Information Sciences 258 (2014) 403–415

Contents lists available at SciVerse ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Anomaly secure detection methods by analyzing dynamic characteristics of the network traffic in cloud communications☆

Wei Xiong[a,b], Hanping Hu[a], Naixue Xiong[c,*], Laurence T. Yang[d], Wen-Chih Peng[e], Xiaofei Wang[f], Yanzhen Qu[c]

[a] Institute of Pattern Recognition & AI, Huazhong University of Science and Tech., China
[b] Center of Computing & Experimenting, South Central University for Nationalities, China
[c] School of Computer Science, Colorado Technical University, USA
[d] Department of Computer Science, St. Francis Xavier University, Canada
[e] Department of Computer Science, National Chiao Tung University, Taiwan
[f] School of Computer Science and Engineering, Seoul National University, Republic of Korea

ARTICLE INFO

Article history:
Available online 18 April 2013

Keywords:
Anomaly detection
Cloud communication
Network traffic
Synergetic neural networks
Catastrophe theory
Chaotic dynamics

ABSTRACT

Cloud computing represents a new paradigm where computing resources are offered as services in the world via communication Internet. As many new types of attacks are arising at a high frequency, the cloud computing services are exposed to an increasing amount of security threats. To reduce security risks, two approaches of the network traffic anomaly detection in cloud communications have been presented, which analyze dynamic characteristics of the network traffic based on the synergetic neural networks and the catastrophe theory. In the former approach, a synergetic dynamic equation with a group of the order parameters is used to describe the complex behaviors of the network traffic system in cloud communications. When this equation is evolved, only the order parameter determined by the primary factors can converge to 1. Then, the anomaly can be detected. In the latter approach, a catastrophe potential function is introduced to describe the catastrophe dynamic process of the network traffic in cloud communications. When anomalies occur, the state of the network traffic will deviate from the normal one. To assess the deviation, an index named as catastrophe distance is defined. The network traffic anomaly can be detected by the value of this index. We evaluate the performance of these two approaches using the standard Defense Advanced Research Projects Agency data sets. Experimental results show that our approaches can effectively detect the network traffic anomaly and achieve the high detection probability and the low false alarms rate.

1. Introduction

Cloud computing represents a new paradigm where computing resources are being offered as services in the world via communication Internet. The security of cloud communications is an important task to guarantee the quality of services. However, the increasing new types of attacks appear in cloud communications network, which indicates that it is a severe challenge to detect the network attacks effectively and instantaneously.

☆ This work was supported by the National Natural Science Foundation of China (No. 60773192).

* Corresponding author.

E-mail addresses: ccnuxw@sina.com (W. Xiong), hphu@mail.hust.edu.cn (H. Hu), nxiong@cs.gsu.edu (N. Xiong), lyang@stfx.ca (L.T. Yang), wcpeng@cs.nctu.edu.tw (W.-C. Peng), dobby@mmlab.snu.ac.kr (X. Wang), yqu@coloradotech.edu (Y. Qu).

0020-0255/$ - see front matter © 2013 Published by Elsevier Inc.
http://dx.doi.org/10.1016/j.ins.2013.04.009

Generally, network anomaly detection approaches can be divided into two classes: signature-based detection and anomaly detection. In signature-based detection, the monitored network information is analyzed by comparing with attack characteristics from the large database. Once it is similar to a special attack in the database measured by some certain rules, the attack of this type will be detected [20,22]. Obviously, there are some evident shortcomings in this kind of detection mechanism as it is unable to detect attacks that do not exist in the database. It also needs time to update the database, such as establishing unknown attack characteristics and changing the relative rules. However, unknown attacks occurring for a second can cause serious damage to the network. In anomaly detection, the normal patterns of network behaviors such as network traffic load, breakdown and typical packet size have been established by training firstly. Then the attacks are detected by analyzing whether the monitored network information is inconsistent with the normal patterns [16,17]. It can possibly detect the unknown types of attacks without any further learning from new rules efficiently. Thus, in our research, we focus on the second class approach to detect anomalies of the network traffic in cloud communications.

On the other hand, several kinds of network information can be used to detect anomalies, such as recorded log files, audited data, packet contexts and trace. The information in recorded log files and audited data is always detailed. The anomaly detection based on these data generally obtains high accuracy [8,13,25]. However, there is an obvious deficiency for this information can only be extracted after attacks occurred. Thus, it is not suit for the real-time network anomaly detection. Packet context is a kind of information which can be parsed from the transporting packets in some way. Some researchers analyzed the packet contexts including TCP SYN packet information, port numbers, IP addresses and subnets, etc. [13,37] to detect the anomaly behavior in real-time. Unfortunately, when the amount of packets in network increases sharply, the time-consuming parsing process will make the real-time anomaly detection invalid. In further researches, more investigations [5,37] about the anomaly detection have been performed with the network traffic which was presented by the amount of transmission data of network at a given time. Without the parsing process of the packet, the accessing of the network traffic information is very fast by a packet counter which satisfies the real-time requirement. However, the information capacity of the network traffic is very limited, which makes it difficult to understand the relationship between the variety of the network traffic and the occurrence of anomalies. It can cause a relatively large amount of mistaken reports of the anomaly detection based on the network traffic. Thus, how to discover the evolving process of the network traffic and how to improve the accuracy of the real-time anomaly detection are still unsolved problems.

