Research on DDoS Attack Detection in Software Defined Network

Ma Zhao-hui[1+2] Zhao Gan-sen[1*] Li Wei-wen[2] Mo Ze-feng[3] Wang Xin-ming[1] Chen Bing-chuan[4+5] Lin Cheng-chuang[1]
1. School of Computer Science, South China Normal University, Guangzhou 510631, China
2.School of Information Science and Technology, Guangdong University of Foreign Studies, Guangzhou 510006, China
3.School of Mathematical Sciences， South China Normal University, Guangzhou 510631, China
4. School of Statistics and Mathematics , Guangdong University of Finance and Economics, Guangzhou 510320, China
5. Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012,China

***Abstract*—Software Defined Network(SDN) is a new network construction. But due to its construction, SDN is vulnerable to be attacked by Distributed Denial of Service (DDoS) attack. So it is important to detect DDoS attack in SDN network. This paper presents a DDoS detection scheme based on k-means algorithm in SDN environment. The establishment of this scheme is based on the two hypotheses that the daily network works normally most of the time, and there is a significant difference between the data characteristics of normal situation and abnormal situation. At the same time, these two hypotheses are also true to the daily network condition. After demonstrating the validity of k-means clustering algorithm, the paper proposes 5 flow table features that can be used to detect DDoS attacks. Finally, the DDoS detection scheme was tested by simulation experiment. The test results showed that the method proposed by the author could effectively detect DDoS, with an average success rate of 97.78%.**

***Keywords*—Software Defined Network; Distributed Denial of Service; k-means; attack detection**

With the continuous development of cloud computing, big data and Internet of things, network service providers are required to quickly allocate network resources and rapidly allocate network management authority. However, it is difficult for the traditional network to meet all those needs. For this reason, the Cleanslate research group of Stanford university proposed a new network architecture, that is, Software Defined Network (SDN). SDN divides the network into three levels, realizing the separation of data control and forwarding. At the control layer, SDN requires the control layer to open the standard interface so that it can realize scheduling through programming and realize the dream of programmable network. Network flexibility has been greatly improved.

The centralized control feature of SDN gives it a great advantage in controlling network flow, but the feature of SDN makes it more vulnerable to Distributed Denial of Service (DDoS) attack. Distributed denial of service attack is one of the ways to harm network security. The attack makes use of a large number of puppet machines to directly or indirectly send bogus data packets to the target host, so that the target host cannot provide services due to insufficient link bandwidth or shortage of hardware resources. Being easy to launch, DDoS attacks can be realized with a few simple commands, and they are very aggressive and difficult to detect. Therefore, DDoS attacks are one of the main hazards of SDN at present.

At present, the DDoS attack detection method based on SDN has made some progress. Cai Jiaye [1] et al. proposed to use packet-in data frames as monitoring data and detect DDoS attacks through calculating the sibson distance data. However, this feature selection can only identify the DDoS attacks of bogus IP, but it cannot solve the problem of bandwidth occupation. Xiao fu [2] et al. realized abnormal flow detection by selecting 5 features in SDN and using KNN classification algorithm. However, when the training sample is large enough, the detection rate is increased by using KNN algorithm, but the false alarm rate will also be increased. Wang Xiaorui [3] et al. proposed an improved SDN feature selection scheme based on the analysis of previous studies and implemented DDoS attack detection with BP neural network algorithm. The training time based on neural network algorithm is too long, and more samples cost longer time, which reduces the practicability of the theory. Mei Mengzhe [4] selected the source/destination IP address, the source port, and the packet size as parameters, and took conditional entropy as the feature to realize abnormal flow detection. The use of information entropy as a feature relies too much on decision function. Once the normal change of network breaks the limit of decision function, the decision function needs to be reformulated and its expansibility is insufficient.

Existing researches on the defense against DDoS attack at all levels of the SDN architecture have been carried out [5][6][7][8]. Although it has certain effects, there are still great deficiencies in detecting DDoS attacks. Therefore, it is of great significance to study how SDN detects DDoS attacks, improve SDN network environment and improve network stability. In view of the above problems, this paper proposes a detection method based on the characteristics of

*Corresponding Authors: Zhao Gan-sen gzhao@m.scnu.edu.cn

flow table, and realizes attack detection through k-means clustering algorithm.

