ISSN 2047-3338

Analysis & Study of Application Layer Distributed Denial of Service Attacks for Popular Websites

V. Venkata Ramana[1], P. Shilpa Choudary[2] and Maya B. Dhone[3]
[1]Department of CSE, JNTUH, India
[2,3]Gurunank Engineering College, India

***Abstract*– Application Layer is the most important for development lifecycle, Application layer Distributed denial of service (DDoS) attack is a continuous threat to the World Wide Web. Derived from the lower layers, new application-layer-based DDoS attacks utilizing legitimate HTTP requests over to victim resources are more unavailable. This issue may be more critical when such attacks mimic or occur during the flash crowd event of a popular websites. Our technique presents on the detection of such new DDoS attacks, a novel scheme based on document popularity and also Access Matrix is defined to capture the spatial-temporal patterns of a normal flash crowd. A novel attack detector based on hidden semi-Markov model is proposed to describe the dynamics of Access Matrix and to detect the attacks. The entropy of document popularity fitting to the model is used to detect the potential DDoS attacks in application-layer. This paper analysis the attack detector with existing system drawback which presents proposed approach more efficient.**

***Index Terms*– Application Layer, Distributed System, Denial of Service and Attacks**

I. INTRODUCTION

Distributed denial-of-service attacks are comprised of packet streams from disparate sources. Streams converge on the victim consuming some key resource and rendering it unavailable to legitimate clients. Distributed machines that generate attack flows make traceback and mitigation very challenging. Some defense mechanisms concentrate on detecting the attack close to the victim machine characterizing it and filtering out the attack packets. While the detection accuracy of these machanisms is high the traffic is usually so aggregated that it is difficult to distinguish legitimate packets from attack packets. Internet derives in large part from the end-to-end principle [1] which enabled deploying a simple network infrastructure of packet forwarding nodes supported by a few routing protocols allowing networks applications to evolve independent of the core network. In particular congestion control mechanism of the TCP played vital role in achieving a robust and stable internet. Existing mechanisms have proven ineffective at protecting the internet from distributed denial-of-service attacks and increasingly frequent, global disturbance.

Traditionally DDoS attacks are typically carried out at the network layer. SYN/ACK flooding, UDP Flooding, ICMP Flooding, etc. as network layer DDoS defense are becoming more and more effective it can be identified the trends in the attackers strategy are shifting from network layer to application layer. Order to dodge detection; attackers are increasingly moving away from pure bandwidth floods to stealthy DDoS attacks that masquerade as flash crowds. They profile the victim server and minimic legitimate web browsing behavior of a large number of clients. These attacks target higher layer server resources like sockets. Disk bandwidth, database bandwidth and worker processes such as DDoS attacks in application layer. Countering APP-DDoS attacks is a new challenge because the requests originating from attackers are indistinguishable from the requests generated by legitimate users. The malicious request differs from the legitimate ones in intent but not in content. The malicious requests arrive from a large number of geographically distributed normal machines thus they cannot be use to stop the attacks because checking the password requires establishing a connection and allowing unauthenticated clients to access socket buffers and worker processes making it easy to amount an attack on the authentication mechanism itself. The attack signature of each App-DDoS attack is represented in abnormal user behavior, a technique used to detect such DDoS based on Web user browsing behavior.

II. SURVEY ON APPLICATION DDOS ATTACKS

Statistical approaches for detection of DDoS attacks including the use of MIB traffic variables [2], IP addresses and TTL values and TCP SYN/FIN packets for detecting SYN flooding attacks. In statistics packet attributes are used for detection and prevention of filtering policy for packet dropping and entropy Chi-Square statistics are used to differentiate between attack and normal packets while computes the conditional legitimate probability of a packet. Another way to defined DDoS attacks is the use of pushback. If the source of the attacks can be identified and traceback incrementally hop-by-hop to the source then rate limiting can be used to limit the scope and damage of the attacks. Packets are randomly marked for tracking the routers of the attack packets when sufficient packets are marked the victim can identify the network paths traversed by the attack traffic without requiring operational support from ISPs. Network layer provides partial solution to DDoS attacks no sufficient information is available at the transport layer to make intelligent decisions regarding App-attack.

Web user behaviors study in use of data mining to analyze the web user browsing behavior based on click-steam data [8].

To model the browsing behavior of users on the web and used the model in performance evaluation of web applications in use Markov chains to model the URL access patterns observed in navigation logs based on the previous state. However all these methods are not designed for on-line anomaly detection and their computational complexity is expensive to be run online.

