ATLANTIS PRESS
Advances in Computer Science Research, volume 50
4th International Conference on Electrical & Electronics Engineering and Computer Science (ICEEECS 2016)

Application-Layer DDoS Detection by K-means Algorithm

Chuyu She [1,2,3, a], Wushao Wen [1,2,b,*], Kesong Zheng[1,c] and Yayun Lyu[1,d]

[1] School of Data and Computer Science, Sun Yat-Sen University,
Guangzhou 510006, PR China

[2] SYSU-CMU Shunde International Joint Research Institute,
Shunde 528300, PR China

[3] School of Mathematics and Statistics, Guangdong University of Finance & Economics,
Guangzhou 510320, PR China

[a] shechuyu@gdufe.edu.cn, [b]wenwsh@mail.sysu.edu.cn, [c]745379050@qq.com, [d]87912856@qq.com

*Corresponding author: Wushao Wen

Keywords: Application-layer DDoS attack, User behavior, Clustering methods, K-means.

Abstract. Lots of methods have been proposed to detect Distributed Denial-of–Service (DDoS) attacks focus on the transport layer and the network layer. However, these methods may not work well when application-layer DDoS attack is launched. In this paper, we introduce a clustering method based on some features to detect application-layer DDoS attack. Firstly, we extract features from normal users' sessions. Then, we cluster users' sessions by K-means algorithm and build normal users' behavior model. Finally, we detect the application-layer DDoS attack based on the normal users' behavior model.

Introduction

Distributed denial of service attack (DDoS) is a major threat in the Internet. There are many classical DDoS attack methods and tools. Traditionally, DDoS attacks such as ICMP flooding, SYN flooding and UDP flooding [1] are carried out at the network layer and transport layer. Lots of methods have been proposed to defend this serious security threaten. Statistical approaches [2] to DDoS attacks detection and mitigation involve packet attributes like source IP and destination IP address, time to live (TTL), and so on. These attributes are used to build kinds of defense mechanisms. Clustering Method can be also found to detect DDoS attacks focus on IP or TCP layers. Lee et al. [3] cluster IP addresses and TCP and UDP ports on backbone routers to find DDoS attacks. These methods are effective for network-layer DDoS attack detection. But they cannot protect against application-layer DDoS attack effectively. Application-layer DDoS attacks use true IP address. Attack agent simulates a normal web user and sends requests to application server. As we known, application server can handle limit requests in a short time. All attack agents cooperate to overload the application server. Application-layer DDoS can bypass defense methods which use IP or TCP header attributes' distribution by making full and successful TCP connections.

In this paper, we use k-means clustering method to detect application-layer DDoS attack. We select proper features of users' sessions, and cluster these normal user sessions to build normal users behavior models. There are many differences between normal users' sessions and attackers' sessions, like request rate, requested resources, and so on. As we have built the normal users behavior models, we can use these models to detect anomaly of behavior.

Copyright © 2016, the Authors. Published by Atlantis Press.
This is an open access article under the CC BY-NC license (http://creativecommons.org/licenses/by-nc/4.0/).

Model and Algorithm

In this session, we introduce a clustering method to detect App-DDoS. Clustering methods need to select proper features first, so we introduce the detail as follows.

User Browsing Process. Usually, a web user visits a website by a browser, like chrome, IE. Firstly, user inputs a url or clicks a hyperlink in browser windows. And then browser resolves the URL to get web server's IP, port, protocol information, URI and some necessary information. At last, browser establishes a TCP connection with the web server, and sends the HTTP request from user to web server. Browser receives server's reply and shows the content to the user. If the content contains pictures or other resources needed to show, browser will automatically request these resources for user. If user is interested in one hyperlink, he can click it and then wait, like the process above.

Feature Selection. For a user, his browsing behavior is a request sequence. Let r_i denote request i in a request sequence. And let t_i denote request i timestamp. In Fig. 1, if t_i-t_{i-1}<1800, r_i and r_{i-1} are in the same session, otherwise in separate session [4].

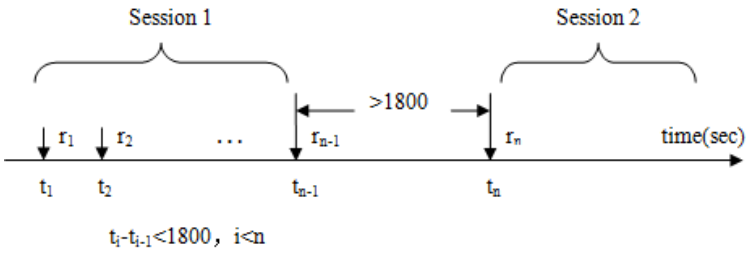

Fig. 1. Session definition

According to the definition above, we consider Su = {IPu, <r_1, t_1>,..., <r_n, t_n>} as a session describing the interaction between user *u* and web server.

There are many differences between normal users' session and attacks' session, so we select some features to build users' behavior models.

(1) The total number of request in a session. For HTTP flood DDoS attack, it's very important that a session should have enough requests for attack purpose. So we consider this feature as a very important feature for detection of this kind of Application layer DDoS attack.

(2) The total size of all requests in a session. An attacker can just request large resources which a simple request may cost a large volume of bandwidth, for example requesting a large video resource. So the total size of all requests in a session is a very important feature for detection of attack focusing on large resources.

