ATLANTIS PRESS

Advances in Intelligent Systems Research, volume 133

2nd International Conference on Artificial Intelligence and Industrial Engineering (AIIE2016)

Detection of Application-Layer DDoS by Clustering Algorithm

Chuyu She[1,2,3], Wushao Wen[1,2,*], Zaihua Lin[1] and Kesong Zheng[1]

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

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

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

[*]Corresponding author

***Abstract*—Affinity Propagation (AP) algorithm is a relatively new clustering algorithm that can handle large datasets to obtain more satisfactory results. This paper introduces a detection mechanism for application-layer DDoS attack by using AP algorithm. In this detection strategy, we first extract some features from normal users' sessions. Then, we cluster these normal users' sessions by AP algorithm to get K clusters. Finally, we use these models to detect application-layer DDoS attacks.**

Keywords-application-layer; DDoS attack; affinity propagation; clustering algorithm; features

I. INTRODUCTION

Distributed denial of service attack (DDoS) is a major security problem for the Internet. Distributed denial of service showed its power since 1996. And after that many types of DDoS methods were developed. There are many classical DDoS attack methods and tools. Historically, Ping of death (POD) is a very famous Dos attack method. And then another kind of ping attack appeared—ping flooding. ICMP flooding, SYN flooding and UDP flooding [1] are also carried out at the network layer and transport layer. Many methods have been proposed to defend from this serious security threaten.

Statistical approaches [2] to DDoS attacks detection involve packet attributes like source IP and destination IP address, time to live (TTL), and so on. These methods often assume network traffic characteristics distribution will change when DDoS attack happens. Actually, many methods based on statistic are effective for network-layer and transport-layer DDoS attack. But they may not effective for application-layer DDoS attack. Clustering methods can be also found to detect DDoS attacks by cluster IP addresses and TCP ports on backbone routers [3, 4]. However, they may not effective for application-layer DDoS attack as well.

Application-layer DDoS attack uses true IP address, and simulates a normal web user to send requests to application server. So, it can bypass defense methods based on detection of spoofed IP or TCP header attributes' distribution. And application-layer can use many attack strategies to evade volume detection schemes.

In this paper, we use Affinity Propagation (AP) algorithm to cluster normal users' sessions and then get the k clusters. AP algorithm is a relatively new clustering algorithm that has been introduced by Frey and Dueck [5]. It can handle large datasets in a relatively short period to obtain more satisfactory results. Unlike clustering algorithms such as k-means or k-medoids, AP does not require the number of clusters to be determined or estimated before running the algorithm. So we use AP algorithm to get k clusters. And then we use k-means to detect application-layer DDoS attacks.

II. CLUSTERING ALGORITHM

A. Normal Sessions

A user's browsing behavior is a request sequence. A session is a request sequence by a user, and a user can initiate several sessions. For a user, two consecutive requests are less than 1800 seconds away are treated as in the same session [6]. We get users' sessions by Algorithm 1.

Algorithm 1 generate users' sessions

```
Input:
    Request queue of web server
Output:
    Users' sessions
Method:
    repeat
        Get a request R from queue
        Add this request to sessions[R.IP]
    until
        Queue is empty
```

B. Select Features

There are many differences between normal users' session and attacks' session, so we select some features from the sessions.

- The duration of a session. The duration of a session means that the time from the first request to the last request in a session. Generally, normal users won't stay in a website for a long time. If a user stay too long in a website, it may be an attacker. Because an attacker may take a long time to achieve attack effect.
- The statistical popularity of requests in a session. On a website, webpages have different popularity. Study [7] said that 10% webpages may account for approximately 90% of requests. That is most webpages may have low popularity. When launch a random attack, the average popularity of the request in the attack session is lower than the normal session.

- The statistical transition probability in a session. The transition probability between two requests is different. If a request sequence is randomly generated, its transition probability is lower than normal sessions'. So this feature is useful to detect attack.

C. AP Algorithm

We use AP clustering algorithm to cluster normal sessions. AP is a clustering algorithm based on the concept of "message passing" between data points. Unlike clustering algorithms such as k-means or k-medoids, AP does not require the number of clusters to be determined or estimated before running the algorithm. Similar to k-medoids, AP finds "exemplars", members of the input set that are representative of clusters [5].

AP algorithm takes each data point as the candidate exemplar and calculates the similarity between any two sample points.

Let x_1 through x_n be the set of data points, and let s be a function that quantifies the similarity between any two points, such that s(i,j) > s(i,k) if x_i is more similar to x_j than to x_k. In this paper, we use Euclidean distance as in (1) to measure dissimilarity of data points, and get the similarity matrix S.

$$s(i, j) = -d^2(x_i, x_j) = -\|x_i - x_j\|_2^2 \qquad (1)$$

The algorithm proceeds by alternating two message passing steps, to update two matrices [8]:

- The "responsibility" matrix R has values r(i,k) that quantify how well-suited x_k is to serve as the exemplar for x_i, relative to other candidate exemplars for x_i.
- The "availability" matrix A contains values a(i,k) represents how "appropriate" it would be for x_i to pick x_k as its exemplar, taking into account other objects' preference for x_k as an exemplar.