Currently, the reported approaches to detect anomalies of the network traffic anomaly detection mostly adopted the statistical physics methodology. In these methods, the macro-features of the network traffic, including self-similarity [14,15], entropy [18,34,37] and probability distribution [14,19], etc., were extracted, followed by using various pattern recognition techniques, such as neural networks [15], hidden Markov model [21], integrated access control [23] and machine learning [27], to detect network anomalies. Most of these methods comprehensively utilized several characteristics of the network traffic to classify anomalies. In this process, the characteristics were always considered as the primary factors of the network traffic fluctuation. They are mostly extracted from the sub sequences of the network traffic, which can reflect the continuous stationary variation of it, but can hardly describe its local sharp variation. Thus, these traditional methods can work relatively well only when the network traffic meets the stationary hypothesis.

The network traffic measurement in cloud communications has been carried out in Ref. [2]. However, there is little attempt to profile the network traffic in cloud communications. It was insufficient to ensure the instance of the cloud when cloud computing tenants facing security challenges. Traffic profiling has recently become an important method to protect and manage the backbone and edge networks, such as the establishment of normal and abnormal network behavior profiles [36,11], detection of traffic obfuscation and encryption [10], and accurate identification of network applications [9,12]. The behavior profiles of end hosts and network applications [36] are established without any presumption of what is normal or abnormal mode of transportation and communication. While Karagiannis et al. [12] studied the network traffic behaviors to classify traffic flows only using packet header information. Jiang et al. [11] create a traffic profile for each of the transport network by analyzing the aggregated traffic behavior. To enhance the security of networks, applications and data in the cloud, Xu et al. [35] proposes a method to develop a profiling-as-a-service architecture to characterize, understand and profile network traffic at multiple layers in the multi-tenant cloud communication network traffic environment. Thatte et al. [29] proposed a two-variable parameter detection method which established a simple statistical model of the generalized Poisson distribution about the anomalies and background traffic and then used the sequence probability ratio test to detect anomalies of the aggregated traffic directly. Simmross-Wattenberg et al. [28] proposed A non-binding alpha stable process model and used the statistical hypothesis testing to detect the network traffic anomaly. Shin et al. [26] used the K-means clustering analysis based on the network traffic to establish a Markov model based on the determined network status. And when the transition probability of the current network state is lower than that of the Markov model of the normal network state, it is determined that the current network status is abnormal.

In the cloud computing paradigm, the network traffic in cloud communications comes from multiple and heterogeneous administration domains. Moreover, it changes rapidly with respect to its behavior patterns due to the intrinsic change and heterogeneity of the tenants using the cloud and the elasticity of the exposed services. Under such circumstance of cloud computing, there are some major challenges: (1) a huge amount of network traffic in the cloud and the complexity of cloud tenants, (2) various security threats including traditional and emerging threats towards cloud tenants and cloud computing paradigm, (3) launching threats from inside and outside of the cloud [24]. Driven by these cloud factors, the generation of the network traffic in cloud communications is a complex process. Thus, the network traffic often shows non-linear, non-stationary and complex dynamical characteristics, which is time-varied chaotic dynamic [1,3,6]. Namely, it appears as a complex dynamic system. Its macro-behavior is yielded by synthetical and collaborative activities of these factors. When anomalies occur, the network traffic system in cloud communications will transform from the normal equilibrium to the abnormal equilibrium. This transformed process is controlled by a few primary factors (the order parameter) which predominate among the dynamical competition of many network factors. Moreover, this process is catastrophic, but not stationary. Thus, the detection accuracy of the traditional statistical physics methods based on the stationary hypothesis is influenced. As the normal collaborate behavior of cloud tenants is random, the network profiling of the border routers in cloud communications shows the characteristic of the randomness. From this perspective, it is similar with the traditional network traffic. The network traffic profiling of border routers is our focus.

If we know the model of the system, then we can use the method proposed in Ref. [33] to estimate parameters of the system, then to achieve the order parameter of dynamical system directly. However, the network traffic dynamical system is a complex time-varied chaotic system. We cannot get the order parameter of the system by using the above method. Thus, we first present a dynamical methods based on synergetic neural networks (SNN) [7] to get the order parameter and then to detect the network anomaly in cloud communications. In this method, the complex behaviors of the network system are described by using a synergetic dynamic equation. The synergetic order parameters solved by this equation essentially embody the effect of the primary factors of the network that guide the transformation of the network states. When the synergetic order parameters are evolved, only the order parameter determined by the primary factors can converge to 1 and the ones determined by the other factors are converged to 0. Finally, the anomalies can be detected mainly by referring to the order parameters determined by the primary factors of the network traffic in cloud communications. Then, we present another dynamical method based on the catastrophe theory (CT) [30] to detect the network anomaly in cloud communications. In this method, the sudden change process of the network is depicted by introducing a catastrophe potential function. When anomalies occur, the dynamic behavior of the network deviates from the normal behavior. An index named as catastrophe distance is proposed to assess the deviation from the normal behavior model to detect the network anomaly. To evaluate the performance of our approaches, we validate our methods on the standard Defense Advanced Research Projects Agency data sets and compare the results with a reported statistical physics method [32]. The results show that our approaches based on SNN and CT are effective to detect cloud network anomalies.

The rest of the paper is organized as follows. In the next two Sections we provide our methods based on SNN and CT separately. Then we show some experimental results and validate the performance of our methods. After that we make some discussions and propose our future research. The final section summarizes and concludes this paper.

2. The network traffic anomaly detection method based on SNN in cloud communications

As it is known that the network traffic often shows non-linear, non-stationary and complex dynamical characteristics and it is a complex dynamic system. Its macro-behavior is yielded by the collaborative activity of many factors [1,3,6]. Thus, the network problem can be considered as a high dimension problem determined by these factors. The generation of the network in cloud communications is the result of the collaboration of many factors. However, under some situations the changing trend of the network traffic in cloud communications is only determined by a few primary factors and less contribution of other factors. When the state of the network traffic is normal, the several factors almost equally dominate the network, the network traffic shows a strong randomness. Since anomalies occur, these factors do not equally contribute on the network traffic anymore. The primary factor is dominated by the behavior of abnormal users or attackers. The network traffic shows a strong certainty on the abnormal state. According to Synergetic, the primary factors leads to the generation of order parameters. The order parameters present the characteristics of the similarity and the randomness of the network traffic in cloud communications.