The following arrangements are as follows: Section 1 introduces the background knowledge of SDN; Section 2 elaborates the detection method of DDoS attack based on k-means; Section 3 illustrates the prototype experiment and the analysis of experimental results systematically. Section 4 summarizes the full text and looks forward to the future research focus and development trend.

1. INTRODUCTION TO BACKGROUND KNOWLEDGE

A. *Software Defined Network*

SDN is a three-layer independent structure, namely application plane, control plane and data plane. The architecture of the software defined network is shown in figure 1.

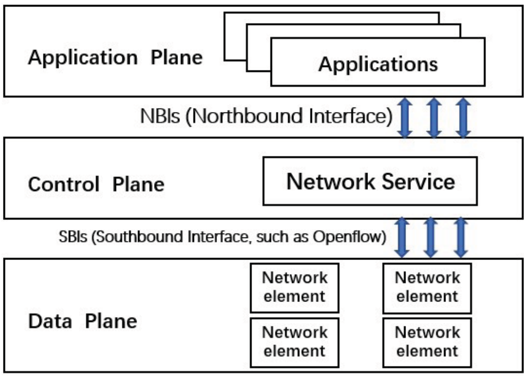


Figure 1. The Architecture of Software Defined Network

B. *Openflow Protocol*

Openflow is a standard of SDN south interface proposed by the ONF(Open Network Foundation). It is the communication protocol between the controller and the bottom switching equipment. The controller sends flow tables or any other messages to an Openflow switch through a transmission channel. Openflow switches match and forward each received packet according to the flow table items sent by the controller. Once the match fails, the Openflow switch will send the Packet_In message to the controller requesting the controller to issue the flow table rules. When the controller receives the Packet_In message, it will calculate the rules for the switch to forward the unknown flow by the established rules and will send the Packet_Out message to the switch. The basic architecture of Openflow is shown in figure 2.

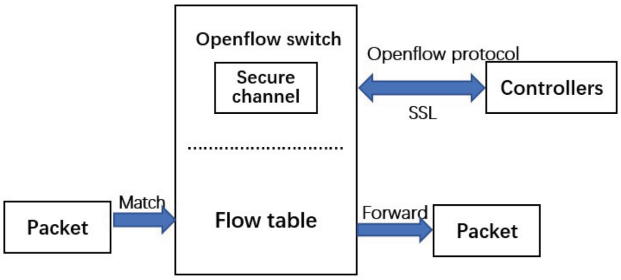


Figure 2. The Basic Architecture of Openflow

Currently, the Openflow version has been updated to 1.5.1,but the commonly used version is still 1.3.

C. *DDoS attack of SDN*

In the traditional network, DDoS can be divided into two types from the attack type. The first type is the resource consumption of the target host system, such as Syn flood attack and Ack flood attack; And the other consumes the resource of the flow bandwidth, such as UDP flood attack.

In the SDN architecture, the attack can be divided into three types according to targets of DDoS attack. The first is to take the controller as the target. By sending Packet_In messages to the controller frequently, the attack consumes the resources of controller, causing the controller to fail to work and therefore the entire network will be out of control. The second is to take the switch as the target. By sending garbage data packets, the switch flow table resources are occupied by the garbage flow table, making the switch unable to match and forward the normal data. The third is to take the network terminal in SDN as the target. It consumes its bandwidth resources or system resources, causing the target terminal to be down and unable to provide normal services.

D. *K-means Clustering Algorithm*

The k-means algorithm has a good effect on anomaly detection [9][10]. K-means algorithm is an unsupervised machine learning algorithm. Its ability to detect anomaly without distinguishing whether the data is abnormal is mainly based on the following two assumptions:

1) the normal flow behavior data in the training set is much larger than the abnormal flow behavior data;

2) there is a big difference in data between normal and abnormal behaviors.

The first hypothesis guarantees that the detection method can recognize normal and abnormal clusters, and the second ensures that the detection method can recognize normal and attack flows.

The k-means algorithm process is described as follows:

1) Randomly select k objects from dataset D as the initial Cluster center;

2) Calculate the distance from each object to each cluster center through the decided distance measurement method and assign each object to the nearest cluster;

3) Recalculate the clustering centers of each cluster;

4) Repeat the second and third steps until the clustering center no longer changes or reaches the given threshold value.

