering from Anna University in 1982 and 1987. He is currently the Dean of College of Engineering Guindy, Anna University, Chennai. He was the Director of Ramanujan Computing Centre (RCC) for 3 years at Anna University (2002–2005). He has published more than 70 papers in reputed International Journals and Conferences. His research areas are computer networks, Distributed /mobile computing and soft computing, software agent, object oriented design and network security.

R. Vaishnavi is a M.E student in the Department of Computer Science and Engineering at Anna University, Chennai, India. She received her B.E Computer Science and Engineering from Srinivasa Institute of engineering and Technology under Anna University in 2011. Her current research is on DDoS and network security.