Synergetic, founded by Haken [7], is an interdisciplinary science which explains the formation and self-organization of patterns and structures in open system that is far from thermodynamic equilibrium. It concentrates on how the cooperation of the various individual factors of the dynamic system brings about spatial, temporal and functional structures on macroscopic scales.

According to synergetic [7], a dynamical system can be expressed as follows:

$$\dot{q} = -\frac{\partial V}{\partial q^{+}}, \quad \dot{q}^{+} = -\frac{\partial V}{\partial q} \tag{1}$$

where q is the system state; q^{+} is the adjoint state of q; V is the potential function of the system; $\dot{q}$ is the differential of q and the other is same on the following equations in this paper.

The control principle of synergetic proved that the stable models rely on those unstable ones. On the evolving process of the system, the stable models gradually decrease while certain unstable ones increase constantly and grow up into the primary factor in the system. In this way, the high dimension problem is transferred into the low dimension one and the values of the unstable models can be called order parameters. In fact, the final state of the system is determined by the unstable models with the largest original order parameter.

From the top-down perspective, synergetic describes the basic construction principle of pattern recognition and brings forward an important viewpoint: the process of pattern recognition is the process of pattern formation. In pattern recognition, macro-qualitative change of the system can correspond to pattern formation and the process of pattern recognition is equivalent to the transformation process from the testing data to the training data. Therefore there is a strong similarity between pattern recognition and pattern formation.

The network traffic anomaly detection in cloud communications based on SNN is a process of pattern recognition. In this process, the identified pattern is presented by the testing data and the prototype pattern is presented by the training data. The process to identify the testing data is a process to map the testing data to some existing training data set.

The treatment to identify the testing network traffic patterns q in cloud communications can be described as a dynamic process. After mapping the initial testing data $q(0)$ from intermediate state $q(t)$ to a training data vector v_k, that is, the training data vector v_k is most near with $q(0)$. The process can be described as $q(0) \rightarrow q(t) \rightarrow v_k$. Where, $q(0)$ is the testing network traffic data, v_k is the stored normal or abnormal network traffic, the intermediate state $q(t)$ is the order parameter ξ_k. This process can be specifically described by a dynamic Eq. (2). It is assumed that the number of the training data vector is M and the dimension of the training data vector is N. To guarantee the linear independency of the M training data vector, $M \leqslant N$ is required.

$$\dot{q} = \sum_{k=1}^{M}\lambda_k v_k\left(v_k^+ q\right) - B\sum_{k=1}^{M}\sum_{k'=1,k\neq k'}^{M}\left(v_{k'}^+ q\right)^2\left(v_k^+ q\right)v_k - C(q^+ q)q + F(t) \tag{2}$$

where q is the testing network traffic data vector in cloud communications with the original input data value $q(0)$. Scalar value λ_k is the attention parameter. Only when it is positive testing data can be identified. $F(t)$ is the fluctuation factor of the network traffic in cloud communications and can be ignored. Scalar values B and C are specified coefficients and must be greater than 0. v_k is the training data vector, $v_k = (v_{k,1}, v_{k,2}, \ldots, v_{k,N})^T$. Where, superscript T is vector transposition. v_k^+ is the adjoint vector of v_k (see Appendix A, the same below) and they have:

$$v_k^+ v_{k'} = \delta_{k,k'} = \begin{cases} 1, & k = k' \\ 0, & k \neq k' \end{cases} \tag{3}$$

v_k should be prepared with normalization and zero-mean:

$$\sum_{l=1}^{N} v_{k,l} = 0, \quad \sqrt{\left(\sum_{l=1}^{N} v_{k,l}^2\right)} = 1 \tag{4}$$

To reduce the dimension of the system, the order parameters ξ_k are features extracted from vector q. q can be represented by the order parameters ξ_k, a training data vector v_k and the remaining vector w:

$$q = \sum_{k=1}^{M}\xi_k v_k + w, \quad v_k^+ w = 0 \tag{5}$$

The adjoint vector of q is defined as follows:

$$q^+ = \sum_{k=1}^{M}\xi_k v_k^+ + w^+, \quad w^+ v_k = 0 \tag{6}$$

There is a relationship:

$$v_k^+ q = q^+ v_k \tag{7}$$

Typing (5) into (7), according to the orthogonal relationship, the order parameter ξ_k is defined as follows:

$$\xi_k = v_k^+ q \tag{8}$$

Style described in (2) is a powerful dynamics equation. If we neglect the fluctuation factor $F(t)$ of the network traffic in cloud communications, according to the Eqs. (1) and (2), the potential function V can be described as follows:

$$V = -\frac{1}{2}\sum_{k=1}^{M}\lambda_k\left(v_k^+ q\right)^2 + \frac{1}{4}B\sum_{k=1}^{M}\sum_{k'=1,k\neq k'}^{M}\left(v_k^+ q\right)^2\left(v_{k'}^+ q\right)^2 + \frac{1}{4}C\left(\sum_{k=1}^{M}\left(v_k^+ q\right)^2\right)^2 \tag{9}$$

