# Detection and Counter Measure of AL-Ddos Attacksin Web Traffic

R.Sreenath[1] and E.S.Phalgunakrishna[2]

[1] SREE VIDYANIKETHAN ENGINEERING COLLEGE

*Received: 6 February 2015 Accepted: 3 March 2015 Published: 15 March 2015*

## Abstract

Distributed Denial-of-Service (DDoS) assaults are a developing danger crosswise over Internet, disturbing access to Information and administrations. Presently days, these assaults are focusing on the application layer. Aggressors are utilizing systems that are exceptionally hard to recognize and relieve. In this task propose another technique to recognize AL-DDoS assaults. This work separates itself from past techniques by considering AL-DDoS assault location in overwhelming spine activity. In addition, the identification of AL-DDoS assaults is effectively deceived by glimmer group movement. By analyzing the entropy of AL-DDoS assaults and glimmer swarms, these model output be utilized to perceive the genuine AL-DDoS assaults. With a quick AL-DDoS identification speed, the channel is equipped for letting the real demands through yet the assault movement is halted.

***Index terms—***

# 1 Introduction

Denial-of-Service (DoS) assault is an endeavor by aggressors to keep the true blue clients from utilizing the data administration. In a DDoS assault, these endeavors originate from an extensive number of circulated hosts that organize to surge the exploited person with a plenitude of assault bundles all the while. Conveyed foreswearing of-administration (DDoS) assaults present genuine dangers to servers in the Internet. DDoS assaults include in soaking the target machine with appeals, such that it can't react to authentic movement. Such assaults for the most part prompt a server over-burden.

To dispatch a DDoS assault, the aggressors first creates a system of bargained PCs that are utilized to produce the colossal volume of activity expected to refuse any assistance to honest to goodness clients of the victimized person. At that point the aggressor introduces assault apparatuses on the bargained hosts of the assault system. The hosts running these assault apparatuses are known as zombies, and they can be utilized to complete any assault under the control of the aggressor. The vast majority of the current procedures can't segregate the DDoS assaults from the surge of honest to goodness getting to.

These assaults are focusing on the application level. Application layer DDoS assaults may concentrate on debilitating the server assets, for example, Sockets, CPU, memory, circle/database data transmission, and I/O transfer speed. These assaults are normally more productive than TCP or UDP-based assaults, obliging less system associations with accomplish their malevolent purposes. They are likewise harder to distinguish, both on the grounds that they don't include a lot of activity and in light of the fact that they appear to be like ordinary kind movement.

# 2 II.

# 3 Rival Methods

? We have adopted a hidden semi-Markov process to present the behavior of Internet users .The hidden semi-Markov approach is a complex algorithm. When users visit a website, it traces and records the whole visiting history of each user. ? It is noticeable that the hidden semi-Markov method is unlikely to perform effectively in backbone traffic. ? Another typical approach against AL-DDoS attacks is to use CAPTCHA. This method requires users to recognize strings in a fuzzy picture and submit a response to a web server for authentication. However, users sometimes consider this operation as a negative experience to surf the Internet. ? The introduced wavelets to identify anomalies in network traffic. But wavelet analysis is generally a post-mortem analysis and cannot be used for online processing.

Previously proposedsignal analysis of network traffic anomalies mechanism to voluntarily increase the bandwidth utilization of legitimate users. However, this approach cannot reduce the network congestion and the load of web servers. A countermeasure that consisted of a suspicion assignment process and a DDoS-resilient scheduler.

The suspicion process assigns a continuous 'valued vs. binary' measure onto each client session. It further utilizes these values to determine if and when to schedule the requests of a session. However, this approach is still too time-consuming to detect AL-DDoS attacks in large volume traffic.

# 4 III.

# 5 Proposed System

In this paper, we are motivated to design a defense system at the backbone level. This system is able to detect AL-DDoS attacks targeting Internet web servers. Currently, most of these web servers are deployed together in a data center connecting directly to the backbones. Thus, it is critical to implement an effective method to detect AL-DDoS attacks and filter the malicious traffic in backbones before they causes detriments to the web servers. The proposed system has low complexity and can real-timely run in high volume traffic.