In k-means clustering, the similarity among each data objects is usually represented by the distance between objects. The commonly used distance calculation methods include Euclidean distance, Manhattan distance and Minkowski distance etc. Euclidean distance algorithm is adopted in this paper.

Euclidean distance is defined as follows:

Suppose there were two n-dimensional objects x（x_1,x_2, … ,x_n）, y （y_1,y_2, … ,y_n）, the distance from object x to y is calculated as formula (1) :

$$d(x,y) = \sqrt[2]{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots + (x_n - y_n)^2} \ldots\ldots （1）$$

In the clustering results, the following definitions are given to describe the difference between different clusters:

Suppose the training set s is divided into c= {c_1, c_2 , … , c_k}, the cluster c_i is the clustering center of the cluster. The average distance between the cluster and other clusters is used to represent the difference degree f(c_i) between the cluster and other clusters, and the formula is formula (2) :

$$f(c_i) = \frac{\sum_{j=1}^{k} d(c_j, c_i)}{k-1} \ldots\ldots\ldots\ldots\ldots\ldots\ldots （2）$$

According to the formula, the greater f(c_i) is, the greater the difference between clusters and other clusters is.

II. DDOS ATTACK DETECTION METHOD BASED ON K-MEANS

The detection method based on k-means includes four modules, namely flow table collection module, feature extraction module, data training module and attack identification module. The flow table collection module sends the flow table request message to the switch regularly through the encrypted channel. After the switch receives the request, it also sends the flow table information to the controller through the encrypted channel. After receiving the flow table information, the feature extraction module extracts the relevant feature values from the flow table information. In the training stage, the feature extraction module takes the extracted feature vector to the data training module as data, while the training module clusters the data according to the k-means algorithm and establishes the detection vector based on the results, and then provides the detection vector to the attack identification module. In the detection phase, by comparing the feature vector extracted from the feature extraction module with the detection vector provided by the data training the attack identification module judges whether the DDoS attack occurs. If so, the warning can be given. If not, recollect the flow table and conduct the next detection. The implementation of DDoS detection algorithm is shown in figure 3, where the dotted line is the training stage and the solid line is the detection stage.

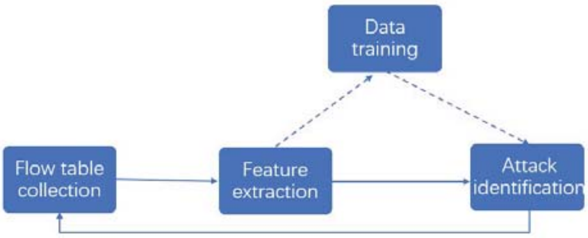


Figure 3 DDoS attack detection algorithm implementation diagram

A. Flow Table Collection

Under the openflow protocol, the controller sends the ofp_flow_stats_request message to the switch on a regular basis, asking the switch to submit its flow table information. Its implementation has been written to the feature_extractor.py file.

B. Feature Extraction

DDoS attack is an attack that its attacker controls a large number of puppet machines to send flow packets to the target so that bandwidth or resources of the target host have been consumed. Attackers often forge large numbers of random source IP addresses, random port packets. They often reduce the size of packets in order to increase the number of the sent packets. As a result, the number of ports in the network increases as the source IP addresses proliferate. Changes in these data are relatively stable on a normal network. However these flow features will change drastically in the event of a DDoS attack. Therefore, the attack flow can be detected by calculating the feature changes of network flow in unit time. According to the above analysis, 5 features can be extracted from the relevant information in the flow table items to detect DDoS attacks:

1) Average number of Packets in Per flow(AP)

Normally, the average number of packets per flow is relatively stable. Once a DDoS attack occurs, the number of flow table items increases while the number of packets per flow table item generated by the attack decreases. Therefore, when an attack occurs, the value drops. It’s shown as formula 3.

$$AP = \frac{\sum_{i=1}^{flow\ num} packet\ num_i}{flow\_num} \ldots\ldots\ldots\ldots\ldots （3）$$

In it flow_num is the number of flows received per unit time and $packet\_num_i$ is the number of packets in flow(i).

