EXPLORATIONS ON ENGINEERING LETTERS (EEL)
VOLUME 1, ISSUE 1 (2016):PP.246-252
SANA ACADEMIC PRESS

Preventing the Application Layer Intrudes Attacks for Cloud attack

Gowtham.B[1], R.Subhashni[2], R. Latha[3]
[1]Research Scholar, St. Peter's University, Chennai.7
[2]Asst.Prof, Dept. of Computer Application, St. Peter's University, Chennai
[3]Prof. & Head., Dept. of Computer Science & Applications, St. Peter's University, Chennai
samoshik@gmail.com

Abstract— *Intrudes attack is a critical threat to the Internet. Derived from the low layers, application layer based Intrudes attacks HTTP requests to affect victim resources are more undetectable. The case may be extremely serious when such attacks occur during the flash crowd event of a popular Website. Focusing on such new Intrudes attacks, this scheme is introduced. An Access Matrix is to capture the spatial temporal patterns of a flash crowd. Principal component analysis and independent component analysis are used in the multidimensional Access Matrix. A detector based on hidden semi Markov model is proposed to describe the dynamics of Access Matrix and to detect the attacks over the network layer. This model is used to detect the application layer Intrudes attacks which demonstrates the effectiveness of the proposed method.*

Keywords: *Application-layer, distributed denial of service (Intrudes),Access Matrix(AM),Hidden-semi Markov Model(HsMM)*

I.INTRODUCTION

Distributed denial of service (Intrudes) attack has caused severe damage to servers and will causes even intimidation to the development of new Internet services.Intrudes attacks are carried out at the network layer which are called Net Intrudes attacks. The intent of these attack is to consume the bandwidth and deny service to users of the victim systems. When the simple Net Intrudes attacks fail, attackers shift their offensive strategies to application-layer attacks and establish a more sophisticated type of Intrudes attacks. Another new phenomenon of network traffic called flash crowd [4], [5] has been noticed by researchers, which produces a surge in traffic to the Website App Intrudes attacks may be stealthier and more dangerous for the popular websites than the general Net Intrudes attacks. This paper introduces a scheme to capture the spatial temporal patterns of a normal flash crowd event and to implement the App-Intrudes attacks detection. Our contributions in this paper are four fold: 1) the Access Matrix (AM) to capture spatial-temporal pat-terns [6], [7], we use hidden semi-Markov model (HsMM) [8] to describe the dynamics of AM and to deal with the multidimensional data for HsMM; and 4) we design the monitoring architecture and validate it by a real flash crowd traffic.

II. APP-INTRUDES ATTACKS

The first characteristic of App-Intrudes attacks is that the application layer requests originating from the compromised hosts are indistinguishable from those generated by legitimate users.Usually, App Intrudes

attacks utilize the weakness enabled by the standard practice of opening services such as HTTP through most firewalls. Many protocols and applications, can use these openings to tunnel through firewalls by connecting over a standard TCP port 80 (e.g., Code Red virus). Attack requests aimed at these services may pass through the firewall without being identified. Furthermore, attackers may request services to the point where other clients are unable to complete their transactions. The second characteristic of App Intrudes attacks is that the attackers aiming at some special popular Websites, Since such Websites become more and more for the increasing demands of information and electronic commerce, network security has to face a new challenge i.e., the fourth scenario of our clusters for Intrudes attacks.Finally, compared with the consumption of resources , App-Intrudes attacks may not need to consume a lot of network bandwidth. Therefore, the traditional Intrudes detection schemes designed for bandwidth exhausting attacks become ineffective.

III. LITERATURE SURVEY

Various researches have been done during the past with regard to the detection of Intrudes attacks from three layers of OSI namely layer 3-Network layer, layer 4-Transmission layer and layer 7-Application layer. The attacks done on the application layer were very fewer in the past and hence the researches done on Application-Layer protection are also few in number. Techniques to detect Application-layer INTRUDES attacks are highlighted below:

A. Client Puzzle Protocol

Client Puzzle Protocol (CPP) is an algorithm that will not allow any abuse of the server resources. In this algorithm, any client who needs to establish a connection with the server has to first correctly solve a mathematical puzzle. After solving the mathematical puzzle, the client returns the solution to the server and the server will either quickly confirm, reject or drop the connection based on the solution of the client. The client needs to perform only a minimum amount of computation as the puzzle is made simple and easily solvable. Only negligible computational cost would be experienced by genuine used. Any client who tries to simultaneously establish a large number of connections will be unable to do so because of the computational cost .

B. Intrusion Detection System

A software that is used to automate the intrusion detection process is known as intrusion detection system (IDS).

