DoS Attack Pattern Mining Based on Association Rule Approach for Web Server

Hsing-Chung Chen[1,2(✉)] and Shyi-Shiun Kuo[1,3]

[1] Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan
shin8409@ms6.hinet.net, cdma2000@asia.edu.tw
[2] Department of Medical Research, China Medical University Hospital, China Medical University, Taichung, Taiwan
[3] Department of Multimedia Animation and Application, Nan Kai University of Technology, Nantou County, Taiwan

Abstract. In recent years, lots of web servers increasingly often suffer from Denial of Service (DoS) attacks within application layer. Many approaches provide abnormal traffic detecting in order to prevent any malicious traffic. However, the attack features or patens of the malicious traffics did not addressed clearly. Thus, the aim of this paper is to provide an approach based on the association rule mining technique for traffics appeared in the integrated web services, such as HTTP, HTTPS, and FTP traffic, in order to discover the strong attack features or patens of DoS attacks. Association rule mining is employed in this paper to deal with the DoS patens and then find out the strong relations among features of DoS attacks in large well-known dataset, e.g. NSL-KDD. The strong relations which are determined on when the major attack features are discovered from the open dataset would be considered as the strong patterns of DoS attacks. Finally, the outputted strong patterns could be used in the intrusion detection system (IDS) to enhance the effects of detecting application layer DoS attacks.

1 Introduction

Because of the popularization of the Internet applications, more and more commercial services are built on the Internet. The threats on the network are becoming more frequent and complicated. In recent year, the increasing amount of cyber-attacks has caused unavailability of website, even considerable commercial losses. There have been a variety of threats emerged, in the past years, due to the vulnerability of original design of the Internet. One of the most common threats is Denial of Service (DoS) attacks. Moreover, the traffic volume caused by modern DoS attacks have been more than 500 Gbps [1] and have become a major threat to network security [2], in the recent years. Typically, DoS attack is intended to disrupt the availability of web service to legitimate users by exhausting the network or computing resources of their victims. The kind of attack trying to exhaust the network resources, such as bandwidth or router processing capacity, could be referred to a network layer flooding attack. On the other hand, the kind of attack trying to exhaust the computing resources, such as sockets,

L. Barolli et al. (Eds.): IMIS 2018, AISC 773, pp. 527–536, 2019.
https://doi.org/10.1007/978-3-319-93554-6_50

CPU time, memory, and disk bandwidth, is known as an application layer flooding attack. The network layer flooding attack disrupts the legitimate user's connectivity, and the application layer flooding attack disrupts the legitimate user's service requests. An attacker may launch one or both attacks to block the victim's services. Due to the vulnerabilities of application layer protocol, application layer, DoS attacks are able to cause a same level of impact but much lower cost, if they are compared to network layer DoS attacks. Application layer DoS attack is a rapidly growing category of network threat in past few years. The duration, frequency and size of attacks are represented to be increasing every year [3]. Although a lot of literatures had been designed on the detection and prevention of network layer DoS attack, the researchers focused on detection and prevention of application layer DoS attack is more urgent and important for modern network.

Because of the attack packets are similar to legitimate traffic, the attacks are difficult to detect. Nevertheless, there are many features could be observed in the network traffics. Each type of attack may represent strong relationship among features, that is, some feature has high intensity is possibly related to another high intensity feature. Thus, if a relationship among features is identified, the relationship could be considered as a rule to filter the attack. Therefore, this paper presents an approach to discover the strong attack pattern of application layer DoS attack by using association rule mining technique [4]. Association rule technique is intended to mine the potential relationship among features based on two measured metrics consisting of support and confidence. The features of network traffic are evaluated on the well-known dataset known as NSL-KDD [5] which consists of the 41 features of network traffics. The NSL-KDD dataset is a refined version of its predecessor KDDcup'99 excluding redundant instances [6]. It includes two main classes of network traffic. One is t is normal class and another one is attack class. The attack class is further classified into four categories: DoS, Probing, users-to-root (U2R), and remote-to-local (R2L) [6]. The DoS attack disrupts the availability of victim's service to make it unable to handle legitimate requests. The DoS category includes different types of attacks, such as Apache2, Back, Land, Neptune, and Smurf [7]. The probing attack attempts to scan the network to gather information about the victim in order to find its vulnerabilities. The probing category includes attacks issued by Satan, Ipsweep, Nmap, Portsweep, Mscan, and Saint [7]. The U2R attacker uses a normal user's account to exploit the vulnerabilities of the victim's system by trying access to the root of the system. The examples of U2R attack include Buffer_overflow, Loadmodule, Rootkit, Sqlattack. In R2L attack, an attacker intrudes into a remote machine and gain local access to it. The R2L category includes the attacks such as Guess_Password, Ftp_write, Imap, Multihop [7].

