ABNORMAL USER BEHAVIOUR PERCEPTION METHOD IN SDN

Yabin Xu,* Xiaoqiang Li,* and Xiaowei Xu**

Abstract

In order to address the security issue of SDN specifically, an abnormal user behaviour perception solution is proposed. In this solution, we obtained historical data of users' network access behaviour from the flow-table entries of SDN, and applied complex network analysis to abnormal user behaviour perception. First, user's network access behaviours are divided into some sub-clusters. Then, we quantified user's network access behaviours by calculating their relative entropies, which are mapped to the appropriate network access behaviour patterns. Finally, users' real-time network access behaviours within a certain period of time are identified accordingly. In addition, by calculating the ratio of the sum of relative entropy in abnormal user behaviour, sub-cluster and the sum of relative entropy in complex network also can sense the current network security situation. Comparing with traditional user behaviour perception methods, this solution has higher recognition accuracy and lower algorithm complexity. Thus, it effectively improved the computing efficiency.

Key Words

Software-defined network, user behaviour perception, complex network, relative entropy

1. Introduction

Because of the cost advantage and of easy control feature [1], [2], SDN has been widely deployed in data centre. But, the solution of SDN network security problem has not been proposed by SDN organizations [3], [4]. For a solution which is specifically to SDN's network architecture and equipment feature, an effectively perception of abnormal user behaviour in SDN is very essential.

As a kind of new network architecture, SDN network is still in the initial stage of development, and there is no method for SDN user behaviour perception. In view of the traditional network architecture, researchers have proposed some methods of user behaviour perception. So, we can adapt traditional user behaviour perception methods to SDN and apply them into industries.

There are two types of traditional network behaviour perception methods. The first type, methods based on log mining, is mainly to analyse user behaviour by mining historical information in WEB server log [5]–[7]. But, the log is the record of users' accessing behaviour to a particular server in a past period of time, which may not be consistent with users' current accessing behaviour. What is more, the log only includes local information. Only with users' accessing behaviour to a particular server, we cannot refer users' accessing behaviour to other servers, and thus cannot know users' accessing behaviour in the whole data centre network. Therefore, the first type of methods is only suitable for the analysis of user users' accessing behaviour to a single server.

The second type, methods based on traffic analysis, can be subdivided into four categories. The first category method is based on the statistics. The second category method is based on rule matching. The third category method is based on machine learning. The fourth category division method is based on dividing sub-clusters of complex network.

The first category method which is based on statistics is to extract features of normal and abnormal data flow and establish rules. In the process of application, the byte number of each data flow, the packet number of each data flow, the duration of each data flow, and the time interval between each data flow is statistically calculated, and classified according to rules [8], [9]. The first category method is used widely but has low efficiency and poor real-time performance.

The second category method, which is based on rule matching, is to pre-determine the matching rules of different data flows and to make deep packet inspection (DPI) to realize rule matching [10], or to use the established Markov model to realize rule matching for every packet [11], [12]. This method can be used to analyse real-time use behaviour and has high efficiency and high accuracy. But, it cannot be used to identify data flow with encryption and unknown flow.

The third category method, which is based on machine learning, is to extract features of data flow, to establish support vector machine, artificial neural network and Naive

* School of Computer, Beijing Information Science & Technology University, Beijing 100101, China; e-mail: xyb@bistu.edu.cn, xql_84@163.com

** Department of Information Science, University of Arkansas at Little Rock, Little Rock, AR 72204, USA; e-mail: xwxu@ualr.edu

Corresponding author: Yabin Xu

Recommended by Dr. Xiaonv Hu

(DOI: 10.2316/Journal.206.2018.5.206-0059)

Bayesian classifier model based on these features, to adjust and optimize model through training and test, and finally to be applied into industries [13]–[16]. Although the accuracy and efficiency of this method are improved, the model has the problem of drift.

The fourth category method, which is based on dividing sub-clusters of complex network, is to establish complex network graph according to users' accessing behaviour, to divide users' accessing behaviour into sub-clusters with different features by using community detection algorithm, to match sub-clusters with user behaviour modes, and thus to identify normal and abnormal users' accessing behaviours [17]–[19].

Through the analysis of the structure of data centre network based on SDN architecture, we can see that a data centre is composed of a large number of physical or virtual servers, each server can be regarded as the server nodes in the network. At the same time, a large number of users can be considered as user nodes, and a lot of access relationship between the server nodes and the user nodes can be regarded as the edges connected the nodes. Therefore, the network composed of server nodes, user nodes, and edges can be regarded as a complex network; so, the theory and method for complex network can be used to analyse the validity and effectiveness of the relationship between the user nodes and server nodes.