C. Ingress Filtering

In computer networks a technique used to make sure that packets coming into the networks are actually from the edge router's previous connection history. Also when the database size happens to be large enough search time increases and hence the delay.

2) Srikanth Kandula, Dina Katabi, Matthias Jacob and Arthur Berger in their IEEE paper "Botz 4 Sale: Surviving Organized INTRUDES Attacks That Mimic Flash Crowds" [5], described the design and implementation of Kill Bots, a kernel extension to protect Web servers against INTRUDES attacks that masquerade as flash crowds. The author suggests the use of CAPTCHAs to distinguish the IP addresses of the legitimate clients from those of attack machines. The design is implemented in the Linux kernel and evaluated it in Planet lab. Kandula et al. design a system to protect a web cluster from INTRUDES attacks by designing a probabilistic authentication mechanism using CAPTCHAs. Unfortunately, requiring all users to solve graph puzzles may result in the possibility of annoying users and introducing

additional service delays for legitimate users. This paper may not serve the purpose if any automated techniques are being used by attackers to solve the graphical puzzles. Fig. 1. shows the Kill Bots Overview.

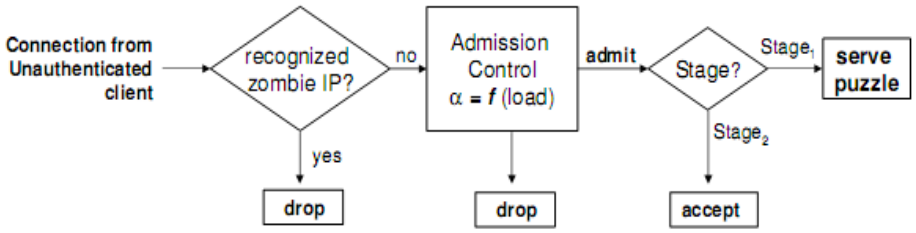

Fig. 1. Kill-Bots overview (Note that graphical puzzles are only served during stage).

3) "Monitoring the Application-Layer INTRUDES Attacks for Popular Websites" IEEE 2009, proposed by Yi Xie andShun-Zheng Yu [1]. This method involves high mathematical computations.

IV. RESEARCH ELABORATION

Web user behavior is mainly influenced by the structure of website (e.g., the Web documents and hyperlink) and the way users access web pages. In this paper, the Application Denial of Service attack is considered as anomaly browsing behavior. Characteristics of Web access behavior shown in Fig. 2, plots the HTTP request number (average user hits)per 5 sec during the burst Web workload on the popularwebsite that is requesting for protection. It is observed that the normal flash crowd is mainly caused by the sudden increment of user request rate.

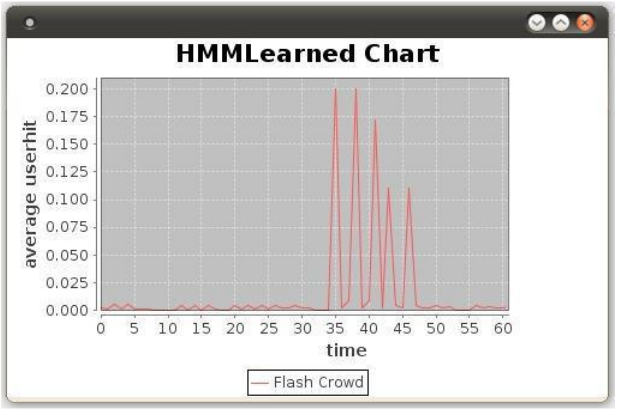

Fig. 2. Normal flash crowd.

Characteristics of Web access behavior whenever there is any attempt of an application-layer INTRUDES attack by increasing the traffic to a popular-website is shown in Fig. 3

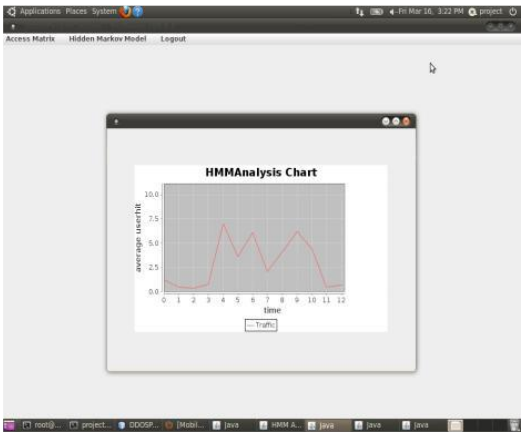

Fig. 3. Application-Layer INTRUDES attack.