(3) The request rate of the session. Obviously, if the request rate of the session is high, it is abnormal. So this feature is proper to detect attack.

(4) The average access frequency of the request in a session. In a website, webpages have different access frequency. Jung et al. [5] said that 10% webpages may account for approximately 80%-90% of requests. That is most webpages may have low access frequency. When launch a random attack, the average access frequency of the request in the attack session is lower than the normal session.

K-means Algorithm. K-means clusters objects into k groups by assigning all objects to the closest centroid. Generally, initial centroids are selected randomly, and use the minimum within-cluster sum of squares to judge the selection. Since clusters from k-means are spheres, a centroid and a radius could be enough to characterize a cluster. This makes our normal users' model simpler [6]. Algorithm 1 shows that normal session clusters are built by K-means clustering method.

Algorithm 1 cluster sessions

1. **Input:**
2. Sessions
3. K
4. **Output:** clusters
5. **Method:**
6. Step1: extract features from the set of sessions.
7. Step2: normalize feature vectors.
8. Step3: use k-means clustering algorithm to get k clusters:
9. select k points as the initial centroids
10. **repeat**
11. from k clusters by assigning all points to the closest centroid
12. recompute the centroid of each cluster
13. **until** the centroids don't change

We use K-means clustering method to cluster normal session and build normal users' behavior model. And then when a new session coming, system calculates whether the session is in a normal session cluster. If the session is found to deviate from all the normal clusters, the session will be recorded as abnormal.

Numerical Results

To validate our defense method, we used the web-log of a real website. The log was collected from the website of Sun Yat-sen University. It has 20978 IP, 1,281,876 requests through the whole day. We get requests from the HTTP request queue and add the requests to the corresponding user's session. And then, we sort the sessions according to the ascending order of the length and observe the distribution of session length. Here, session length means that the number of the requests in a session. Fig.2 shows the distribution of session length. As we can see in Fig.2, most of session's length is shorter than 200.

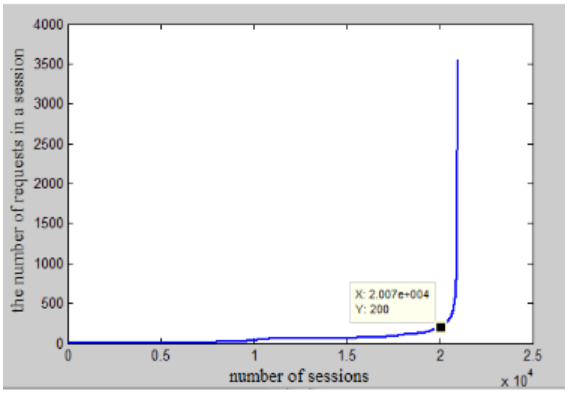

Fig. 2. The number of requests in sessions

The normal dataset last from 0s to 86382s. We train the normal data from 0s to 40000s by K-means algorithm. And then we launch a random attack from 40000s to 55000s. So attackers' requests are mixed with the normal users' requests from 40000s to 55000s. And then the system detects attacks. If a session is detected as abnormal, the system adds this user to blacklist and block the user's request. Table 1 is the detection results of dataset. Where k is input parameter of k-means and d_{len} means that the number of requests used to detect in a session. When d_{len} = 40 and k= 9, the detection effect is excellent. Its detection rate is high and false negative is low.

Table 1. Results of dataset when d_{len} is 40

K	Detection rate	False positive
6	75.60%	0.19%
7	82.92%	0.81%
8	90.24%	2.06%
9	97.56%	2.67%
10	98.37%	5.17%

Summary

In this paper, we proposed an application-layer DDoS detection method based on K-means algorithm. To build user behavior model, we extract features from users' sessions and cluster these sessions by K-means clustering method. And then, we use the model to detect anomaly of user behavior. Numerical result demonstrates that our detected method is effective.

Acknowledgements

This work was supported by the Science and Technology Project of Guangdong province (2014B010114002，2015B010108004)

References

[1] Y. Xie and S. Z. Yu, Monitoring the Application-Layer DDoS Attacks for Popular Websites, IEEE/ACM Transactions on Networking, Sci. Vol.17, No. 1,(2009), p. 15-25.

[2] F. Simmross-Wattenberg et al., Anomaly Detection in Network Traffic Based on Statistical Inference and α-Stable Modeling, IEEE Transactions on Dependable and Secure Computing, Sci. vol. 8, no. 4, (2011), p. 494-509.

[3] K. Lee ,J. Kim, K. H. Kwon et al., DDoS attack detection method using cluster analysis, Expert Systems with Applications, Sci. vol. 34, no. 3, (2008), p.1659-1665.

[4] R. Cooley, B. Mobasher, J. Srivastava, Data preparation for mining world wide web browsing patterns, Knowledge and information systems, 1(1): (1999), p.5-32.

[5] J. Jung, B. Krishnamurthy and M. Rabinovich, Flash crowds and denial of service attacks: Characterization and implications for CDNs and websites, in Proc. The 11th IEEE International World Wide Web Conference, Honolulu, Ha-waii, USA, ACM, (2002), p.252-262.

[6] J. Han, M. Kamber, Data Mining Concepts and Techniques, Burnington: Morgan Kaufmann, (2006).p.251-351.