According to the Eqs. (1), (2) and (8), correspondingly the dynamic equations and the potential function described by the order parameter are as follows:

$$\dot{\xi}_k = \lambda_k\xi_k - B\sum_{k'=1,k\neq k'}^{M}\xi_{k'}^2\xi_k - C\left(\sum_{k'=1}^{M}\xi_{k'}^2\right)\xi_k \tag{10}$$

$$V = -\frac{1}{2}\sum_{k=1}^{M}\lambda_k\xi_k^2 + \frac{1}{4}B\sum_{k=1}^{M}\sum_{k'=1,k'\neq k}^{M}\xi_{k'}^2\xi_k^2 + \frac{1}{4}C\left(\sum_{k'=1}^{M}\xi_{k'}^2\right)^2 \tag{11}$$

We know that when the potential energy of a system reaches the lowest value, the system dominated by the order parameters comes into the most stable state. In this case, the order parameters reach their extreme value. That is to say, the stable state of the network system is decided by the following formula:

$$\dot{\xi}_k = 0, \quad 0 \leqslant k \leqslant M \tag{12}$$

That is:

$$\dot{\xi}_k = \lambda_k \xi_k - B\sum_{k' \neq k} \xi_{k'}^2 \xi_k - C\left(\sum_{k'=1}^{M} \xi_{k'}^2\right)\xi_k = 0 \tag{13}$$

If we define:

$$D = (B + C)\sum_{k'=1}^{M} \xi_{k'}^2 \tag{14}$$

Then the following equations can be inferred from Eqs. (10) and (12):

$$\dot{\xi}_k = \xi_k\left(\lambda_k - D + B\xi_k^2\right) \tag{15}$$

$$\xi_k\left(\lambda_k - D + B\xi_k^2\right) = 0 \tag{16}$$

According to Eq. (15), SNN is constructed with three layers in Fig. 1. The top layer is the input layer in which unit j receives component $q_j(0)$ of need-recognized pattern vector's original value $q(0)$. All the order parameter components form the middle layer, where the order parameter ξ_k is gotten by summing all angle indexes j through each input value $q_j(0)$ multiplying its joint unit $v_{k,j}^{+}$. The active order parameter ξ_k recognizes the special training data chosen by the angle index k. According to the dynamical equation, SNN will be evolved to the end state that only one of order parameters survives and q_j is gotten through reciprocity and competition of D. The down layer is the output layer in which output pattern can be expressed $q_j(t) = \sum_k \xi_k(t) v_{k,j}$, where q_j is active of output unit j and ξ_k is the end state of the middle layer. There is $\xi_k = 1$ if $k = k_0$, otherwise $\xi_k = 0$. $v_{k,j}$ is the component j of the training data vector.

We consider the time series of the network traffic in cloud communications denoted as $y_1, \ldots, y_N$, which are sampled in bytes, bits or packets per time unit. The time series can be dealt by SNN to detect anomalies. We select some normal and the abnormal network traffic to construct a training data set including M components. The network traffic time series share the same size N. The purposed anomaly detection method is to distinguish the network traffic from the normal and abnormal.

The process of the anomaly detection includes two stages: the training stage is to learn the training data of the normal and abnormal network traffic and the testing stage to detect the network traffic anomaly. The detailed detection steps are given as follows.

(1) The training stage
  (a) Choose the training data vectors $\{y_1, \ldots, y_N\}$ from the train data set.
  (b) Deal the training pattern vectors $\{y_1, \ldots, y_N\}$ with normalization and zero-mean and then compute the training data vectors v_k.
  (c) Compute the corresponding adjoint vector v_k^{+} of the training data vectors v_k.
(2) The testing stage
  (a) Test on the testing data vector $q(0)$ consisted of the testing network traffic data in cloud communications dealt with normalization and zero-mean.
  (b) Compute the corresponding order parameter ξ_k of each training data according to the Eq. (8).

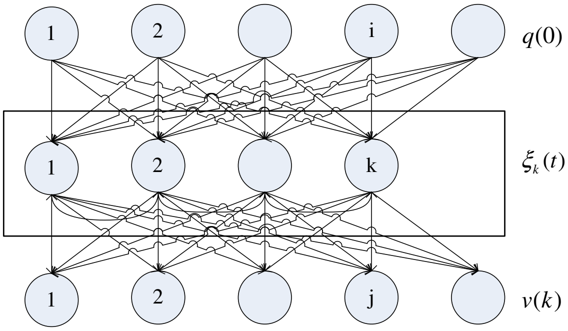

Fig. 1. The framework of SNN.

(c) Evolve by the following order parameter dynamic equation (Eq. (17)) until the order parameters becomes converging to a specific training data and then the specific training data is the detection result of the testing data vector $q(0)$. Thus the processes of the network traffic anomaly detection in cloud communications based on SNN have completed.

$$\xi_k(n+1) - \xi_k(n) = \gamma\left(\lambda_k - D + B\xi_k^2(n)\right)\xi_k(n) \tag{17}$$

where γ is the iterative step.

3. The anomaly detection of the network traffic based on CT in cloud communications

The evolving process of the network traffic dynamic system in cloud communications depends on the transformations among equilibriums determined by the primary factors. In the normal network traffic in cloud communications (when the state of the network traffic is normal, that is to say, no anomalies happened, the network traffic is called as the normal network traffic), the network traffic system in cloud communications maintains the stationary variation tendency dominated by the normal primary factors even if the fluctuation of the network traffic generated by the other factors may be great. The network state is called as the normal equilibrium state. After anomalies occur, the network traffic system in cloud communications maintains stationary abnormal equilibrium dominated by the primary abnormal factors. The network state is called as the abnormal equilibrium state. When anomalies are occurring, the network state will transform from the normal equilibrium state to the abnormal one driven by the primary abnormal factors. The change process of the network traffic is transient and catastrophic. The approach based on SNN solves the problem of the primary factors to domain anomalies in the network traffic dynamic system in cloud communications. But it cannot describe the sudden change process of the network traffic in cloud communications. In this section we discuss our anomaly detection approach based on CT which can describe the sudden change process of the network traffic in cloud communications and give the corresponding framework of an algorithm to detect the network traffic anomaly.