Above all, we believe that the method based on dividing sub-clusters of complex network is advisable. In addition, in practical, we can easily access to the accessing behaviour of each data flow *via* the flow table of controller, in the sensing process of user behaviour in SDN. Compared with traditional network, SDN offers easier and convenient access to current user accessing behaviour. By redrawing sub-clusters of user accessing behaviour in the current time window and matching them with, we quickly judge the legality of accessing behaviour according to the behaviour pattern and realize the abnormal user behaviour sensing. In addition, by calculating the ratio of the sum of relative entropy in abnormal user behaviour, sub-cluster and the sum of relative entropy in complex network also can sense the current network security situation.

2. The Scheme Design of User Behaviour Perception System in SDN

At present, a large number of extensive functions and new applications are directly realized in the controller of SDN [20], [21]. In this project, to reduce the burden of SDN controller, also to meet the need of the transition at the same time, we apply the user behaviour perception service to the application layer of SDN as a third-party security application. The network topology of user behaviour perception system based on SDN is shown in Fig. 1.

The specific workflow is as follows:

SDN switch send data package which contains information of each data flow to controller, according to OpenFlow protocol. The information includes source IP, destination IP, source port number, and destination port number, *etc.*

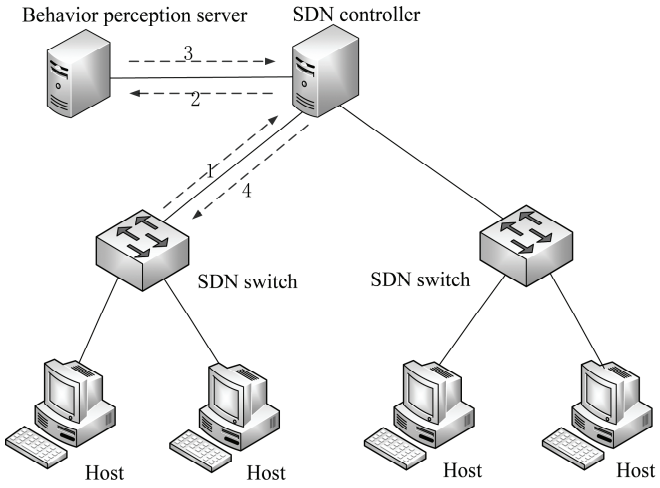

Figure 1. Network topology structure of SDN user behaviour perception system.

The controller analyses the data package and transmits the information to user behaviour perception server through RestAPI interface.

The user behaviour perception server updates complex network graph according to the IP and port information sent by controller and calculates users' behaviour patterns. Thus, the potential user abnormal user behaviour can be identified and sent to controller.

According to the feedback from behaviour perception server, for normal data flow, controller will send a flow-table entry which includes transmission path to SDN switch; and for abnormal data flow, controller will send rejected instruction to SDN switch.

3. User Behaviours Sub-Cluster Partition Based on Complex Network

3.1 Building of Complex Network Graph

In this project, users and servers are abstracted to node N in network, and the accessing relationship between users and servers is abstracted as edge E. If there is accessing relationship between n_x and n_y, they can be connected by edge e_{xy}. $G(n,e)$ denotes the constructed complex network graph.

The definition of node and edge of complex network graph in this project is as below.

Definition 1 (the node n of $G(n,e)$). *In network, IP represents the host of user or server, and port number corresponds to different network service. So the {IP, Port} is defined as a complex network node n. The analysed node is the source point of complex network, and denoted by {IP, Port}; the node connected with source point is destination node, and denoted by {DstIP, DstPort}.*

Definition 2 (the edge e of $G(n,e)$). *In the project, the complex network graph is constructed according to the access behaviour of nodes in network. The edge*

Table 1
Performance of Discovery Algorithms

	KL	Newman	GN	CNM
Time complexity	O(n^2)	O((m+n)n)	O(m^2n)	O ($n^*(\log_2 n)$)
Priori	Yes	None	None	None
Accuracy	Low	Higher	Higher	High

e represents the access relationship between user node and server node. {SrcIP, SrcPort, DstIP, DstPort} denotes the connection in the network.

In SDN data centre network, the information of SrcPort, DstPort, DstIP, and SrcIP is extracted from the flow-table sent by controller and used to build the original complex network graph.

3.2 User Behaviour Sub-Cluster Partition

By analysing complex network graph, we can get access rules of each node and divided nodes with same accessing rules into a behaviour sub-cluster. The identifying algorithms of behaviour sub-cluster can use community discovery algorithm in complex network and social network [22], [23]. There are some classic algorithms of identifying behaviour sub-cluster, such as KL algorithm [24], Newman algorithm [25], GN algorithm [26], CNM algorithm [27], *etc*. In addition, there are a lot of improved algorithms [28]–[31]. Table 1 shows the performance of each classical algorithm. Among them, m represents the edge, and n represents the node in the network.

Among them, CNM algorithm does not require priori information and has the higher accuracy. For CNM algorithm, the time complexity is only O ($n^*(\log_2 n)$), which is closer to linearity and more suitable for sub-cluster partition in large-scale complex network. So, we use CNM algorithm to divide sub-cluster in complex network graph.