One way to protect from DoS attacks is to allow only authorized clients to access the web server. Compared with non-attack cases, the number of requests in a session increases significantly in a very short time period in DDoS attack cases. Considering the above two issues, a hybrid approach for countering application layer DDoSatttacks is proposed. This approach gives priority to the good (legitimate) clients, while severely limiting the access to the attackers.

Each client is assigned with a trust value by the server based on the access behavior. A client's trust value is embedded in a HTTP cookie that is included in all server responses to the client. Using the cookie, a legitimate client can include the trust value in all its future requests to identify itself to the server. A client presenting a valid trust value to the server will be given the priority over other requests. New clients are assumed to be assigned with the lowest trust value by default by the server and updated in the response. The trust value varies according to the access pattern of the client. The trust values are assigned in such a way that trustattacker<trustnew user <trustlegitimate user In addition, the user's browsing behavior in multiple aspects is extracted from the system log during non-attack cases. Then the entropy of requests per session is calculated. Entropy is an information theoretical concept, which is a measure of randomness. The entropy is employed in this paper to measure changes of randomness of requests in a session for a given time interval. Entropy is applied as a second layer of filtering the suspicious flow. The second filtering mechanism is required to identify an attacker who acts like a legitimate client because, an attacker may behave benignly until it attains a highest trust value and then begin to misbehave.

The detection mechanism is deployed at the server. A session connection request first reaches the system, and then the proposed scheme either drops or forwards the requests based on the trust value obtained in the past session, calculates the entropy deviation of request rate. If the deviation is more (exceeds threshold), then drop the session immediately. Otherwise, schedule the session based on the system workload and the trust value of the user. The client who behaves better in past session will obtain higher degree of trust. The highest trust value first policy is used to schedule the requests for the server.

IV.

# 6 Block Diagram

The above figure shows system architecture of the application. A session connection request first reaches the system, and then the proposed scheme either drops or forwards the requests based on the trust value obtained in the past session, calculates the entropy deviation of request rate. If the deviation is more (exceeds threshold), then drop the session immediately. Otherwise, schedule the session based on the system workload and the trust value of the user. The client who behaves better in past session will obtain higher degree of trust. The highest trust value first policy is used to schedule the requests for the server. Analogy The detection of DDoS attack is carried out as follows:

? Initially, the client embeds its trust value (Trust on the session request (rxy) and sends it to the server. ? The server, on receiving the session request, validates the trust value. ? If valid, it forwards the request (r ).

? Otherwise, the session is considered suspicious and dropped. ? Then the entropy (H(R)) for the incoming requests in a session is calculated and the degree of deviation with the predefined value is estimated. ? The greater the deviation, the more suspicious the session is. ? If the session is found suspicious, then it is assigned with the lowest trust value and dropped immediately. ? Otherwise, the requests are scheduled to get the service from the web server.

# 7 a) Trust value computation

Once the request is accepted, the request is forwarded to the application. When the server sends a response to the client, it updates the trust value as follows:

Let req be the client's request and res be the corresponding response generated by the server. Let t rs be the time taken by the server to respond for the request req and ut denotes the utility of the request, req.

In this approach, a simple benefit function [2] is used.

Where ? is a tunable parameter.

Here, additive increase multiplicative decrease strategy is used to calculate the new trust value.

If B(req) > 0, then the new trust value is computed as follows:Trust new = trustold + ? * B(req)?..(2)

Otherwise,Trust new = trust old /(?(1-B(req)))?.(3)

The additive increase ensures that the trust value slowly increases as the client behaves benignly; while the multiplicative decrease ensures that the trust value drops very quickly upon detecting a DoS attack from the client.

# 8 b) Entropy calculation

Let the request in a session be denoted as r xy , where x, y I, a set of positive integers. 'x' denotes the request number in session 'y'. Let |(r y ,t)| denote the number of requests per session y, at a given time t. Then, For a given interval Î?"t, the variation in the number of requests per session y is given as follows;

The probability of the requests per session y, is given by Let R be the random variable of the number of requests per session during the interval Î?"t, therefore, the entropy of requests per session is given as Based on the characteristics of entropy function, the upper and lower bound of the entropy H(R) is defined as 0 ? H(R) ? log N? (8) where N is the number of the requests.