CT is a special branch of dynamical systems theory created by French Thom [30]. It explores that systems may respond to continuous changes in control variables by producing sudden, drastic and discontinuous changes from one equilibrium state to another. The catastrophe characteristics of a system are dependent on whether it has catastrophe properties. As mentioned in the above section, the evolving process of the network traffic dynamic system in cloud communications rely on the transformations among equilibriums. When anomalies occur, the network state will transform from the normal equilibrium to the abnormal one. The state change of the system is a sudden change process. We investigate the catastrophe properties of the network traffic which is shown in Fig. 2. From the plot of the observed network traffic data we can find that there are several burst traffic points. It shows the sudden jump process of the network traffic. The changing process is catastrophic but not stationary.

Considering the network users psychology and economy, the time benefit or patience of users makes users not linger in these unstable regions or web sites in order to reduce delay and operation time as possible (unless there are hot news or sole owner information). Thus, there exists inaccessible behavior. Besides, when the network traffic system in cloud communications reaches its capacity, it indicates a state of critical equilibrium state. But the equilibrium state is an ideal and instantaneous state that would be destroyed once it is disturbed by outside factors. The destroyed process is not a gradual process but a sudden jump. This process is unstable. It is feasible to explain the network traffic operation in cloud communications

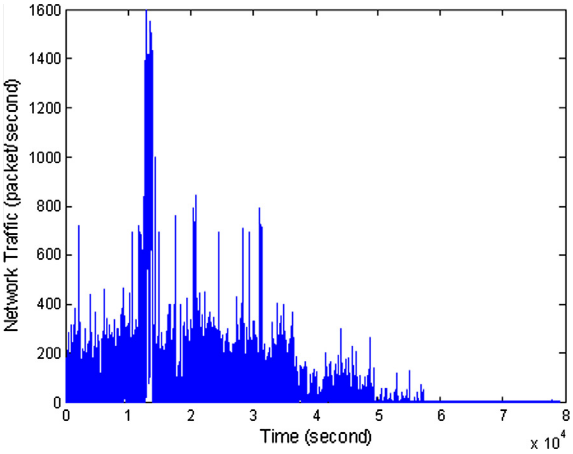

Fig. 2. The sudden jump of the network traffic in cloud communications.

by using CT. When a system has catastrophe properties, the corresponding catastrophe model can be established to describe it.

CT analyzes degenerate critical points of the potential function. The critical points satisfy the condition that the first derivative and higher derivatives of the potential function are zero. These are called the germs of the catastrophe geometries. The degeneracy of the critical points can be unfolded by expanding the potential function as a Taylor series in small perturbations of the parameters. When the degenerate points are not merely accidental, but are structurally stable, the degenerate points exist as organizing centers for particular geometric structures of lower degeneracy, with critical features in the parameter space around them. If the potential function depends on two or fewer active variables, and four or fewer active parameters, then there are only seven generic structures for these bifurcation geometries, with corresponding standard forms into which the Taylor series around the catastrophe germs can be transformed by diffeomorphism. Among the catastrophe models, the cusp catastrophe model is the most common model and our paper uses the cusp catastrophe model to describe the network traffic anomaly in cloud communications.

The potential function $F(x)$ of the cusp catastrophe model is as follows:

$$F(x) = x^4 + aux^2 + bvx$$

where x is a state variable, u, v are control variables and a, b are the coefficients.

The critical point set of the potential function $F(x)$ compose a balance surface. By seeking a derivative of $F(x)$ and making $F'(x) = 0$, the equilibrium surface equation $G(x,u,v)$ can be gotten:

$$4x^3 + 2aux + bv = 0$$

Making $F''(x) = 0$, the singularity set of the cusp catastrophe is gotten:

$$6x^2 + au = 0$$

According to the equations $F'(x) = 0$ and $F''(x) = 0$, the difference set $G(x,u,v)$ of the cusp catastrophe model is as follows:

$$8a^3u^3 + 27b^2v^2 = 0$$

The difference set of the cusp catastrophe model, which can reflect the relationship between each control variables and the state variable presented by the state variable, is very important because the difference set is in the control space that can be observed and in which all the sudden jumps will happen. Fig. 3 shows the basic form of the cusp catastrophe model. The top surface is the equilibrium surface of the cusp catastrophe model which is divided into upper sheet A and lower sheet C. When the state of the system transfers from the stable equilibrium state B to another stable equilibrium state D, there is a sudden jump between the stable state B and D and the sudden change phenomena appear. The bottom one is the control space represented by the control variables u, v.

Various features of the network traffic in cloud communications exhibit the nature of this dynamic system in different aspects. As the behavior of the system is dominated by the control variables of the catastrophe model, the state and the control variables extracted from the features of the system can be more accurate to analyze this model. In this paper, Hurst index [31] and dynamics associated factor [4] are selected as the control variables u, v of the cusp catastrophe model to describe the similarity, randomness and catastrophe characteristics of the network traffic in cloud communications. Hurst index reflects the degree of the self-similarity of the network traffic. Dynamics associated factor quantifies the structure dissimilarity and randomness between the current and the next state of the network traffic dynamics system. The volume of the network traffic is selected as the state variable x of the cusp catastrophe model. Thus the cusp catastrophe model is well constructed. The steps of the network traffic anomaly detection in cloud communications based on the cusp catastrophe model have two stages: the training stage and the testing stage.