2) Average number of Bytes of per packet (AB)

In a normal network, flow is usually accompanied by data interactions and all the data are of meanings and large. This feature should be stable on the whole. But when a DDoS attack occurs, the packets forged by the attacker are usually small and meaningless. It’s shown as formula 4.

$$AB = \frac{\sum_{i=1}^{flow\_num} bytes_i}{\sum_{i=1}^{flow\_num} packet\_num_i} \ldots\ldots\ldots\ldots\ldots （4）$$

In the formula, flow_num is the number of flows received per unit time, the $bytes_i$ is the number of bits in flow i, and $packet\_num_i$ is the number of packets in flow i.

3) Port Generating Speed (PGS)

In a normal network, the number of ports changes relatively steadily. But when DDoS attacks occur, a large number of random ports are generated. As a result, port generating speed will change dramatically in the event of an attack. It’s shown as formula 5.

$$PGS = \frac{port\_num}{time} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots （5）$$

In it, port_num is the total number of ports corresponding to each IP, and time is the cycle time to extract the flow table information.

4) Flow Generating Speed (FGS)

When a DDoS attack occurs, the number of requests to a specific host will increase greatly. Meanwhile, the bogus IP address cannot match the flow table, so the controller will have to send more corresponding flow information, which makes the change rate of the flow increase. It is shown as formula 6.

$$\mathrm{FGS} = \frac{flow\_num}{time} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (6)$$

5) Source IP Generating Speed (SIG)

Under normal network, the change rate of source IP address is relatively stable, but DDoS attack will randomly generate a large number of source IP addresses, resulting in an increase in the number of source IP addresses. Therefore, the change rate of the source IP address is one of the characteristics of DDoS attack. It is shown as formula 7.

$$\mathrm{SIG} = \frac{src\_ip\_num}{time} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (7)$$

In the formula src_ip_num is the number of source IP addresses in the time unit.

The data of the above five features constitutes the eigenvector in the corresponding unit time. D_i =<AP，AB，PGS，FGS，SIG>

C. Data Training

Since the normal state of the network is always longer than the attack time, the feature vector set extracted from the feature extraction module should include the vector features of normal flow and the vector features of a small amount of abnormal flow. Perform k-means algorithm clustering for the five-dimensional feature vectors selected for training set, and establish detection model.

The establishment of the detection model is described as follows:

1) Cluster n feature vectors extracted from the training set.

2) Arrange the clustering result c in ascending order according to the difference degree of the cluster, and get f(c_1) ≤ f(c_2) ≤⋯≤ f(c_k)

3) Find the smallest number of clusters x, and satisfy equation 8

$$\frac{\sum_{j=1}^{x}|c_j|}{n} < \mathrm{y} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (8)$$

In it, c_i represents the number of feature vectors of the cluster, and y represents the ratio of normal feature vectors to all feature vectors in the training set.

4) Mark c_1，c_2，⋯，c_x generated from the above calculation as normal cluster.

5) Construct the detection model m(c_i)=<$center_i$，r_i> with normal cluster, in which $center_i$ is the clustering center of c_i, and r_i is the maximum distance between the data object of c_i and its clustering center.

D. Attack Identification

The feature vector A extracted from the feature extraction module is detected with the detection model: if there is cluster ci that satisfies d(A,center$_i$)≤r$_i$ (i is one of the normal clusters), A is normal vector. The program returns to perform flow table collection, and the next round of detection begins. If there is no cluster ci that satisfies d(A,center$_i$)≤r$_i$, A is the attack vector and the program will give the warning of being attacked.

III. Experiment and Analysis

A. Experimental Environment

This simulation experiment topology is based on the Mininet network simulation tool. The controller uses the open source controller Ryu and the Openflow switch uses the Open vSwitch virtual switch. Normal flow simulation uses Scapy library programmed in python to generate flow and inject it into the Mininet network. The attack flow uses the Trafgen tool to generate the attack flow and inject it into the Mininet network.

The topology of this experiment is shown in figure 4. Three OVS switches, S1, S2 and S3, are deployed on Mininet, which are respectively connected to 5 terminals under them, numbered h1 and...... , h15. S1, S2 and S3 are all connected to the Ryu controller C0, while S1 to S2 and S1 to S3 are connected through the encrypted channel. In the figure, the blue solid line is the normal network connection channel, and the red dotted line is the encrypted transmission channel. It is stipulated that the network bandwidth of S1 and S3 connection is 100Mbps, and that of S2 and S3 connection is 100Mbps.