The Internet service providers may offer subscribers DDoS defense in enhanced security services such as virtual-private networks ensure traffic flows only among a designated set of trusted computers and managed firewalls. Talpade [19] designed NOMAD a network traffic monitor deployed in a single transit route to detect network anomalies by analyzing packet-header information such as time-to-live source and destination addresses. Akella et al. a detection technique proposed to identify anomalies by comparing current traffic profiles with profiles of normal traffic as observed at edge routers which exchange information with other edge routers growth of confidence. Lakhina et al. [21] suggested a subspace method for characterizing network-wide anomalies by examining the multivariate time series of all origin-destination flows among routers in a transit network. Using principal component Analysis origin destination flows are decomposed into constituent Eigen values where top few eigenvectors depict normal traffic and remaining eigen values expose anomalies.

A. Review on Application Layer Data Security

Design of Layer is a technique is possible to dismantle complicated programs into a hierarchy service interface. It is possible to add stronger services by adding new layers over the layers rendering more basic services. Principles constitute the basis of the layered system

Each of the parallel layers on the server and client together provide service. Protocol specifies how the work is divided the format of the messages and the order of the transactions.

Each layer is built on the service of the layer. Service interface defines how each layer requests and receives the services of the layer under it. Interface must hide all the details of the work carried out under it and supply a collection of services.

At the higher layers the services is simpler example the lower layers may use the system services for hardware access on the computer while the higher layers render services such as transfer of files etc.

III. PROBLEM DEFINITION

Application layer is the most important to development, accepts the user request, and retrieves data from the database. Application layer may cause with distributed denial of Service attacks to avoid this we need to check whether the packet is normal or abnormal. Our proposed system identifies the user sending request, which the packet is normal allows into the application layer otherwise rejects the request from user.

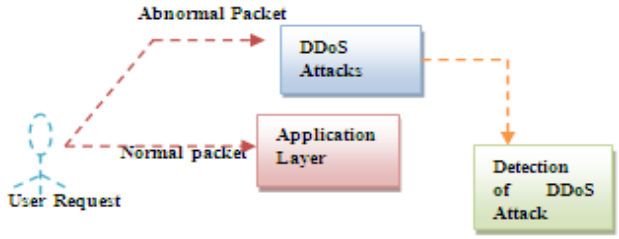

Fig. 1: Proposed System

Our work prevents and detects the distributed denial of service attacks using Semi-Hidden Markov model.

A. Classification of Distributed Denial of Service Attacks

1) Direct Flooding: The simplest case of a DDoS attack is the direct flooding attack. In this case, the attacker sends packets directly from his computer(s) to the victim's site. In this attack, the source address of the packets may be forged. There are many tools available to allow this type of attack for a variety of protocols including ICMP, UDP and TCP. Some common tools include stream2, synhose, synk7, synsend and hping2.

2) Remote Controlled Network: Remote controlled network attacks involve the attacker compromising a series of computers and placing an application or agent on the computers. The computer then listens for commands from a central control computer. The compromise of computers can either be done manually or automatically through a worm or virus. Typical control channels include IRC channels, direct port communication or even through ICMP ping packets. Remote controlled attacks are very difficult to trace to the original control computer.

3) Reflective Flooding: Reflective attacks forge the source address of the IP packets with the victim's IP address and send them to an intermediate host. When the intermediate host sends a reply, it is sent to the victim's destination address, flooding the victim. Depending on the type of protocol used and the application and configuration involved, amplification factors of 3 to several hundred are possible. Reflective attacks can be difficult to trace to the original attacker because the flood packets are actually sent from intermediate servers. In many types of reflective attacks, the intermediate servers are usually well known, public servers. The victim's service provider cannot block access to these sites and many times end up blocking all the traffic to the victim's site to allow other network traffic to get through:

- Smurf and Fraggle Attacks
- ICMP
- TCP SYN
- UDP ATTACK
- TTL expiration
- DRDOS

4) Worms: Worms are distinguished from viruses in the fact that a virus requires some form of human intervention to infect a computer where a worm does not. Worms have had

the ability to significantly disrupt the normal operation of the Internet since the Morris worm in 1988.Worms can create Internet wide events based on scanning and infection traffic volumes (Code Red, Slammer), automated DDoS events (MS Blaster), or by creating zombie networks used to launch large scale DDoS attacks. Worm propagation technology has advanced significantly in the past several years.