In CNM algorithm, modularity is used to describe the tightness between user behaviour sub-clusters. Modularity refers to the difference between ratio of edge number in sub-cluster C_i to total edge number and ratio of the number of edges connected to sub-cluster C_i to total edge number.

The formula is defined as

$$Q = \sum\nolimits_i (e_{ii} - a_i^2) \tag{1}$$

Among them, e_{ii} represents the ratio of edge number in sub-cluster C_i to total edge number, $a_i = \sum_j e_{ij}$ represents the ratio of the number of edges connected to sub-cluster C_i to total edge number. CNM algorithm starts to divide sub-cluster when each node occupy a community and merges clusters in the direction of maximum modularity increasing or minimum modularity decreasing. Thus, we can get the maximum network modularity.

In order to improve the efficiency of CNM algorithm, three data structures are used [17].

(1) Modularity increment matrix: ΔQ : ΔQ stores each row of matrix as a balanced binary tree. We can find arbitrary element Δq_{ij} in time $O(\log n)$ by using balanced binary tree structure. Δq_{ij} is the modularity increment between sub-cluster C_i and C_j.

(2) Big top heap: H : H stores max$\{\Delta q_{ij}\}$, the max value of row i in modularity increment matrix ΔQ, and also i and j, the number of two corresponding sub-clusters.

(3) Auxiliary vector matrix a : a is used to store a_i in modularity calculation.

The specific steps of CNM algorithm are as below.

Step 1. To initialize ΔQ and H

We denote each user node in the network as a sub-cluster. The initial modularity is $Q = 0$, $a_i = k_i/2\,\mathrm{m}$. K_i indicates the connection degree of node i. We use (2) to initialize Δq_{ij} in the modularity incremental matrix ΔQ:

$$\Delta q_{ij} = \begin{cases} \dfrac{1}{2m} - \dfrac{k_i k_j}{(2m)^2} & \text{if node } i \text{ and node } j \text{ connected} \\ 0 & \text{if node } i \text{ and node } j \text{ not connected} \end{cases} \tag{2}$$

Building the largest heap H by extracted the max value of each row in initialized modularity increments matrix ΔQ.

Step 2. To merge sub-clusters

We choose the max Δq_{ij} from big top heap H, merge C_i and C_j, the corresponding user behaviour sub-clusters, and denote the merged sub-cluster as C_j.

Step 3. To update data

(1) To update modularity increments matrix ΔQ

Remove the element in i-th row and j-th column from ΔQ and use the following formula to update the j-th row and j-th column element.

$$\Delta q_{ij} = \begin{cases} \Delta q_{ik} + \Delta q_{ik} & \text{if Sub} - \text{cluster } C_k \\ & \text{connected to } C_i \text{ and } C_j \\ \Delta q_{ik} - 2a_j a_k & \text{if Sub} - \text{cluster } C_k \text{ is only} \\ & \text{connected to } C_i \\ \Delta q_{jk} - 2a_i a_k & \text{if Sub} - \text{cluster } C_k \text{ is only} \\ & \text{connected to } C_j \end{cases} \tag{3}$$

(2) To update big top heap H

According to the updated ΔQ, we update the big top heap H.

(3) To update auxiliary vector matrix a

According to the result of the merging, we use formula $a_i = a_i + a_j$ to update a_i, which is corresponding to the sub-cluster C_j.

(4) To record the merged value of modularity

The value of merged modularity is $Q = Q + \max\{\Delta q_{ij}\}$.

Step 4. To repeat Steps 2 and 3 until max $\{\Delta q_{ij}\} < 0$. Thus, sub-clusters in number of P are obtained.

4. Identifying User Behaviour Pattern

The nodes in each sub-cluster obtained from user behaviours sub-cluster partition by using CNM algorithm have same behaviour. But, the pattern of behaviour still cannot be determined. In the following, we quantified accessing behaviours by computing relative entropies and mapped the relative entropy to the corresponding user behaviour pattern.

4.1 Representation of User Behaviour Pattern in Sub-Clusters

Information entropy describes the uncertainty of event, which is related to the probability of the event. The smaller the probability of the event is, the greater the uncertainty is. On the contrary, the greater the probability of event to occur is, the less the uncertainty is.

We suppose that {SrcIP, SrcPort, DstIP, DstPort} is used to describe users' accessing behaviour in complex network. x can be any one of SrcPort, DstIP, DstPort. The value of x is in discrete set $A=\{a_1, a_2, \ldots, a_n\}$. We denote m as the total accessing time of SrcIP when it has relation with all the elements of set A, and m_i as the accessing time of SrcIP when it has relations with a_i. So, the probability that SrcIP has relation with a_i is $p\ (a_i)=m_i/m$. Entropy x is defined as

$$H(x) = -\sum_{i=1}^{n} p(a_i)\log p(a_i) \tag{4}$$

When the value of x follows equal-probability distribution, the information entropy is max. $H\max(x)=\log\min\{n,m\}$. General accessing time $m>n$, so accessing time n meets $2H\max(X)$. We suppose that $n\geq 2$ and $m\geq 2$ (the uncertainty of value exists). In order to take normalization process of information entropy, the relative entropy $R(x)$ is defined as

$$R(x) = \frac{H(x)}{H\max(x)} = \frac{H(x)}{\log\min\{n,m\}} \tag{5}$$

where $R(x)$ represents the statistical uncertainty of x: When $R(x)$ is small, the value of x is relatively certain, which means that the accessing frequency to one or more locations is relatively high. If $R(x)$ is large, the value of x is distributed evenly, and the accessing uncertainty of is relatively large.