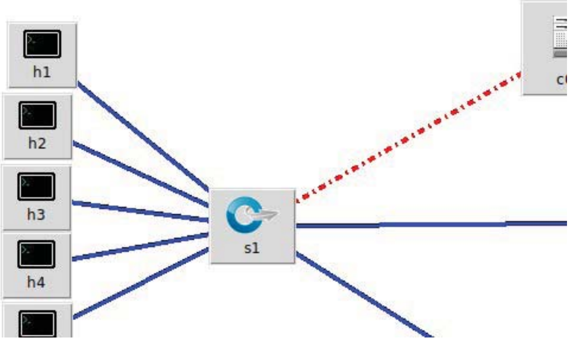
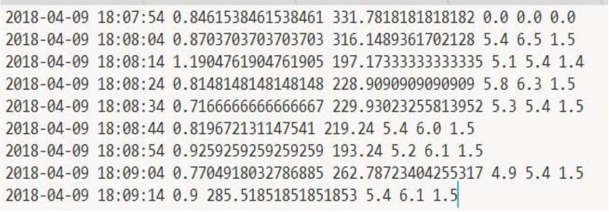


Figure 4 Experimental Topology Diagram

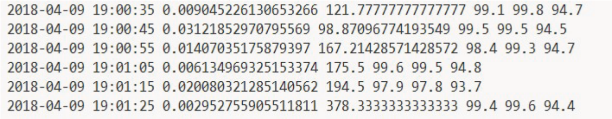
B. Experimental Process

After starting the controller, the normal flows are injected into the switch S1, S2 and S3. After that attack flows are injected into S3. The target of attack is terminal h1. The data recorded are time, average number of flow packets, average bits of flow packets, port generating speed, flow speed ratio, and source IP speed. Normal data and attack data are recorded in figure 5 and 6:



```
2018-04-09 18:07:54 0.8461538461538461 331.7818181818182 0.0 0.0 0.0
2018-04-09 18:08:04 0.8703703703703703 316.1489361702128 5.4 6.5 1.5
2018-04-09 18:08:14 1.1904761904761905 197.17333333333335 5.1 5.4 1.4
2018-04-09 18:08:24 0.8148148148148148 228.9090909090909 5.8 6.3 1.5
2018-04-09 18:08:34 0.7166666666666667 229.93023255813952 5.3 5.4 1.5
2018-04-09 18:08:44 0.819672131147541 219.24 5.4 6.0 1.5
2018-04-09 18:08:54 0.9259259259259259 193.24 5.2 6.1 1.5
2018-04-09 18:09:04 0.7704918032786885 262.78723404255317 4.9 5.4 1.5
2018-04-09 18:09:14 0.9 285.51851851851853 5.4 6.1 1.5
```

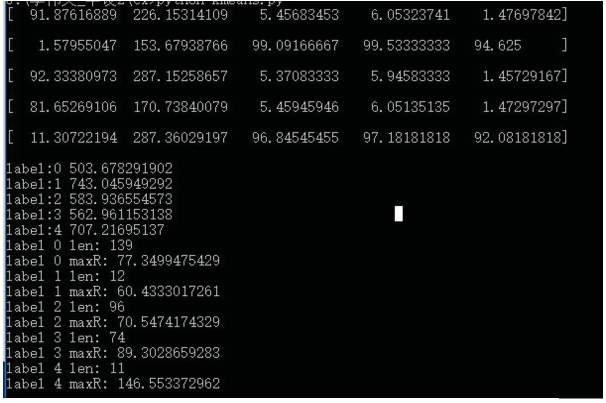
Figure 5 Normal Data



```
2018-04-09 19:00:35 0.009045226130653266 121.77777777777777 99.1 99.8 94.7
2018-04-09 19:00:45 0.03121852970795569 98.87096774193549 99.5 99.5 94.5
2018-04-09 19:00:55 0.01407035175879397 167.21428571428572 98.4 99.3 94.7
2018-04-09 19:01:05 0.006134969325153374 175.5 99.6 99.5 94.8
2018-04-09 19:01:15 0.020080321285140562 194.5 97.9 97.8 93.7
2018-04-09 19:01:25 0.002952755905511811 378.3333333333333 99.4 99.6 94.4
```