5) Viruses: Viruses have had a lesser but significant impact on network providers. They are often used today to build large zombie networks. These are usually dire warnings that tell the person to notify all their friends about a fictitious worm, virus, or other situation. Although never a significant Internet problem, these have clogged enterprise email systems and continue to circulate today.

6) Protocol Violation: All attacks could be considered protocol attacks in the sense that the attacker is sending packets in a manner not originally intended. Sometimes this is beneficial to the community as when Van Jacobson developed the trace route program using ICMP return codes from the routers. In many situations, however, this is not the case. Protocol violation attacks are generally referring to attacks that use IP protocols that are not valid or are reserved.

7) Fragmentation: Packet fragmentation can be used in two distinct areas: evasion of IDS detection and as a DoS mechanism. As a DoS mechanism, fragmentation is used to exhaust a system's resources while trying to reassemble the packets. These types of attacks have occurred against Check Point firewalls, Cisco routers and Windows computers

8) Network Infrastructure: Attacks directed at network infrastructure can have a serious impact on the overall operation of the Internet. These attacks can create regional or global network outages or slowdowns. Recent attacks against the Internet's root name servers caused enough concern for an FBI investigation into the attack. It sent a warning signal to the root Name servers operators to fortify the robustness of their infrastructure .Backbone services can cause significant network outages. This would include DNS and to a lesser extent RADIUS.

- Control Plane Attacks
- Management Plane Attacks

B. Anomaly

Anomaly detection refers to detecting patterns in a given data set that do not conform to an established normal behavior. The patterns thus detected are called anomalies and translate to critical and actionable information in several application domains. Anomalies are also referred to as outlier, surprise deviation etc.

Most anomaly detection algorithms require a set of purely normal data to train the model and they implicitly assume that anomalies can be treated as patterns not observed before. Since an outlier may be defined as a data point which is very different from the rest of the data, based on some measure, we employ several detection schemes in order to see how efficiently these schemes may deal with the problem of anomaly detection. The statistics community has studied the concept of outliers quite extensively. In these techniques, the data points are modeled using a stochastic distribution and points are determined to be outliers depending upon their relationship with this model. However with increasing dimensionality, it becomes increasingly difficult and inaccurate to estimate the multidimensional distributions of the data points. However recent outlier detection algorithms that we utilize in this study are based on computing the full dimensional distances of the points from one another as well as on computing the densities of local neighborhoods.

The deviation measure is our extension of the traditional method of discrepancy detection. As in discrepancy detection, comparisons are made between predicted and actual sensor values, and differences are interpreted to be indications of anomalies. This raw discrepancy is entered into a normalization process identical to that used for the value change score, and it is this representation of relative discrepancy which is reported. The deviation score for a sensor is minimum if there is no discrepancy and maximum if the discrepancy between predicted and actual is the greatest seen to date on that sensor. Deviation requires that a simulation be available in any form for generating sensor value predictions. However the remaining sensitivity and cascading alarms measures require the ability to simulate and reason with a causal model of the system being monitored. Sensitivity and cascading alarms an appealing way to assess whether current behavior is anomalous or not is via comparison to past behavior.

This is the essence of the surprise measure. It is designed to highlight a sensor which behaves other than it has historically. Specifically, surprise uses the historical frequency distribution for the sensor in two ways: It is those sensors and to examine the relative likelihoods of different values of the sensor. It is those sensors which display unlikely values when other values of the sensor are more likely which get a high surprise scores. Surprise is not high if the only reason a sensor's value is unlikely is that there are many possible values for the sensor, all equally unlikely.

1) Types of Anomaly Detection Systems: Anomaly detection build models of normal data and then attempt to detect normal model in observed data. The broad categories of anomaly detection techniques exist

Supervised anomaly detection techniques learn a classifier using labeled instances belonging to normal and abnormal class and then assign a normal or anomalous label to a test instance.

IV. USER REQUEST

User behavior is mainly influenced by the structure of web documents and the way users access web pages. Our proposed system considers the Application layer DDoS attack as a normal or abnormal browsing behavior.