Under DoS attack, the number of request increases significantly and the following equation holds Where C is the maximum capacity of the session.

# 9 c) Rate Limiter

To avoid falsely detection, rate-limiter is introduced. Once the entropy is calculated, compute the degree of deviation from the predefined entropy. The system first sets a threshold for acceptable deviation. If the computed deviation exceeds the threshold, then the session is forced to terminate immediately. Otherwise, second level filter is applied by the rate limiter. The system also defines a threshold for validating a user based on the trust score. A user is considered to be legitimate only if the trust score exceeds the threshold. Otherwise, the user is considered malicious and the session is dropped immediately. The legitimate sessions are then passed to the scheduler for getting service from the server.

# 10 d) Scheduler

If the user is legitimate, then the scheduler schedules the session based on the highest trust value first (user with highest trust value) policy. The wellbehaved users will have a little or no deviation. In such case, the legitimate user gets a quicker service. In addition to the scheduling policy, system workload is also considered before scheduling the request for getting service.

# 11 e) Algorithms

Algorithm to compute the entropy from system log Input: system log 1. Extract the request arrivals for all sessions, page viewing time and the sequence of requested objects for each user from the system log. 2. Compute the entropy of the requests per session using the formula: a. Detection Algorithm Input the predefined entropy of requests per session.

Define the threshold for allowable deviation (Td) For each session waiting for detection Extract the trust value from the request Validate the trust value If the trust value issued is valid Extract the requests arrivals Compute the entropy for each session using (7) Compute the degree of deviation

# 12 Conclusion

In this paper, an effective and efficient hybrid scheme against DDoS attacks based on trust value and information metric (entropy) is proposed. This approach not only counters the illegitimate flows but also avoids the flooding of the legitimate flows. Further is add detect trust value is used to detect the legitimate user from the attackers at the first level. Then, based on the information metric of the current session, the sessions that are assumed to be suspicious are dropped. The legitimate flows are then scheduled by the scheduler based on the system workload the trust value of the client. Thus the legitimate clients gets more priority in accessing the information and services. [1] [2]