Due to its high rank among various types of attack, the application layer DoS attacks to web servers are focused to explore their attack patterns. Therefore, the three attack types, Apache2, Back, and Neptune, are considered in this paper, which they are the popular application layer flooding attacks issued toward a web server. These three attack types are illustrated in Table 1.

The rest of this paper is organized as follows. The related work for the detection of application layer DoS attacks is described in Sect. 2. Section 3 presents the proposed approach which consists of two phases. In first phase, the necessary features are selected from the 41 features of NSL-KDD in which represents application layer attack.

In second phase, association rule mining method is applied to discover strong relationships among those selected features. Section 4, features selection is discussed. Finally, the conclusions are drawn in Sect. 5.

Table 1. The description of application layer DoS attack types.

Attack types	Description
Apache2	An attacker sends requests with many http headers to an apache web server. The server will slow down and may eventually crash if it receives too many this type of requests
Back	An attacker issues requests with URL containing many front-slashes. The server tries to process these requests and it will slow down and becomes unable to process other requests
Neptune	An attacker generates a large number of SYN packets to a target sever to request session establishment. The server waits for session establishment and its buffer will exhaust

2 Related Work

Detection of DoS attacks is a challenging issue because it has high rank among various types of attacks. Especially, the increasing resources and techniques available to attackers in modern network, the detection of DoS is more difficult in the application level. In recent years, many approaches [8–11] have been proposed to detect anomaly on the web based services.

The defense mechanisms against DoS flooding attacks in application level could be classified into two categories: server side and distributed mechanisms [8]. The classification is based on the deployment location of defense mechanisms. The server side application level DoS attacks are focused in this paper. The application layer DoS flooding attacks usually target the specific characteristics of application services, such as HTTP, DNS, and SIP. In recent year, the HTTP based, i.e., web based, DoS flooding attacks have become the major threats in the Internet. Ajagekar and Jadhav [9] propose a classifier based system to extract the important fields from captured application layer packets, and then to apply classifier to detect attacks. The aim of this method is to improve the accuracy of DDoS attack detection at application layer. Adi *et al*. [10] conduct DoS attack traffic models against HTTP/2 services and present a novel and stealthy DoS attack variant. Then, they conduct the attack traffic analysis which employs four machine learning techniques, namely Naïve Bayes, decision tree, JRip and support vector machines, to show the stealthy attacks through higher percentage of false alarms. Jazi *et al*. [11] focused on the impact of sampling techniques on detection of application layer DoS attacks. They demonstrated that the most sampling techniques introduce significant distortion in the traffic. This type of distortion minimizes the ability of a detection algorithm to capture the traces of stealthy attacks.

A number of studies discussed the detection strategies for application layer DoS attacks. There are few researches focus on the pattern of DoS attack. A pattern consists of several essential features captured from a network connection, which could be represented the characteristics of the connection. The same type of application layer DoS attacks has probably similar patterns. If the pattern of a certain type of attack is determined, the attack connection will be identified. In this paper, the association rule mining technique will be employed to explore the relationships among features of application layer DoS attacks.

3 Proposed Approach

This paper attempts to find possible patterns of application layer DoS attacks by analyzing the features in NSL-KDD dataset. When an attack occurs, some features' values arise significantly in contrast with those values in normal connections. For example, when an attack of Apache2 occurs, the time duration of a connection becomes high and the number of bytes transferred from source to destination also turns into high, thus, the duration and the number of bytes have an association with each other. This feature association may represents one pattern of Apache2 attack, which could be used as a rule to identify this type of attack. In this work, there are three phases to complete the feature association rule mining.

Phase 1: *Feature selection*. It is not necessary to mine the association rules with all features in dataset, since there is no correlation between the number of features and attack detection rate. The essential features will be selected in this phase which is explained in Subsect. 3.1.

Phase 2: *Feature value normalization*. There are two types of feature value in the dataset: continue and discrete. The type of 'continue' is an integer or real number. The type of 'discrete' is an enumerated value in a finite set. The association rule mining technique works only on discrete type of values. Thus, the feature values of continue type need to be converted into discrete type, then they will be normalized into a set of finite numbers. First, all the instances of attack types listed in Table 1 are screened out as a dataset for mining. Then, the values of each feature in this attack dataset will be normalized. The normalization is described in Subsect. 3.2.