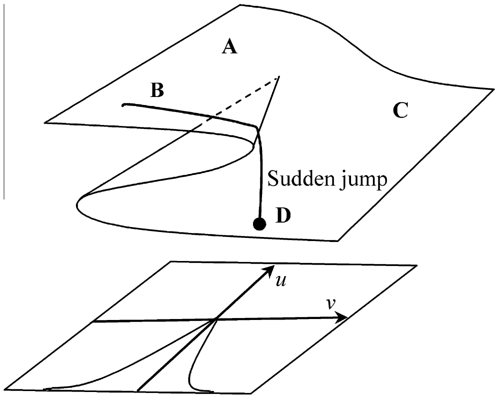

Fig. 3. A cusp catastrophe model.

The steps of the training stage are as follows.

(1) Consider the time series $y_1, \ldots, y_N$ of the training data, for each time t, construct the vector set $\{Y_t^p = (y_{t-p+1}, \ldots, y_t) | t = 1, \ldots, N - p + 1\}$ with the time window Win_p, where p is the size of the time window.
(2) Obtain the series of the state variable $\{x_t\}$ and the control variables $\{u_t\}$ and $\{v_t\}$ based on normalized features extracted from each vector Y_t^p.
(3) Compute the parameters a, b of the cusp catastrophe model using the series $\{x_t\}$, $\{u_t\}$ and $\{v_t\}$.

In the testing stage, the main steps are as following.

(1) Construct the vector Y_t^p (with the same time window Win_p in the training stage) of the testing data at the observed time i, which is labeled as observed point P_i.
(2) Extract the selected normalized features to present the state variable x_i and control variables u_i, v_i.
(3) Compute the catastrophe distance between the observed point $P_i(x_i, u_i, v_i)$ and the bifurcation set $G(x, u, v)$, labeled as D_p. The catastrophe distance D_p is defined as follows: Assuming that $P_i(x_i, u_i, v_i)$ is an observed point in the testing data of the network traffic in cloud communications and P_t (x_t, u_t, v_t) is a point of the equilibrium surface $G(x, u, v)$, the distance between two points $P_i(x_i, u_i, v_i)$ and $P_t(x_t, u_t, v_t)$, labeled as $D_E(P_i, P_t)$, is computed by the Euclidean distance. The catastrophe distance D_p between the observed point $P_i(x_i, u_i, v_i)$ and the equilibrium surface $G(x, u, v)$ is defined as:

$$D_p(P_i, G(x, u, v)) = \min_{P_t \in G(x,u,v)} \{D_E(P_i, P_t)\}$$

When the catastrophe distance D_p is beyond a given threshold η, it can be argued that there is an anomaly existed at the observing point $P_i(x_i, u_i, v_i)$. The threshold η is obtained by training.

4. Experiments

4.1. Experiment data set

The network traffic data in cloud communications used in our experiments is obtained from the standard Defense Advanced Research Projects Agency (DARPA) data sets, which is widely used in network intrusion detection and includes 5 weeks data [17]. The intrusion network monitor information of each day is recorded in four types of files, including tcpdump files, tcpdump list files, Solaris BSM audit data files, and ps monitoring data files. Only the tcpdump files record the network traffic information in cloud communications. In this paper, we focus on the network traffic information in cloud communications. Thus we extract the aggregated network traffic in bytes or packets per second from the tcpdump data files.

In the DARPA data set, the traffic data of the weeks 1 and 3 contain no attack. The 2 weeks' network traffic data are used as training data. The traffic data of week 2 lack the labeled information of the exact time when attacks occur. In addition, in data of week 4 and 5, there are various types of attacks mixed in the normal background traffic. Entirely labeled information of these attacks is also provided. The 2 weeks' network traffic data are used as testing data.

4.2. Results of the method based on SNN

In the classifying method using SNN, each point of the network traffic in cloud communications was detected as normal or abnormal data. The given constants B and C in SNN were chosen as $B = C = 1$, which are greater than 0 to guarantee the convergence of SNN. The attention parameter λ_k, $k = 1, 2$ is also initialized as 1 to make the same attentions with the two types of data (normal and abnormal data).

Fig. 4a and b show the evolving curves of the order parameters on the normal and abnormal network traffic data respectively. It shows that one order parameter converges to 1 and the other one converge to 0 in the two evolving process. Moreover, the order parameter with the converged value 1 just corresponds to the real training data of the testing data. Fig. 5 shows the detection result of a sub-serial traffic data on Friday of week 5. The normal traffic data are corresponding with the normal pattern (with the pattern value 0) and the abnormal traffic data are corresponding with the abnormal pattern (with the pattern value 1).

4.3. Results based on CT

We also detected anomalies of the network traffic data in cloud communications based on CT. The data in week 1 and 3 were used to construct control variables for the cusp catastrophe model. Then we observed the data on each day in week 4 and 5. We investigated how it deviated from the reference network traffic of the similar day in week 1 and 3. We made the serial of catastrophe distance as the input data and applied the anomaly detection algorithm based on CT to detect anomalies. The given parameters p, η were chosen as $p = 30$ and $\eta = 0.85$. The anomaly detection result of catastrophe distance of the second day in week 5 is shown in Fig. 6. The top gives the series of the catastrophe distance with a detection line by the