These results show that the user's access behavior profile can be used to detect the abnormal varieties of user'sbrowsing process during the flash crowd. The document popularity has been widely used to characterize the user behavior and improve the performance of Web server and Internet cache. In this paper the defensive mechanism against Application Layer Intrudes attack requires the following

A. Access Matrix

Information about the users accessing the website are stored in a matrix called Access Matrix [1]. The access matrix model is the policy for user authentication. It is used to describe which users have access to what objects.

B. Hidden Markov Model

If a system being modeled is having unobserved state and is assumed to be a Markov process then such a statistical model is known as Hidden Markov Model (HMM). In a regular Markov model, the state is directly visible to the observer, and hence the state transition probabilities are the only parameters. In a Hidden Markov model, the state is not directly visible, but output dependent on the state is visible. Each state has a probability distribution over the possible tokens. Therefore the sequence of tokens generated by an HMM gives some information about the sequence of states. The adjective 'hidden' refers to the state sequence through which the model passes, not to the parameters of the model [7]. Some of the applications of Hidden Markov models are in temporal pattern recognition.

V. SYSTEM ANALYSIS

A. Existing System

Existing detection mechanisms operate at the network level to detect INTRUDES floods in the Cloud network. For example, the anomaly detection system assigns every packet a score based on the probability of it being a legitimate packet given the attribute values it carries. In contrast, there are other mechanisms which detect anomalies in the traffic distribution instead of traffic volumes. However, the attacks considered in this paper cannot be detected by such tools as the attacks may not necessarily deviate the network statistics in either volume or distribution. Other detection mechanisms attempt to catch intrusions both at the network and the host level. Distinguishing a INTRUDES attack from a flash crowd has also proven difficult. Internet Intrudes attack is real threat on websites such as Yahoo,eBay, etc i.e. services were unavailable for several hours due to Lack of defense mechanism on current Internet and

also for individual Systems. The on hand feature for user behaviors can be in the following ways. The first is based on probabilistic model, a double Pareto distribution for the linkchoice, and a log-normal distribution for the revisiting. The second is based on click-streams and web content, e.g., data mining to capture a web user's usage patterns . The third is based on the Markov model.

B. Proposed System

The objective of proposed system is to protect the popular websites from Application-Layer INTRUDES attack. The proposed system monitors the traffic and detects the application layer INTRUDES attack based on user logs and user behavior. Once the attack is identified the attacker system will be blocked from further overwhelming the traffic. Techniques used are Access Matrix and Hidden Markov Model. The Access Matrix defines access privilege of various users of a particular Websites. The administrator of popular websites has to register themselves with the INTRUDES software for protecting their site from application layer Intrudes attack. After registration process the administrator of popular website has to generate the access matrix. The data provided to access matrix are as follows:

1) Information regarding their legitimate users.
2) Url's which their legitimate users are not allowed to access.
3) The level of accessing by the legitimate users etc.

Whenever any user is requesting a page from the administrator of the popular website, the packets are captured and the following informations are identified:

1) The IP address of the requesting system.
2) User id of the user who is requesting the service.
3) Access level of the user.
4) The url which is tried to access often.
5) The services (webpages),which is frequently requested.

These information will be crosschecked with access matrix and if there is any discrepancy (ie. If a user is trying to access the webpage which is restricted to him) then an attack is detected and the requests coming from that IP will be blocked. A Threshold value is set and if a user is requesting a webpage greater than the threshold value is also considered as an attempt of attack. Hence this technique is used to detect application INTRUDES attack using the logs of the web server and threshold value. The Hidden Markov Model is used to detect application Intrudes attack based on the user behaviour. The input to the Hidden Markov model is the average hit and average page count of the user and the output of the model will be the decision made (whether he is a normal user or an intruder) as shown in Fig.4. Whenever input is given we get the output and the state transitions within the system is not visible, hence the name Hidden Markov Model. There are two phases in Hidden Markov Model. They are Training phase and Comparison phase.

Training phase (Learning phase): The Training model is generated periodically during this phase. Training period is set to 1 minute. The legitimate users of the popular website are allowed to access the website until the training period expires. All the requests coming to that popular website will be captured by the INTRUDES Protection system and the information about the IP address of the system requesting service, the user-id of the user who is requesting the service, access level of the user, which url he is trying to access and what are the services (webpages) he is requesting can be identified. For every 5 second the average hits of the user accessing the webpage is computed and average hit for 1 minute is

computed based on usertypes. The Training Model contains information about the average hits for all user-types accessing the website and this represents the normal flash crowd. Characteristics of Web access behavior as shown in Fig. 3, plots the HTTP request number (average user hits) per 5 sec during the burst Web workload on the popular-website that is requesting for protection. Comparison phase (Analysis phase): whenever any user,requests service from the popular website, the INTRUDES Protection system will capture the requests. The user behavioral patterns are computed and are compared with the trained model which is developed during the Learning phase.