A. Access Matrix

Web document popularity is defined by the request Hit Rate as $p_{it} = b_{it} / \sum^{N}_{i=1} b_{it}$ is the users average revisitation to the t th unit r_{it} is the normalized revistiation and T is the number of observation time units. Then we construct an N*T dimensional Access Matrix $A_{N*T.}$

$$\mathbf{A}_{N\times T} = [\bar{a}_1 \quad \bar{a}_2 \quad \cdots \quad \bar{a}_T] = [\mathbf{a}_1 \quad \mathbf{a}_2 \quad \cdots \quad \mathbf{a}_N]^{\mathrm{T}}$$

Where $\bar{a}_t = (a_{1t}, \ldots, a_{Nt})^{\mathrm{T}}, \mathbf{a}_i = (a_{i1}, \ldots, a_{iT})^{\mathrm{T}}$, and $a_{it} = p_{it}$ or r_{it}. We will use $a_{it} = T_{it}$ it is more suitable to detect the attacks that repeatedly request the same pages such as homepage, "hot" pages, or randomly selected pages from a given set. In some other cases when the attacks may cause the document popularity away from the Zipf-like distribution, let $a_{it} = p_{it}$. For the analysis in this work consider a spatial-temporal space constructed by AM in which presents the spatial distribution of popularity at the t^{th} time unit and presents the t^{th} document's popularity varying with time. It is mainly related to users' interest and website's structure (e.g., the distribution of contents and hyperlinks between web-pages) is mainly controlled by users' actions (e.g., click rate and browsing time).

B. Semi Hidden Markov Model

Existing work [6], [7] extends the Semi Hidden Markov Model algorithm to describe the stochastic process on document popularity's spatial distribution varying with time and monitor the App-DDoS attacks occurring during flash crowd event.

Semi hidden Markov model (HMM) with variable state duration. The SHMM is a stochastic finite state machine specified by (Q, $\{\pi_i\}$, $\{a_{ij}\}$,$\{b_i(k)\}$,$\{ p_{i(d)}\}$) where Q={1,….M} is a discrete set of hidden states with cardinality M; q_t€Q denotes the state that the system takes at time is the probability distribution for the initial state satisfying $\sum_i\pi_i=1$ a_{ij}= Pr[q_t=j q_{t-1}=i] is the state transition probability from state i to j state satisfying $\sum_j a_{ij}=1$, for I ,j€Q ;T_t €{1,….D}denotes the remaining time of the current state q_t with D representing the maximum interval between any two consecutive state transitions Pi(d)=Pr[T_t=d/ q_t =i] is the state residual time distribution satisfying $\sum_d$Pi(d)=1, for i€Q ,d€{1,…D};$b_i(k)$= Pr[O_t = v_k q_t=i] is the output distribution for given state i, satisfying $\sum_k b_i(k)=1$, for i€Q,k€{1,…K} and Q_t denotes the observed vector at time t taking values from $\{v_{1,\ldots}v_k\}$ if the pair of process takes on value , semi-Markov chain will remain in the current state until time and transits to another state at time.

The parameter estimation of SHMM can be done by the following forward and backward algorithm [8]. The forward and backward variables are defined as follows:

$$\alpha_t(i,d) \equiv \Pr\left[\vec{o}_1^t, q_t = i, \tau_t = d|\lambda\right]$$
$$\beta_t(m,d) \equiv \Pr\left[\vec{o}_{t+1}^T\, q_t = i, \tau_t = d, \lambda\right]$$

Which can be iteratively calculated by the forward and backward algorithms? Three joint probability functions are defined by:

$$\xi_t(i,j) \equiv \Pr\left[\vec{o}_1^T, q_{t-1} = i, q_t = j\right]$$
$$\eta_t(i,d) \equiv \Pr\left[\vec{o}_1^T, q_{t-1} \neq i, q_t = i, \tau_t = d\right]$$
$$\gamma_t(i) \equiv \Pr\left[\vec{o}_1^T, q_t = i\right]$$

Which can be readily determined by the forward and backward variables. Then, the model parameters can be estimated by the following formulas:

$$\hat{\pi}_i = \gamma_1(i) \Big/ \sum_{i=1}^{N} \gamma_1(i)$$
$$\hat{a}_{ij} = \sum_{t=1}^{T} \xi_t(i,j) \Big/ \sum_{t=1}^{T}\sum_{n=1}^{N} \xi_t(i,j)$$
$$\hat{b}_i(\vec{r}) = \sum_{t\,\vec{o}_t=\vec{r}}^{T} \gamma_t(i) \Big/ \sum_{t=1}^{T} \gamma_t(i)$$
$$\hat{p}_i(d) = \sum_{t=2}^{T} \eta_t(i,d) \Big/ \sum_{t=2}^{T}\sum_{d=1}^{D} \eta_t(i,d).$$