Figure 6 Attack Data

The sample of this experimental training set is a mixture of normal flow and DDoS attack flow. The flow injection time is 3100s, of which DdoS attack flow inserts about 230s Syn flood attack. A total of about 332 feature vectors were obtained from the time interval extracted by taking 10 seconds as the feature vector, respectively 309 normal feature vectors and 23 DDoS attack feature vectors. K value was selected as 5, that is, samples were clustered into 5 clusters. K-means clustering of the training set is shown in figure 7.



```
[  91.87616889  226.15314109    5.45683453    6.05323741    1.47697842]

[   1.57955047  153.67938766   99.09166667   99.53333333   94.625     ]

[  92.33380973  287.15258657    5.37083333    5.94583333    1.45729167]

[  81.65269106  170.73840079    5.45945946    6.05135135    1.47297297]

[  11.30722194  287.36029197   96.84545455   97.18181818   92.08181818]

label:0 503.678291902
label:1 743.045949292
label:2 583.936554573
label:3 562.961153138
label:4 707.21695137
label 0 len: 139
label 0 maxR: 77.3499475429
label 1 len: 12
label 1 maxR: 60.4333017261
label 2 len: 96
label 2 maxR: 70.5474174329
label 3 len: 74
label 3 maxR: 89.3028659283
label 4 len: 11
label 4 maxR: 146.553372962
```

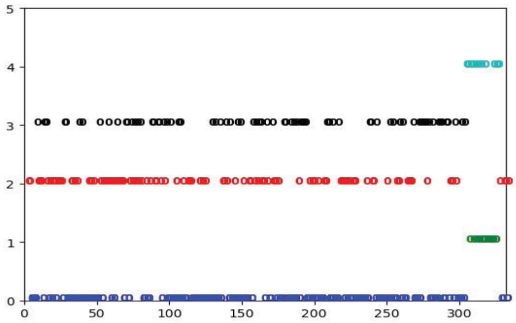


Figure 7 Training Data

What have been recorded here are the center of each cluster, the degree of heterogeneity of each cluster, the number of objects and the maximum radius of each cluster. For example, a cluster labeled 0 has 139 features and a cluster labeled 1 has 12 features. Cluster 0 to cluster 4 are labeled as c0, c1, c2, c3 and c4 successively. The clustering centers of each cluster are:

c0: [91.87616889, 226.15314109, 5.45683453, 6.05323741, 1.47697842]

c1: [1.57955047, 153.67938766, 99.09166667, 99.5333333, 94.625]

c2: [92.33380973, 287.15258657, 5.37083333, 5.94583333, 1.45729167]

c3: [81.65269106, 170.73840079, 5.45945946, 6.05135135, 1.47297297]

c4: [11.30722194, 287.36029197, 96.845455, 97.18181818, 92.081818]

The different degrees of each cluster are showed as below:

f（c_0）= 503.678291902;

f（c_1）= 743.04594929;

f（c_2）= 583.936554573;

f（c_3）= 562.961153138;

f（c_4）= 707.21695137.

According to the algorithm, c0, c2 and c3 are set as the detection model, r_0 = 77.3499475429 , r_2 = 70.5474174329，r_3 = 89.3028659283. Finally, the above data are detected by intrusion detection, and a series of 0 and 1 results were obtained, in which 1 was the attack and 0 was normal. The detection of DDos attack flow is realized.

C. Analysis of Experimental Results

Five data sets were collected in this experiment. In order to verify the test results, three formulas for testing are defined:

Detection rate = number of attack data correctly detected/total number of attack data

False alarm rate = number of normal data detected by error/total number of normal data

Error rate = number of incorrectly identified/total number of data

Based on the above formula, the detection results are shown in table 1 as follows:

Table 1 Detection results of DDoS attacks based on k-means algorithm

Test set	Detection rate	False alarm rate	Error rate
100 normal/100synflood	98%	1%	1.5%
286 normal/303synflood	99.3%	0.6%	0.6%
143 normal/54synflood	98.1%	1.3%	1.5%
269 normal/206synflood	98.5%	1.8%	1.6%
50 normal/ 20 udpflood	95%	0	1.4%
Average	97.78%	0.94%	1.32%

As can be seen from the results in table 1, this detection model has a good effect, with the average detection rate reaching nearly 98%. Therefore, the DDoS attack detection scheme based on k-means is effective. The DDoS attack detection method proposed in this paper selects the average number of flow packets, which can effectively identify the DDoS attack which occupies link bandwidth. If the training sample data of the k-means algorithm is enough and the K value is selected appropriately, the recognition degree of attack judgment will be increased and the misjudgment rate will be reduced.