When analysing a user's behaviour, we can calculate the time of network connection of three types: {SrcIP, SrcPort, *, *}, {SrcIP, *, DstIP, *}, and {SrcIP, *, *, DstPort} by user's SrcIP. * is an arbitrary value. {SrcIP, SrcPort, *,*} is the accessing behaviour which is corresponding to DstIP and DstPort when SrcIP and SrcPort are arbitrary values. {SrcIP, *, DstIP, *} and {SrcIP, *, *, DstPort}, respectively, represent all accessing behaviour of SrcIP to DstIP and of SrcIP to DstPort. According to the connection time that we calculated, we apply relative entropy theory to calculate relative entropy of different elements in the four tuple, and respectively, denote as R{SrcPort}, R{DstIP}, and R{DstPort}. R{SrcPort}, R{DstIP}, and R{DstPort}, respectively, represent the relative entropy of {SrcIP, SrcPort, *,*}, {SrcIP, *, DstIP, *}, and {SrcIP, *, *, DstPort}.

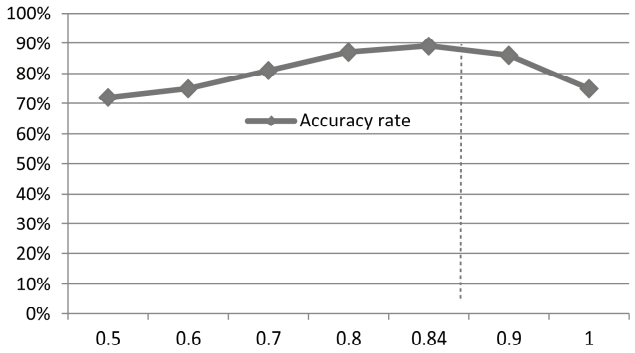

Figure 2. Accuracy rate of behaviour perception when α takes different value.

Given a user's SrcIP, we calculate the relative entropy R{SrcPort}, R{DstIP}, and R{DstPort}, respectively, by using the following rule to map the relative entropy R(X):

$$\mathrm{R}'(x) = \begin{cases} -1, & \text{if } 0 \leq \mathrm{R}(x) \leq 1-\alpha \\ 0, & \text{if } 1-\alpha < \mathrm{R}(x) < \alpha \\ 1, & \text{if } \alpha \leq \mathrm{R}(x) \leq 1 \end{cases} \quad (0.5 \leq \alpha \leq 1) \tag{6}$$

The mapping set we get, R(SrcIP) = {R′{SrcPort}, R′{DstIP}, R′{DstPort}}, is the user behaviour pattern. R′(x) = −1 indicates that SrcIP have connection with few x, and the connection is normal; R′(x) = 0 indicates that SrcIP have connection with some x but it is no more than the threshold of α, and the connection is normal but with a certain risk; R′(x) = 1 indicates that SrcIP have connections with a large number of x, and this kind of connection may be sniffer scanning. $x \in$ {SrcPort, DstIP, DstPort}.

For a DstPort, the value of relative entropy R{SrcIP}, {DstIP}, {SrcPort} can also be calculated, and then be mapped with the above rules. The mapping set R(DstPort) = {R′{SrcIP}, R′{SrcPort}, R′{DstIP}} is the behavior patterns of this access. If R(x) = 1, said that has connections with a large number of x, such connections could be DDOS attack.

The value of α decides the home range of the R(x) and affects the estimate for node's behaviour patterns by mapping set R(SrcIP), thereby further affecting the perception of user behaviour.

Based on statistical data of user accessing behaviour in data centre network, the experiment result shows that the accuracy of user behaviour is as below when α takes different values.

Figure 2 shows that the accuracy of behaviour perception varies with α, when $\alpha = 0.84$, the accuracy of behavior perception is the maximum, which is 89.1%. So, $\alpha = 0.84$ is taken as the boundary point of R(x) in this paper.