**11**

Figure 1: A 11 Global

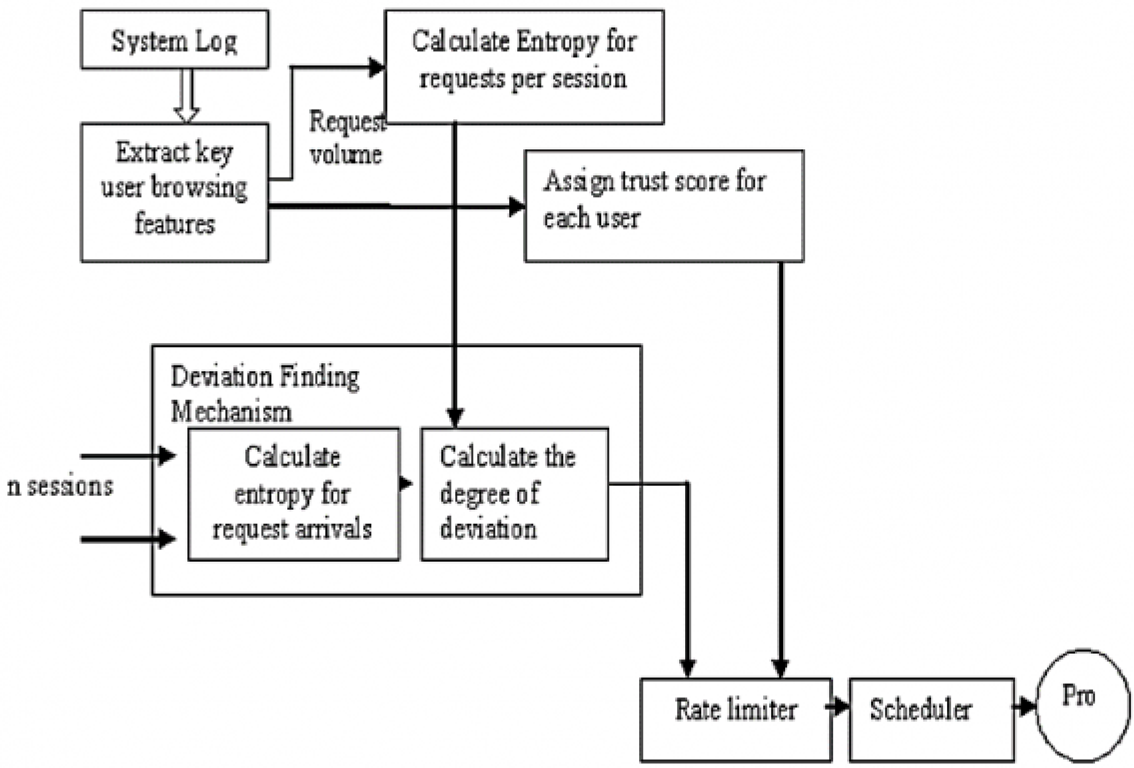


Figure 2: ?

| Client IP Address | Trust value | Degree of derivation | Policy | Attack Type |
| --- | --- | --- | --- | --- |
| 172.016.112.100 | 4 | 20% | legitimate client | no |
| 194.027.251.021 | 5 | 10% | legitimate client | no |
| 135.008.060.182 | 1 | 50% | suspicious | Ntis attack |
| ...... | .... | .... | .... | |
| ...... | .... | .... | .... | |

Figure 3:

**1**

Figure 4: Table 1 :

[D = |hnew (ed.)] , D = |hnew . R)| -|H(R)| (ed.)

[Devi and Yogesh (2012)] ‘A Hybrid Approach to Counter Application Layer DDoS Attacks’. S Devi , P Yogesh . *International Journal on Cryptography and Information Security (IJCIS)* June 2012. 2 (2) .

[Xie and Yu ()] ‘A large-scale hidden semi-Markov model for anomaly Detectionon user browsing behaviors’. Y Xie , S Yu . *IEEE/ACM Trans.Netw* 2009. 17 (1) p. .

[Barford et al. ()] ‘A signal analysis of network traffic anomalies’. P Barford , J Kline , D Plonka , A Ron . *Proceedings of the 2 nd ACMSIGCOMM Work shop on Internet Measurement*, (the 2 nd ACMSIGCOMM Work shop on Internet MeasurementNew York, NY, USA) 2002. IMW’ 02, ACM. p. .

[Kandula et al. ()] ‘Botz-4-sale: surviving organized Ddos attacks that mimic flash crowds’. S Kandula , D Katabi , M Jacob , A Berger . *Proceedings of the 2nd Conference On Symposiumon Networked Systems Design and Implementation,” NSDI’05,USENIX Association*, (the 2nd Conference On Symposiumon Networked Systems Design and Implementation,” NSDI’05,USENIX AssociationBerkeley, CA, USA) 2005. p. .

[Von Ahn et al. ()] *Captcha: using hardai problemsFor security*, L Von Ahn , M Blum , N J Hopper , J Langford . 2003. EUROCRYPT. p. .

[Walfish et al. ()] ‘DDoS defense by offense’. M Walfish , M Vutukuru , H Balakrishnan , D Karger , S Shenker . *ACM Trans. Comput. Syst* 2010. 28 (1) p. .

[Ranjan et al. ()] ‘DDoS-resilient scheduling to counter application layer attacks under imperfect detection’. S Ranjan , R Swaminathan , M Uysal , E Knightly . *Proceedings”. INFOCOM 2006. 25th IEEE International Conference on Computer Communications*, (”. INFOCOM 2006. 25th IEEE International Conference on Computer Communications) 2006. p. .

[Zhou et al. ()] ‘Detection and defense of application-layer DDoS attacks in backbone Web traffic’. Wei Zhou , Weijiajia , Yangxiang Shengwen , Wanlei Zhou . *Future Generation Computer Systems* 2014. 38 p. .

[Xie and Zhengyu ()] ‘Monitoring the application-layer ddos attacks for popular websites’. Y Xie , S Zhengyu . *IEEE/ACM Trans. Netw* 2009. 17 (1) p. .

[Arbor and Networks ()] *Worldwide network infrastructure security report*, Arbor , Networks . 2011. Arbor Networks. (Tech. Rep.)