C. Advantages

1) Any one can make these systems to take into account in the user's series of operations information.
2) Suitable for online detection as there is an intensive computation for the page content processing.
3) The best effectiveness of packet filter.

VI. DISCUSSIONS

The conventional security technologies such as firewalls [8] Intrusion Detection Systems (IDSs) [9] and access control lists in routers are unable to defend networks from App-Intrudes attacks. The main reason is that, it is almost impossible to differentiate between legitimate and attack packets since the potency of flooding Distributed Denial of Service attacks depends only on the volume of attack traffic and does not depend upon the exploitation of software bugs or protocol vulnerabilities. Consequently, flooding Intrudes packets do not need to be malformed, such as invalid fragmentation field or a malicious packet payload. As a result, the flooding Intrudes traffic looks very similar to legitimate traffic [10]. It is a real challenge to defend against these attacks as flooding Intrudes attacks are very dynamic to elude existing defense systems. Due to the seriousness of Distributed Denial of Service attacks and the growth of sophistication of the attackers led to development of numerous defense mechanisms. But still, the tremendous growth in the number of Distributed Denial of Service attacks and their financial implications press the need of a comprehensive solution. The comprehensive solution against Distributed Denial of Service attacks can be devised only if the Internet community incorporates better ways to accumulate details of attack

VII. CONCLUSION

In order to create defense for attacks ,it is necessary to maintain timely and significant information by monitoring dynamic network activities. Most of the efforts and researches focuses on detecting Net-Intrudes attacks with stable background traffic. This paper aims to signal the Application Layer Intrudes attacks during flash crowd event. This reveals the dynamic shifts in normal burst traffic and thus monitoring Web traffic. The proposed method is based on Access Matrix and Hidden Markov Model. This early attacks merely depending on the threshold specified, user logs, user behavior and gives the privilege for administrator to effectively identify and block the connections for specified attacking host. Measures are taken to check for IP spoofing as an additional detection process. Further this scheme can be applied in the client server architecture thus providing double protection.

REFERENCES

[1] Y. Xie and S. Z. Yu, "Monitoring the Application-Layer Intrudes Attacks for Popular Websites," in *Proc. Networking, IEEE/ACM Transactions,* Feb. 2009, pp. 15-25.

[2] G. Coulouris and J. Dollimore, *DISTRIBUTED SYSTEMS*, 4th *EDITION*, pearson education 2005.

[3] C. Chang, "Defending Against Flooding-Based Distributed Denial of Service Attacks: A Tutorial," *Computer Journal of EEE Communication Magazine*, vol. 40, no. 10, pp. 42-51, 2002.

[4] Incident Note IN-2004-01 W32/Novarg. (2004). A Virus. CERT.[Online]. Available:http://www.cert.org/incident_notes/ IN-2004-01.html.

[5] S. Kandula, D. Katabi, M. Jacob, and A. W. Berger. (2004). *Botz-4- Sale: Surviving Organized Intrudes Attacks that Mimic Flash Crowds*.MIT,Tech.Rep.TR-969. [Online]. Available: http://www.usenix.org/events/nsdi05/tech/kandula/ kandula.pdf

[6] S. Z. Yu and H. Kobayashi, "An efficient forward-backward algorithm for an explicit duration hidden Markov model," *IEEE Signal Process. Lett.*, vol. 10, no. 1, pp. 11–14, Jan. 2003.

[7] T. Peng ,C. Leckie, and K. Ramamohanarao, "Protection from Distributed Denial of Service Attacks Using History- based IP Filtering," June 2003, vol. 1, pp. 482 - 486

[8] J. Mirkovic and P. Reiher, "A Taxonomy of Intrudes Attack and Intrudes Defense Mechanisms," *Computer Journal of ACM IGCOMM*, vol. 4, no. 2, pp. 39-53, 2004.

[9] C. Douligeris and A. Mitrokotsa, "Intrudes attacks and defense mechanisms: Classification and state-of-the-art," *Telecommunications Networking*, vol. 44, no. 5, pp. 643– 666, Apr. 2004.

[10] J. Mirkovic, G. Prier, and P. L. Reiher, "Attacking Intrudes at the source," in *Proc. 10th IEEE Int. Conf. Network Protocols*, Sep. 2002, pp. 312–321.