Table 2
User Behaviour Pattern

Behaviour	Behaviour Patterns
Client–Server (Web, FTP, Email, ...)	{−1, 0, 1}, {−1, 1, 1}
IP scan	{−1, 1, −1}, {1, 1, −1}
Port scan	{−1, −1, 1}, {1, −1, 1}
DDOS	{1, −1, −1}, {0, −1, −1}
Unknown pattern	The pattern is added to the known pattern table after analysing the sub-clusters of unknown pattern

Table 3
Computing and Identifying Method of User Behaviour

```
1   Input:relative entropy set of node n_i;
2   Output:user behaviour pattern
3   Calculate user behaviour pattern of node n_i in sub-cluster M(x)
4   Switch user behaviour pattern M(x)
5     Case Client–Server pattern, M(x) = {−1, 0, 1}, {−1, 1, 1}
6       Labelled the pattern of node n_i by common port, such as 80(web) and 11(FTP)
7       Labelled the pattern of node n_i by self-defined port, such as 3306(mysql)
8     Case port scanning pattern, M(x) = {1, −1, 1}, {−1, −1, 1}, etc.
9       Mark n_i as port scan mode
10    Case IP scanning pattern, M(x) = {−1, 1, −1}, {1, 1, −1}, etc.
11      Mark n_i as IP scan pattern
12    Case DDOS pattern, M(x) = {1, −1, −1}, {0, −1, −1}, etc.
13      Mark n_i as DDOS pattern
14    Default mark n_i as unknown pattern
```

The common user behaviour and user behaviour pattern is shown in the Table 2.

Computing the relative entropy of user node, then quantitative and mapping with user behaviour patterns. Thus, we can determine the user's behaviour patterns.

The computing and identifying method of user behaviour is given in Table 3.

The algorithm of calculating user behaviour pattern according to R(SrcIP) and R(DstPort) is similar to above.

4.2 Identifying Sub-Cluster Behaviour Pattern

The user behaviour pattern can be identified according to the behaviour pattern of nodes in the sub-clusters. When the user behaviour is quantified, the value of α may affect the identifying of node behaviour pattern. So, the node behaviour pattern in one sub-cluster may not be same. For example, sub-cluster C contains m user nodes, among them, $m-2$ nodes are client-server pattern, node $n1$ is IP pattern and node $n2$ is unknown pattern.

To identify the user behaviour pattern of the corresponding sub-cluster, the behaviour patterns of all nodes in one sub-cluster are statistically analysed by using probability formula $Pn = Cn/\text{Call}$. Cn represents the number of nodes that belong to the pattern n in sub-cluster C. Call represents the total number of nodes in sub-cluster C. P_n is the ratio of the number of nodes that belong to the pattern n in sub-cluster C to the total number of nodes in sub-cluster C. The greater the value of P_n is, the more the number of nodes that belong to the pattern n in sub-cluster C is. According to the maximum likelihood principle, the bigger the ratio of pattern n in sub-cluster C is, the bigger the possibility that the behaviour pattern of sub-cluster C is n. So, we take the behaviour pattern that most nodes in sub-cluster C belong, as the behaviour pattern of the sub-cluster C.

We can identify the behaviour patterns of all the sub-clusters in complex network graph, by computing behaviour pattern of the sub-clusters. The sub-clusters and their behaviour patterns together constitute user's behaviour perception pattern.

5. User Behaviour Perception

In the paper, we analyse user accessing behaviour by using rolling time window. Since that the user node n may have few accesses, and also accessing behaviour may change over time, the original behaviour pattern of user node may not be correct. So, the behaviour sub-clusters of user node n that falls into the access time window need to be recalculated. Then, we identify whether user's behaviour is abnormal according to the behaviour pattern of new behaviour sub-clusters.

In addition, through calculating the ratio of the number of edges in abnormal user behaviour sub-clusters and the number of the edges in complex network diagram within a specific time window, we can perceive the security situation of entire data centre network. It can also be converted to calculate the ratio of the sum of relative entropy in abnormal user behaviour sub-cluster and the sum of the relative entropy in complex network for the network security situational within a specific time window.

The algorithm used to user behaviour perception is in Table 4.

In the user behaviour perception process, it is not only to identify whether the user behaviour is abnormal but also to update the behaviour perception pattern. By updating complex network graph and re-dividing sub-clusters of user accessing behaviour that falls into the time window, it realizes the dynamic update of user behaviour perception model, improves the accuracy of user behaviour perception pattern, and solves the problem of concept drift in the traditional methods.

6. Experiment Result and Analysis

In the experiment, firstly, we divided the complex network of user accessing behaviours into sub-clusters. Then, we used Naive Bayesian Network (NBN), BLINC, and our method to perceive users' network accessing behaviour. Finally, we compared the accuracy of behaviour perception and time complexity of three methods.

We use DARPA98 data set in the experiment. The data of first 2 weeks is used as training set, and the data of last week is used as testing set. For a better comparison, in the experiment, the packets that cannot be recognized are removed, and only the packets that have been marked are used. In the experiment, we averaged the accuracy rates of five tests as the last result.

6.1 User Behaviour Sub-Cluster Partition

We constructed the complex network graph based on the data of first 2 weeks in data set. Then, we divided the complex network graph into sub-clusters. The result is shown in Figs. 3 and 4.