Phase 3: *Feature association rule mining*. The relationships of the selected features will be discovered in this phase by using association rule mining method. After applying association rule mining method on the attack dataset, the resulting rules will be listed in descending order depending on their confidence metric. The details of this phase is illustrated in Subsect. 3.3.

3.1 Features Selection

Each record in NSL-KDD dataset represents the instance of a network connection, and each of which consists of 41 features. All features are grouped into three categories: basic features, content related features, and traffic related features [7, 12]. There are 9 features in basic category, 13 features in content related category, and 19 features in traffic category, respectively. In this work, the application layer DoS attack traffic is focused. The traffic related features have intrinsic relation to DoS attack. The basic features contain the basic information of network connection, such as duration, protocol type, number of data bytes transferred between client and host. The content related features have strong relation to the R2L and U2R attacks, e.g., number of login attempts, number of root accesses. They have weak relation to DoS traffic. Therefore, the basic features and traffic related features are selected to be analyzed the difference between normal connections and DoS attacks.

There is no direct correlation between the number of features and attack detection rate, that is, more number of features not necessarily has higher attack detection rate. This irrelevance has been proven in some researches [12–14]. Table 2 lists the detection rate with regard to the number of features. The classification algorithms, such as Adaboost and Naïve Bayes, are employed to select the essential features.

Table 2. The performance of DoS attack detection [12, 13].

References	Number of features	Percentage of detection rate
Yi and Phyu [12]	41	95
	7	100
Natesan and Balasubramanie [13]	41	97.8
	15	98.9

The goal of this work aims to discover the association rules among features to identify the DoS attack patterns. According to the proven results in Table 2, it is not necessary to select all 41 features to mine their relationships. If too many features are selected, there will be too many association rules to be discovered and the result could not enhance the performance of attack detection. On the other hand, the attack patterns could not be comprehensively mined with too less features, and the discovered association rules will be too less and simple that could not identify the real attacks and may cause a large number of false alarms. Therefore, after carefully examining on 28 features in both basic and traffic related categories as well as referring to the essential features mentioned in [12–14], nine features are selected in this work for further mining their relationships. The descriptions of these selected features are listed in Table 3.

Table 3. The selected features and their descriptions.

Selected features	Label	Value types	Category	Description
duration	f_1	Continue	Basic	Time duration of the connection in seconds
src_bytes	f_2	Continue	Basic	Number of data bytes from source to destination in the connection
count	f_3	Continue	Traffic-related	Number of connections to the same destination host as the current connection in the past two seconds
dst_host_count	f_4	Continue	Traffic-related	Number of connections that have the same destination host IP address in the past two seconds
service	f_5	Discrete	Basic	Network service on the destination host, e.g., HTTP, ftp_data
flag	f_6	Discrete	Basic	Status of the connection representing normal or error
srv_serror_rate	f_7	Continue	Traffic-related	Percentage of connections that have 'SYN' errors to the same service
srv_rerror_rate	f_8	Continue	Traffic-related	Percentage of connections that have 'REJ' errors to the same service
dst_host_srv_rerror_rate	f_9	Continue	Traffic-related	Percentage of connections that have 'SYN' errors from same service to the destination host

3.2 Feature Value Normalization

In order to discovery the feature association rules of attacks listed in Table 1, all the instances of these attack types are screened out as an attack dataset for mining. Then, the values of each feature in this attack dataset will be normalized. Since the association rule mining technique works only on discrete type of values, the feature values of continue type need to be converted into discrete type. The values of each feature is normalized as the values between [0, 1] with fixed decimal precision as defined in Eq. (1).

$$R\left(\frac{f_{ij}}{\max f_i}, d_i\right), \quad i = 2, 3, 4, \tag{1}$$

where R is a rounding function, f_{ij} is the j-th instance value of the i-th feature, and d_i is the precision of decimals. The precision d_i will restrict the maximum cardinal number of a finite set. If $d_i = 2$, the f_i is normalized into [0, 1] with 2 decimals precision in which there are at most 101 cardinalities in this feature. The continue type of feature is

converted to discrete type with a finite set by using Eq. (1). In addition, each feature could have its own precision by determining d_i.

For the duration feature (f_1), after observing all instances in dataset, the feature values almost are larger than 200 s in attack instances. Thus, f_1 is normalized into binary type of $\{0, 1\}$ by setting a threshold (t) of 200. For features f_5 and f_6, they are intrinsic discrete type with a finite set. For other features f_7, f_8, and f_9, they are already in two decimals precision of percentage. For example, Table 4 is the 10 attack instances of NSL-KDDTest+ dataset, and Table 5 is the normalization version of Table 4.