Moreover, compared with neural networks algorithm and other algorithms, the training time of k-means algorithm is very short. This attack detection algorithm takes the normal flow features as the detection model, so it only needs to extend the judgment vector to improve the adaptability to network changes.

IV. SUMMARY AND PROSPECT

A. Summary

Based on the analysis of previous research results and their advanced research experience, this paper proposes a DDoS detection scheme based on k-means algorithm in SDN environment. The establishment of this scheme is based on the two hypotheses that the daily network is normal most of the time, and there is a significant difference between the data characteristics of normal situation and abnormal situation. At the same time, these two hypotheses are also true to the daily network situation. After demonstrating the validity of k-means clustering algorithm, the paper proposes 5 flow table features that can be used to detect DDoS attacks. Finally, the DDoS detection scheme was evaluated by SDN simulation experiment. The results show that the detection scheme is effective.

But there are also shortcomings in the paper: the test data of the experiment are not authoritative, which may make the experiment results contingent.

In addition, the DDoS attack detection algorithm has not been tested in the actual environment.

B. Prospect

SDN is still developing, and the north and south interfaces provided by its control layer are still controversial. Currently, there is no recognized standard for the north interface, and although the south interface has a recognized standard interface, Openflow protocol, there are also loopholes in the Openflow protocol itself. These loopholes make SDN vulnerable to DDoS attacks. DDoS is one of the main factors that harm network security. Although DDoS will hinder the development of SDN to some extent, in the long run, DDoS also promotes the development of SDN, and also encourages developers to propose more improved detection and defense schemes against DDoS attacks.

ACKNOWLEDGMENT

The work was supported by the Research Fund for the Doctoral Program of Higher Education of China (No. 20134407120017), the Ministry of Education - China Mobile Research Fund (No. MCM20121051), Guangdong Provincial Natural Science Foundation of China (No. S2012030006242), Guangzhou Pearl River Science and Technology Development Funds (No. 2011J2200089), Guangzhou Research Conditions of Construction Project (No. 2012224-12) and Educational Commission of Guangdong Province（No. 2017KTSCX074）

REFERENCES

[1] Jiye Cai, Hongqi Zhang, KunGao. Research on detection methods of OpenFlow network DDoS attack based on Sibson distance [J/OL]. Computer Application Research,2018(06):1-2.

[2] FuXiao, Junqing Ma, Xunsong Huang, Ruchuan Wang. DDoS attack detection based on KNN in SDN environment test methods [J]. Journal of Nanjing University of Posts and Telecommunications (natural science edition),2015,35(01):84-88.

[3] Xiaorui Wang, LeiZhuang, YingHu, Guoqing Wang, DinMa, Chenkai Jing.DDoS attack detection method for collaterals [J]. Computer Application Research,2018(03):911-915.

[4] Mengzhe Mei . Research on DDoS attack detection and protection based on multi-dimensional conditional entropy in SDN [D]. Nanchang University of Aeronautics,2016.

[5] Xiaoqiong Xu, Hongfang Yu,KunYang. DDoS Attack in Software Defined Networks: A Survey[J].ZTE Communications, 2017, 15(03): 13-19. distributed

[6] Linyuan Yao, PingDong and Hongke Zhang. Software definition based on object characteristics for network denial of service Attack detection methods [J]. Journal of Electronics and Information, 2007,39(02):381-388.

[7] Shihui Liu. Flood flooding attack detection and defense of links based on SDN and NFV [D]. Wuhan University Science, 2017.

[8] Guoyou Sun. Research on safe and reliable network control methods in SDN/OpenFlow network of cloud environment Study [D]. University of Science and Technology of China,2017.

[9] Xiaodong Xu, YanYang, GangLi. Network traffic anomaly detection based on k-means clustering [J] Wireless Communication Technology, 2013, 22 (4) : 21-26.

[10] Zhixian Zheng, MinWang. K-means clustering algorithm based on big data in network security detection Application [J]. Journal of Hubei Second Normal University,2016,33(02):36-40.