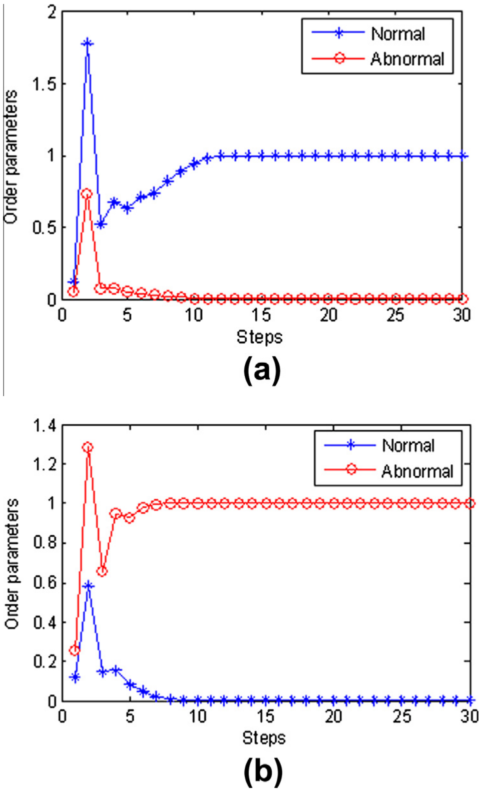

Fig. 4. Evolution results of the order parameters. (a) The evolution of the order parameter of the normal data. (b) The evolution of the order parameter of the abnormal data.

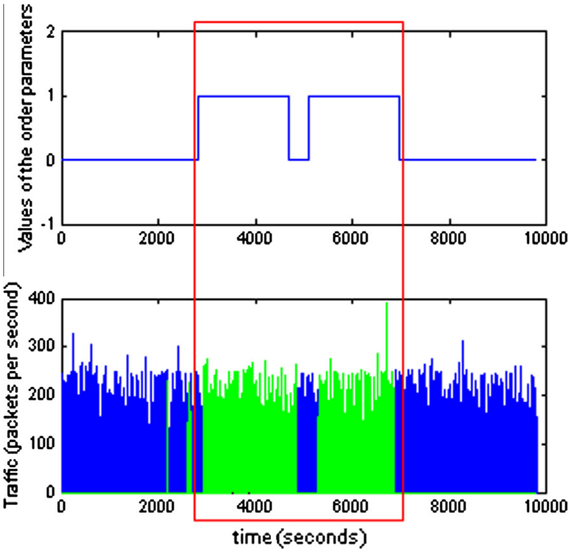

Fig. 5. Detection results of the network traffic anomaly in cloud communications based on SNN. (Top): the detection values by involving the order parameters of the abnormal data on Friday of week 5. (Bottom): the corresponding network traffic with anomalies (the blue parts represent the anomaly-free network traffic, the green parts represent the network traffic with anomalies) on Friday of week 5. The red rectangle identifies the time period when anomalies happened. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

trained threshold η ($\eta = 0.85$), which is obtained by training our model using data on Tuesday in week 1. The bottom shows the network traffic with anomalies. The red parts are the abnormal network traffic. The black rectangle identifies the time period when anomalies occurred.

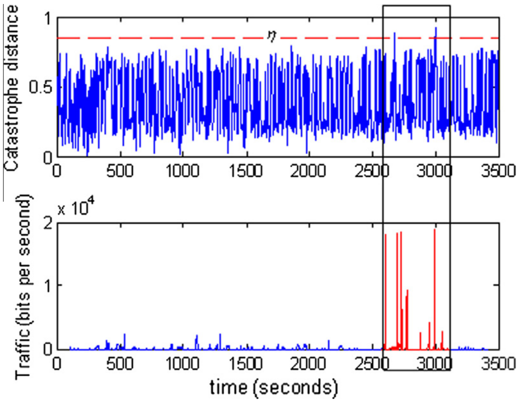

Fig. 6. Anomaly detection by using the catastrophe distance. (Top): the anomaly detection based on cusp catastrophe model. (Bottom): the network traffic in cloud communications on Tuesday in week 5 (with attacks) between 1:30 and 2:30 pm. The red parts represent the abnormal network traffic. The black rectangle identifies the time period when anomalies occurred.

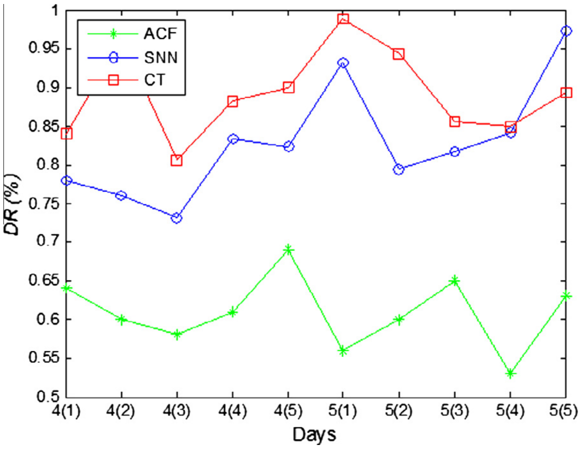

Fig. 7. DR of three methods of each day in week 4 and 5.

4.4. Results with comparisons

In order to validate the performance of our anomaly detection approaches, we use two metrics, the detection rate of abnormal events (*DR*) and the rate of the fault point (*FAR*), which are pretty standard for anomaly detection studies, to quantitatively evaluate the detected results. The definitions are as follows.

$$DR = \frac{NA_{detected}}{NA_{all}} * 100\%$$
$$FAR = \frac{NAF_{detected}}{NN_{all}} * 100\%$$

where $NA_{detected}$ is the number of the abnormal points which have been detected correctly, NA_{all} is the total number of abnormal points; $NAF_{detected}$ is the number of the normal points which have been falsely detected as an anomaly; NN_{all} is the total number of points to be detected.