Table 4. Ten attack instances of NSL-KDDTest+ dataset with selected features as an example.

f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9
902	57964	42	241	http	RSTR	0.02	0.95	0.16
0	0	111	255	http_443	REJ	0	1	1
2061	72564	11	255	http	RSTR	0	1	0.3
904	75484	21	255	http	RSTR	0.05	0.9	0.07
781	79864	1	230	http	RSTR	0	1	0.02
0	0	133	255	ftp_data	S0	1	0	0
0	0	76	255	http	S0	1	0	0.63
2068	93004	7	255	http	RSTR	0	0.86	0.03
2100	68184	8	255	http	RSTR	0	0.88	0.03
2069	101764	3	255	http	RSTR	0	0.67	0.01

Table 5. Feature value normalization with precision of two decimals.

f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9
1	0.57	0.32	0.95	http	RSTR	0.02	0.95	0.16
0	0	0.83	1	http_443	REJ	0	1	1
1	0.71	0.08	1	http	RSTR	0	1	0.3
1	0.74	0.16	1	http	RSTR	0.05	0.9	0.07
1	0.78	0.01	0.9	http	RSTR	0	1	0.02
0	0	1	1	ftp_data	S0	1	0	0
0	0	0.57	1	http	S0	1	0	0.63
1	0.91	0.05	1	http	RSTR	0	0.86	0.03
1	0.67	0.06	1	http	RSTR	0	0.88	0.03
1	1	0.02	1	http	RSTR	0	0.67	0.01

The procedure for feature normalization is described as follows.

Procedure 1. Normalization.

Input: Original dataset $F=\{f_{ij}\}$ *with features selected in Phase 1.*

Output: Normalized dataset $F'=\{f'_{ij}\}$.

Step 1: If $f_{1j} < t$ *then* $f'_{1j} = 0$ *else* $f'_{1j} = 1$, *for every instance j. Let t be a predefined threshold of time duration.*

Step 2: $f'_{ij}=R\left(\frac{f_{ij}}{\max f_i}, d_i\right)$, *for i = 2, 3, 4, and for every instance j. Let* d_i *be a specific precision of decimals.*

Step 3: $f'_{ij} = f_{ij}$, *for i = 5, …, 9, and for every instance j.*

3.3 Feature Association Rule Mining

Let X and Y be the sets of features, the implication $X \Rightarrow Y$ is called an association rule which represents the relationship of X and Y. There are two measures for association rules, i.e., support and confidence, which indicate the strength of the implication $X \Rightarrow Y$. They are defined as Eqs. (2) and (3), respectively. The support of the association rule $X \Rightarrow Y$ is the probability of $X \cup Y$ in dataset. An association rule could be regarded as an strong attack pattern.

$$support(X \Rightarrow Y) = P(X \cup Y) \tag{2}$$

$$confidence(X \Rightarrow Y) = \frac{support(X \cup Y)}{support(X)} = \frac{P(X \cup Y)}{P(X)} \tag{3}$$

Table 6. The top 10 association rules with minimum support of 0.4 and minimum confidence of 0.8.

Association rules	Confidence
{flag=RSTR} ⇒ {service=http}	1
{duration=0, service=http} ⇒ {dst_host_count=1}	0.9985
{duration=0} ⇒ {dst_host_count=1}	0.9975
{service=http} ⇒ {dst_host_count=1}	0.9796
{srv_serror_rate=0} ⇒ {dst_host_count=1}	0.9669
{srv_serror_rate=0, service=http} ⇒ {dst_host_count=1}	0.9642
{flag=RSTR, service=http} ⇒ {dst_host_count=1}	0.9568
{flag=RSTR} ⇒ {dst_host_count=1}	0.9568
{flag=RSTR, service=http} ⇒ {duration=1}	0.906
{flag=RSTR} ⇒ {duration=1}	0.906

In order to discover the appropriate strong rules, the minimum support and the minimum confidence should be determined before association rule mining. A famous algorithm implementing association rule mining is Apriori [15], which is used in this work to mine the association rules in the attack dataset. After applying Apriori algorithm on attack dataset with selected features, top 10 association rules based on confidence value are picked up. In this paper, let the support value be 0.4 and the confidence value be 0.8, resulting rules after Apriori execution are shown in Table 6.