By comparing above two graphs, we can see that our project has a better dividing result of user behaviour sub-cluster. Among the four kinds of sub-clusters, the Client–Server behaviour sub-cluster is the biggest, IPScan and PortScan sub-clusters are smaller, and no DDOS attack sub-cluster. It is completely consistent with the actual situation of data sets.

Table 4
User Behaviour Perception Algorithm

```
1  Input: accessing information of user nodes that fall into the accessing window
2  Output: the identifying result of user behaviour
3  GetNewNote(a)      // Gets the node that falls into the access time window
4  UpdateComplexNet(a)      // update complex network
5  ChangeSubvariety(a)          // re-divide sub-clusters of a
6  GetBehaviorModle(a)          // get behaviour patterns of a
7  ReturnResult(a)              // return identifying result of user behaviours
8  if (the pattern of a is client-server)
9     return(normal behaviour)
10 if (the behaviour pattern of a is IPScan or PortScan)
11    return (sniffer scanning)
12 if (the behaviour pattern of a is DDOS)
13          return (DDOS Attack)
14 if (the behaviour pattern of a is unknown.)
15    return(unknown user behaviour)
```

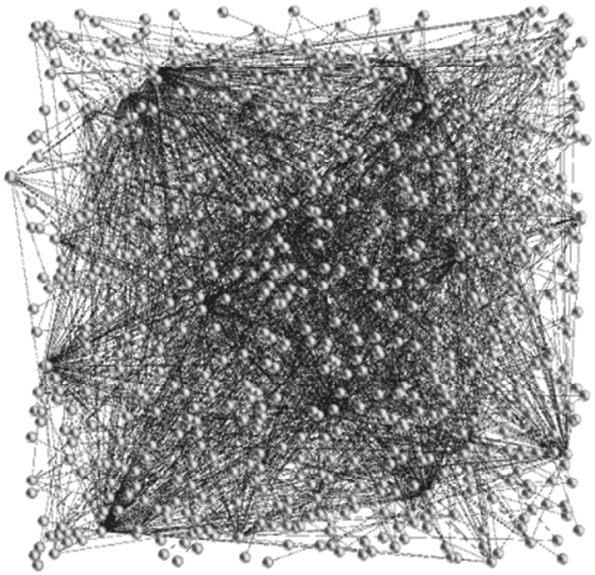

Figure 3. Complex network of network accessing behaviour.

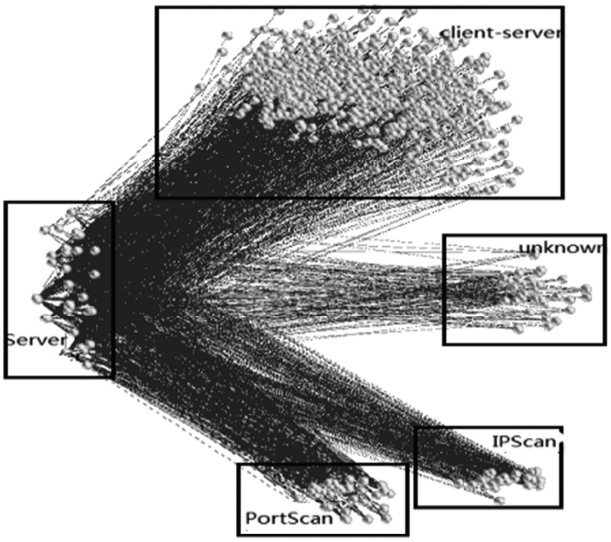

Figure 4. Dividing result of user sub clusters.

Table 5
Accuracy of User Behaviour Perception (Unit %)

	HTTP	FTP	E-mail	IPScan	PortScan	DDOS
NBN	74.83	77.27	76.02	75.71	74.39	75.66
BLINC	76.54	82.09	83.91	79.53	81.62	80.74
This scheme	80.27	86.14	85.74	86.10	87.31	85.11

6.2 The Accuracy of User Behaviour Perception

Table 5 is the accuracy rate of different methods when we use the same testing set.

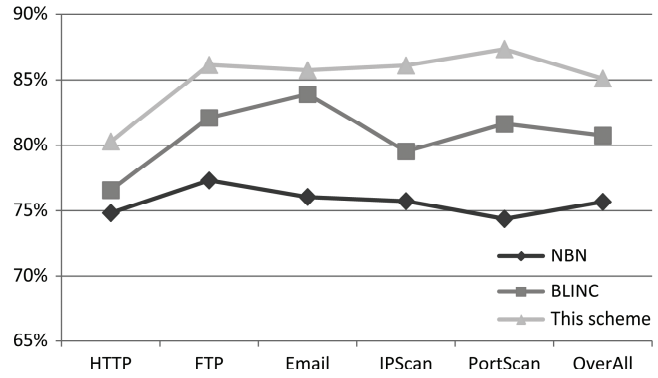

Figure 5. The accuracy rate curve of user behaviour perception.