Defined the entropy (E_n) of observations fitting to the SHMM and the average logarithmic entropy (ALE) per observation as follows:

$$En = \Pr\left[\vec{o}_1^T|\lambda\right] = \sum_{i,d} \Pr\left[\vec{o}_1^T, (q_T, \tau_T) = (i,d)\,\lambda\right]$$
$$ALE = \ln\left(\Pr\left[\vec{o}_1^T|\lambda\right]\right)/T.$$

C. Comparative Study

we have existing systems that consume the network bandwidth and deny service to legitimate users, server overwhelming and large amount of data is required to train, only positive data are used to train to solve all these drawbacks our proposed system identifying abnormalities and serve them in different priority queues, filter when the network heavily loaded, use more accurate identification and identifies the abnormalities with small amount of training data.

V. CONCLUSION

Application layer not only interface with the database and user request, this work creating defenses for attacks requires monitoring dynamic website activities in order to obtain timely and signification information. We proposed a detection architecture technique aiming to monitoring website in order to reveal dynamic shifts in normal burst traffic, which might signal onset of App-DDoS attacks during the flash crowd event. Our method reveals early attacks merely depending on the document popularity obtained from the server log. We analyze with different App-DDoS attack modes (i.e., constant rate attacks, increasing rate attacks and stochastic pulsing attack) during a flash crowd event collected from a real trace. In our experiments, when the detection threshold of entropy is set 5.3, the DR is 90% and the FPR is 1%. It also demonstrates that the proposed architecture is expected to be practical in monitoring App-DDoS attacks and in triggering more dedicated detection on victim network.

REFERENCES

[1] J. Saltzer, D. Read and D. Clark, "End-to-End Arguments in System Design", ACM Trans Computer System, Vol. 2, No. 4, pp. 277-288, Nov 1984.

[2] Burleson, D., "Managing security in a distributed database environment", DBMS, Vol. 8, pp. 72-77, 1998.

[3] R. Stone. Centertrack: An IP Overlay network for tracking dos floods", In Proceedings of 9th USENIX Security Symposium, August, 2000.

[4] D. Dean, M. Franklin, and A. Stubblefield, "An Algebraic Approach to IP Traceback," ACM Trans. Information and System Security, 2004.
[5] S. Ranjan, R. Karrer, and Knightly, "Wide area redirection of dynamic content by Internet data centers," in Proc. 23rd Ann. Joint Conf. IEEE Comput. Commun. Soc., Vol. 2, pp. 816–826, March, 2004.
[6] [Online]:http://www.caida.org/analysis/security/sco-dos/
[7] [Online]: http://ita.ee.lbl.gov/html/traces.html
[8] J. Cao, W. S. Cleveland, Y. Gao, K. Jeffay, F. D. Smith, and M. Weigle, "Stochastic models for generating synthetic HTTP source traffic," in Proc. IEEE INFOCOM, Vol. 3, pp. 1546–1557, 2004.
[9] NS2 [Online]. Available: http://www.isi.edu/nsnam/ns/
[10] W. Wang, X. Guan, and X. Zhang, "A novel intrusion detection method based on principle component analysis in computer security," in Proc. Int. Symp. Neural Networks, Dalian, China, pp. 657–662, Part-II, August, 2004.
[11] Y. Xie and S. Yu, "A dynamic anomaly detection model for web user behavior based on HsMM," in Proc. 10th Int. Conf. Comput. Supported Cooperative Work in Design (CSCWD 2006), China, Vol. 2, pp. 811–816, May, 2006.

V. Venkata Ramana Bachelor of Engineering in Computer Science Engineering from NMaMIT, NITTE, Manglore University, M. Tech CSE from JNTUH. His areas of interest include Data Mining & Warehousing and Computer Networks.

P. Shilpa Choudary pursuing M. Tech Software Engineering at Gurunank Engineering College, B. Tech Electronic Communication Engineering from SRIT (JNTUH). Her areas of interest include Wireless Sensor Networks, Computer Networks currently focusing on Application Layer.

Maya B. Dhone pursuing M. Tech Software Engineering at Gurunank Engineering College, B.E Computers from Maharashtra Academy of Engineering, Alandi (devachi), Maharashtra University– Pune. Her areas of interest include Wireless Sensor Networks, Computer Networks currently focusing on Application Layer.