For more explanation, consider rule 1 and rule 2. Rule 1 indicates that if a connection have been established but aborted by responder (RSTR), and also this connection uses http service, then this connection has high confidence to be regarded as an application layer DoS strong attack pattern. Rule 2 shows that if a connection uses http service and its time duration less than threshold (t = 200 s), but also the number of connections like this one having the same destination IP address (dst_host_count) is almost the maximum number (255 in the attack dataset), then rule 2 has high confidence to be regarded as an strong attack pattern.

The generated association rules will be used in an IDS to enhance its attack detection rate.

4 Discussions

The strong attack pattern is discovered based on the association rule mining method. In this paper, there are some parameters could be adjusted to adapt the strength and precision of association rules. For phase 1, the different set of selected feature will result in different set of strong attack patterns which will lead the IDS to make an appropriate decision. In phase 2, the normalization scheme will affect the complexity of generating combined feature-set. For the time duration feature, different attack type may have different threshold. For continue type features, the normalization function R in Eq. (1) could be replaced by any other normalization method which is able to significantly discriminate attack instances and normal ones. For phase 3, the minimum support and minimum confidence are thresholds to decide the precision and number of association rules. In general, if too many feature selected or too high cardinality of a normalized feature, there will generate too many association rules and may cause a large number of false alarms in IDS.

5 Conclusions

The aim of this paper is to introduce an approach based on association rule mining for web server to discover the pattern of application layer DoS attacks. The features of network traffic are evaluated on NSL-KDD dataset. The proposed approach could be adaptable to any dataset, which could rely on the characteristics of each feature to adjust the parameters for features value normalization. The adaptability of the proposed approach could generate more appropriate association rules. To improve the effects of association rule mining, the machine learning techniques, such as particle swarm optimization and deep learning, could be employed for enhancing the tool of the association rule mining.

Acknowledgement. This work was supported by the Ministry of Science and Technology (MOST), Taiwan, Republic of China, under Grant MOST 106-2632-E-468-003.

References

1. Akamai: State of the Internet Security Report (2017). https://www.akamai.com/
2. Ponemon Institute: Global Report on the Cost of Cyber Crime. HP Enterprise Security (2014)
3. Arbor Networks: Worldwide Infrastructure Security Report (2016). https://www.arbornetworks.com/
4. Agrawal, R., Imielinski, T., Swami, A.: Mining association rules between sets of items in large databases. In: Proceedings of 1993 ACM SIGMOD International Conference on Management of Data, Washington, DC, pp. 207–216 (1993)
5. Canadian Institute for Cybersecurity: NSL-KDD. University of New Brunswick. http://www.unb.ca/cic/datasets/
6. Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.: A detailed analysis of the KDD CUP 99 data set. In: Proceedings of IEEE Symposium on Computational Intelligence for Security and Defense Applications (2009)
7. Dhanabal, L., Shantharajah, S.P.: A study on NSL-KDD dataset for intrusion detection system based on classification algorithms. Int. J. Adv. Res. Comput. Commun. Eng. **4**(6), 446–452 (2015)
8. Zargar, S.T., Joshi, J., Tipper, D.: A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks. IEEE Commun. Surv. Tutor. **15**(4), 2046–2069 (2013)
9. Ajagekar, S.K., Jadhav, V.: Study on web DDOS attacks detection using multinomial classifier. In: Proceedings of 2016 IEEE International Conference on Computational Intelligence and Computing Research, pp. 1–5 (2016)
10. Adi, E., Baig, Z., Hingston, P.: Stealthy denial of service (DoS) attack modelling and detection for HTTP/2 services. J. Netw. Comput. Appl. **91**, 1–13 (2017)
11. Jazi, H.H., Gonzalez, H., Stakhanova, N., Ghorbani, A.A.: Detecting HTTP-based application layer DoS attacks on web servers in the presence of sampling. Comput. Netw. **121**, 25–36 (2017)
12. Yi, M.A., Phyu, T.: Layering based network intrusion detection system to enhance network attacks detection. Int. J. Sci. Res. **2**(9), 499–508 (2013)
13. Natesan, P., Balasubramanie, P.: Multi stage filter using enhanced AdaBoost for network intrusion detection. Int. J. Netw. Secur. Appl. **4**(3), 121–135 (2012)
14. Khan, S., Gani, A., Wahab, A.W.A., Singh, P.K.: Feature selection of denial-of-service attacks using entropy and granular computing. Arab J. Sci. Eng. **43**(2), 499–508 (2017)
15. Agarwal, R., Srikant, R.: Fast algorithms for mining association rules. In: Proceedings of the 20th International Conference on Very Large Databases, vol. 1215, pp. 487–499 (1994)