In Table 5, HTTP, FTP, and E-mail belong to Client–Server pattern; IPScan and PortScan are, respectively, IP scan pattern and port scan pattern; and DDOS for distributed attack pattern.

Figure 5 shows the accuracy rate curve of user behaviour perception based on Table 5.

By comparing the accuracy rates of four different behaviour patterns and overall in Table 5 and Fig. 4, we can see that the overall accuracy rate of our project is highest. Compared with BLINC, our project has a slightly higher accuracy rate of dividing for Client–Server (HTTP, FTP, E-mail) pattern. The accuracy rate of dividing for IPScan, PortScan, and DDOS patterns improved greatly in our project. Compared with NBN, our project has superiority in identifying Client–Server, IPScan, PortScan, and DDOS.

6.3 Time Complexity

NBN consists of three parts: extracting statistical characteristics of data stream, establishing classifier based on Bayesian, and computing support vector machine (SVM). If we assume that the number of data packets is p, the time of extracting statistical characteristics of data stream is $O(p)$. The time complexity of establishing classifier and the computing SVM is larger than $O(p)$. The time length of NBN depends on the number of packets to be extracted. However, the network traffic follows heavy-tail distribution, which means most of traffic only belongs to a small number of data stream. Thus, n is proportional to $n\beta$, and $\beta > 2$, namely $O(p) > O(n^2)$.

BLINC consists of two parts: constructing user pattern and matching user pattern. So far, the best data mining algorithm of constructing user pattern is frequent subgraph algorithm. The time complexity of frequent subgraph algorithm is $O(n^4 * \mathrm{m} * 2^m)$. n is the number of data flow and also the number of points in the graph. m is the number of edges in the graph. The time complexity of its matching pattern is $O(kn)$. k is the number of patterns. Therefore, the complexity of the algorithm is about $O(n^4 * m * 2^m)$.

Our method mainly consists of four parts: constructing complex network, dividing CNM sub-clusters, calculating

user behaviour pattern, and identifying the sub-cluster behaviour pattern. The time complexity of constructing complex network is $O(m)$. The time complexity of dividing CNM sub-clusters is $O(n^*(\log n)2)$. The time complexity of calculating user behaviour pattern and identifying the sub-cluster behaviour pattern is $O(n)$. m is the number of edges in network graph, and n is the number of nodes. Assuming that the complex network graph is the most complete graph, the total number of edges in graph is $\frac{n(n-1)}{2}$. Then, the time complexity of our project is only $O(n2)$.

In summary, the method in our project has the lowest time complexity.

7. Conclusion

Effectively perceiving abnormal user behaviour in SDN and appropriately taking defensive measures are important to the security of SDN. Therefore, a user behaviour perception system scheme of SDN is put forward in the form of third party applications. Based on this, a behaviour perception mechanism based on complex network analysis is proposed in this paper. Firstly, the complex network graph is established based on the data of user network accessing behaviour within a rolling time window. Secondly, sub-clusters are divided according to their behaviour characteristics. Next, the relative entropy which describes user accessing behaviour is computed. Finally, we match the quantified result with the corresponding user behaviour pattern. When our project is used to perceive user behaviour, we only need to re-divide sub-clusters of nodes that fall into the rolling time window, based on the updated complex network graph. Then, the user behaviour pattern of new sub-clusters is the pattern of user behaviour. So, the perception of abnormal user behaviours is realized. In addition, we also give a simple method for network security situational awareness.

The experiment result shows that our method not only has a higher accuracy rate than traditional methods but also reduces the time complexity of algorithm. It is proved that the method of user behaviour perception proposed in this paper can solve the problem of user behaviour perception in SDN. It is important to improve the security of SDN.

Acknowledgement

This work was supported by National Natural Science Foundation of China (Grant No.61672101) and Beijing Key Laboratory of Internet Culture and Digital Dissemination Research (ICDDXN004).

References

[1] ONF, Software-defined networking: The new norm for networks, *ONF White Paper*, 2012.

[2] L. Cui, F.R. Yu, and Q. Yan, When Big Data Meets Software-Defined Networking: SDN for Big Data and Big Data for SDN, *IEEE Network*, *30*(1), 2016, 58–65.

[3] X. Li and Y. Xu, SDN access control strategy based on LE-Trie, *Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition)*, *27*(5), 2015, 674–682.

[4] C. Cui and Y. Xu. Research on load balance method in SDN, *International Journal of Grid and Distributed Computing*, *9*(1), 2016, 25–36.

[5] J. Wang, L. Li, and Y. Zheng, Research on mobile search user behavior review based on log mining, *Information Studies: Theory & Application*, *37*(3), 2014, 134–139.

[6] R. Wang and Yi. Yuan, An empirical study of user behavior analysis based on web log mining, *Library Tribune*, *31*(4), 2011, 100–102.

[7] Z. Ling and N. Hua, Research of user' searching behavior of library new technology of library resource discovery, *New Technology of Library and Information Service*, *18*(12), 2011, 74–78.