  Both matrices are initialized to all zeroes, and can be viewed as log probability tables. The algorithm then performs the following updates iteratively:

First, responsibility updates are sent around in (2):

$$r(i, k) \leftarrow s(i, k) - \max_{k' \neq k}\{a(i, k') + s(i, k')\} \qquad (2)$$

Then, availability is updated in (3) for $i \neq k$.

$$a(i, k) \leftarrow \min\left(0, r(k, k) + \sum_{i' \notin \{i,k\}} \max(0, r(i', k))\right) \qquad (3)$$

For i=k, availability is updated in (4):

$$a(i, k) \leftarrow \sum_{i' \neq k} \max(0, r(i', k)) \qquad (4)$$

To avoid the numerical oscillation, the damping factor λ is introduced in (5) and (6).

$$R_i = (1 - \lambda)R_i + \lambda R_{i-1} \qquad (5)$$

$$A_i = (1 - \lambda)A_i + \lambda A_{i-1} \qquad (6)$$

Algorithm 2 shows that normal session clusters are built by AP clustering method based on the selected features. We extract features from the set of sessions firstly, and then we normalize the feature vectors. Finally, we use AP clustering method to get the clustering result.

Algorithm 2 cluster sessions

Input:
Sessions
Output:
K clusters
Method:
(1) Extract features from the set of sessions.
(2) Normalize feature vectors.
(3) Use AP algorithm to get clustering result:
Compute the similarity matrix S.
Initialize the R and the A matrix.
Repeat
Update the responsibilities
Update the availabilities
until a fixed number of iterations
Or the changes fall below a threshold.

D. Detection Process

When AP clustering method output the K clusters, we use the parameter K as the input of the K-means algorithm. And then we build the normal users' behavior models and detect attack. Detection algorithm calculate whether the current 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, and the corresponding user IP will be added to blacklist.

III. EXPERIMENTS

In our experiment, we use a web-log of a university website. We get requests from the request queue and collect user's sessions. And then we cluster the normal sessions by the three features: the duration of a session, the popularity of requests and the transition probability in a session. As shown on Figure I, it is clearly that normal users' behaviors are very similar in these three dimension.

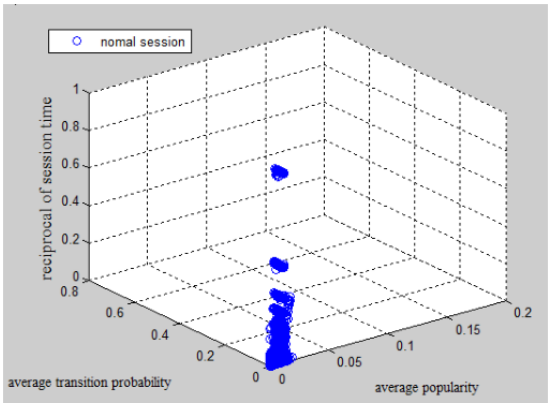

FIGURE I. NORMAL SESSION FEATURE IN 3D

We launch a random attack on the website, and then the system detects attacks. Figure II is the Receiver Operating Characteristics (ROC) curves that shows the performance of our detection model on application-layer DDoS attack.

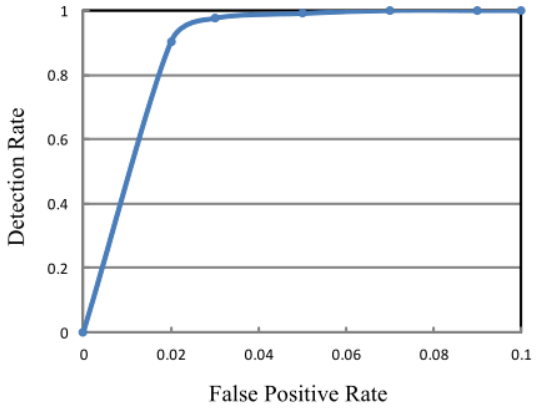

FIGURE II. ROC OF DETECTION MODEL

IV. CONCLUSION

This paper proposed an application-layer DDoS detection method based on clustering method. To build user behavior model, we extract features from users' sessions and cluster these sessions by AP clustering method. And then, we use the model to detect application-layer DDoS attack.

ACKNOWLEDGMENT

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, pp. 15-25, 2009.

[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, pp. 494-509, 2011.

[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, pp.1659-1665, 2008.

[4] M.I.W. Pramana, Y. Purwanto, F.Y. Suratman. "DDoS Detection Using Modified K-Means Clustering with Chain Initialization Over Land mark Window." Proc of International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC) pp:7-11,2015.

[5] B.J. Frey and D. Dueck. "Clustering by passing messages between data points". Science, 315 (5814), pp. 972-976, 2007.

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

[7] 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, pp.252-262, 2002.

[8] C. Lu, S.J. Song and C. Wu. "K-Nearest Neighbor Intervals Based AP Clustering Algorithm for Large Incomplete Data." Mathematical Problems in Engineering, Volume 2015, Article ID 535932, 9 pages, 2015FIGURE II ROC OF DETECTION MODEL