[8] H. Song, D. Wei, and G. Tang, Anomaly detection of single user behaviors based on pattern mining, *Journal of Chinese Computer Systems*, *37*(2), 2016, 221–226.

[9] Z. Tao, Abnormal network behavior detection technology based on statistical learning, *Big Data Research*, *1*(4), 2015, 1–10.

[10] W. Jian, Z. Zhizhong, and L. Yunlong, Design and implementation of DPI-based user's behavior perception system in LTE network, *Telecommunications Science*, *30*(7), 2014, 77–83.

[11] X. Xiao, Q. Zhai, X. Tian, and X. Chen, Novel method for anomaly detection of user behavior based on shell commands and DTMC models, *Computer Science*, *38*(11), 2011, 54–58.

[12] X. Tian, M. Duan, and C. Sun, Detection of anomalous user behavior based on shell commands and hidden Markov models, *Journal of Applied Sciences*, *26*(2), 2008, 175–181.

[13] P. Qiao and M. Li, A research on an improved intranet users' behavior audit model, *Journal of Harbin University of Science and Technology*, *16*(05), 2011, 57–60.

[14] Y. Xu, The abnormal network traffic recognition method based on optimized BP ANN model, *International Journal of Future Generation Communication and Networking*, *8*(3), 2015, 61–70.

[15] C. Qiu, Y. Xu, Y. Li, *et al.*, A fast identification approach to social network traffic based on unsupervised learning, *Mathematics in Practice and Theory*, *44*(3), 2014, 100–107.

[16] X. Liu and Y. Xu, Design of peer-to-peer traffic classification system model based on cloud computing, *Applied Mechanics and Materials , 182–183*, 2012, 1347–1351.

[17] Z. Luo, F. Ding, and X. Jiang, New progress on community detection in complex networks, *Journal of National University of Defense Technology*, *33*(1), 2011, 47–52.

[18] G. Tan, *Research on complex network pattern mining algorithm* (Xi'an: Xidian University, 2012).

[19] Z. Zhang, Y. Li, J. Wang, *et al.* User behavior perception based on mining complex network, *SCIENCE CHINA Information Sciences*, *44*(9), 2014, 1069–1083.

[20] J. Wang, J. Wang, H.-Y. Jiao, and Y. Wang, A method of OpenFlow-based real-time conflict detection and resolution for SDN control policies, *Chinese Journal of Computers*, *38*(4), 2015, 873–883.

[21] X.-L. Wang, M. Chen, and C.-Y. Xing, SDSNM: A software defined security networking mechanism to defend against DDoS attacks, *Journal of Software*, *27*(1), 2016, 2–15.

[22] J. Yang and Y. Zhuang, Towards behavior control for evolutionary robot based on RL with ENN, *Iaes International Journal of Robotics & Automation*, *1*(2), 2013, 463–474.

[23] H. Omranpour and S. Shiry, Reduced search space algorithm for simultaneous localization and mapping in mobile robots, *IAES International Journal of Robotics & Automation*, *1*(1), 2012, 49–63.

[24] M.E.J. Newman, Detecting community structure in networks, *European Physical Journal B*, *38*, 2004, 321–330.

[25] M.E.J. Newman, Fast algorithm for detecting community structure in networks, *Physical Review E*, *69*, 2004, 066133.

[26] M. Girvan and M.E.J. Newman, Community structure in social and biological networks, *Proceedings of the National Academy of Sciences of the United States of America*, *9*, 2002, 7821–7826.

[27] A. Clauset, M.E.J. Newman, and C. Moore, Finding community structure in very large networks, *Physical Review E*, *70*, 2004, 066111.

[28] H. Han, J. Wang, and H. Wang, Improving CNM algorithm to detect community structures of weighted network, *Computer Engineering and Applications*, *46*(35), 2010, 86–89.

[29] F. Havemann, M. Heinz, A. Struck, *et al.*, Identification of overlapping communities and their hierarchy by locally calculating community-changing resolution levels, *Journal of Statistical Mechanics Theory & Experiment*, *1*(01), 2011, 10–23.
[30] F. Ding, Z. Luo, J. Shi, *et al.*, Overlapping community detection by Kernel-based fuzzy affinity propagation, *ISA' 2010*, Wuhan, China, 2010.
[31] C. Liu, Y. Xu, and Z. Wu, Method of rapid detecting microblog communities, *Journal of Frontiers of Computer Science and Technology*, *9*(9), 2015, 1100–1107.

Biographies

Yabin Xu (1962–), M.Sc., Professor, School of Computer, Beijing Information Science and Technology University. His main research interests include cloud computing and big data, social networks, future networks, network security and privacy.

Xiaoqiang Li (1984–), M.Sc., School of Computer, Beijing Information Science and Technology University. His main research direction is future network.

Xiaowei Xu (1961–), Ph.D., Professor, Department of Information Science, University of Arkansas at Little Rock. His main research directions include data mining and machine learning.