The detection results based on SNN and CT are show in Figs. 7 and 8. *DR* and *FAR* of a continuous sub-serial whose length are longer than 3000 have been computed for each day. In the experiment based on SNN, the best detection result is obtained on Friday in week 5, where *DR* is 97% and *FAR* is 8.9%. The average *DR* and *FAR* of all the days are 83% and 8.3% respectively. In the experiment based on CT, the best detection result is obtained on Friday in week 5, where *DR* is 96% and *FAR* is 11.42%. The average *DR* and *FAR* of all the days are 86.62% and 9.06% respectively. It is noticeable that the average *DR* of method based on CT is increased 3.62% than that of SNN, but the average *FAR* of method based on CT is only increased 0.76% than that of SNN.

For further validating our methods, we compare our results with the traditional statistical physics method, an anomaly detection method based on the auto-correlation function (ACF) [32]. The *DR* and *FAR* of each day in week 4 and 5 are also shown in Figs. 7 and 8. The average *DR* of method based on SNN is increased 22% than that of ACF. Meanwhile, the average *FAR* of method based on SNN is decreased 1% than that of ACF. The similar comparison results are given in the method based on CT, its average *DR* is increased 25.72% than that of ACF and its average *FAR* is decreased 0.4% than that of ACF. In other words, methods based on SNN and CT improved the *DR* greatly and maintained the *FAR* in a low level as well.

5. Discussions and future works

In the DARPA dataset, there are four types of network attacks such as DoS (Denial of Service), PROBE (Surveillance/Probing), U2R (User to Super user (root)) and R2L (Remote to Local user) are collected. In our previous experiments, the detected anomalies are not identified with the exact attack types (all attacks are considered as anomalies). However, it is important to detect the exact type of attack in some applications. In our future work, we will try to detect it using SNN and further improve the detection results. Fig. 9a–e shows the evolving curves of the order parameters of the network traffic data of the

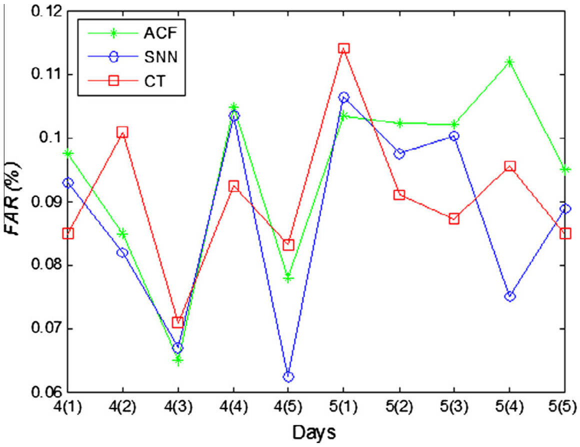

Fig. 8. FAR of three methods of each day in week 4 and 5.

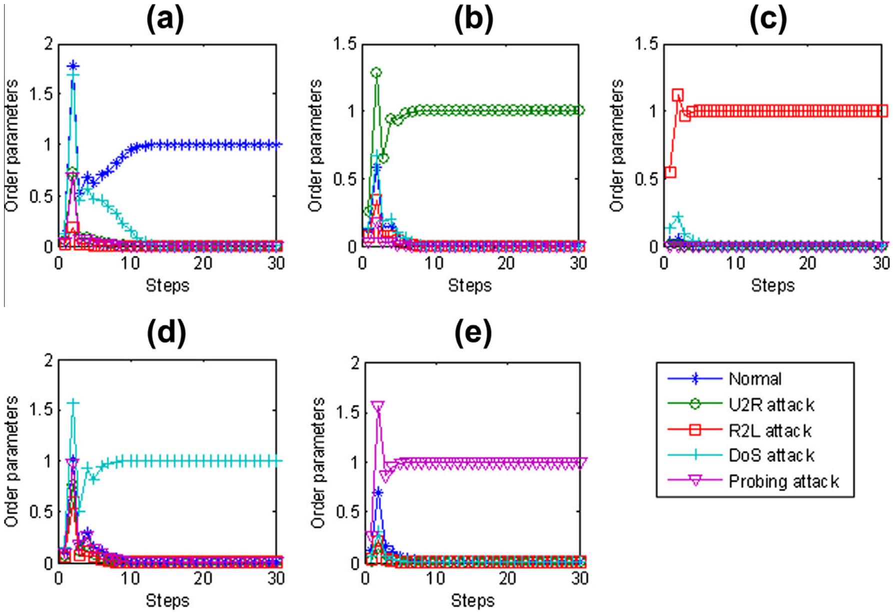

Fig. 9. Evolution results of the order parameters. (a) The evolution of the order parameter on normal data. (b) The evolution of the order parameter on U2R attacks. (c) The evolution of the order parameter on R2L attacks. (d) The evolution of the order parameter on DoS attacks (e) The evolution of the order parameter on Probing attacks.

normal, U2R attack, R2L attack, Probing attack and DoS attack respectively. It can be seen that the order parameter of each type attack with the converged value 1 just corresponds to the each training data of the testing data.

In the results of SNN, though *DR* between attack and normal patterns of the network traffic in cloud communications is relatively high, the detection results of different attack patterns are not very good because the difference between attack and normal patterns is more significant than that among the four attack patterns. When the attention parameters λ_k of the five patterns are set to be same, the possibility of identifying the normal pattern and the attack patterns is identical. In our future works, if some special attacks are focused on to be detected, one way to reach this target is to adjust corresponding attention parameters λ_k with the bigger values. For example, if we set λ_k for the DoS attack pattern as 0.4 and adjust λ_k for the other patterns as 0.15, *DR* of the DoS attack will be improved. Moreover, from Fig. 7 we can see that *DR* of several days based on SNN, such as on Monday and Tuesday in week 4, are not very high. These are due to the following primary reasons: In these days, the proportion of the U2R and R2L attacks is very high, about 67–80% in all attacks. The U2R and R2L attacks always depend on the system loopholes to induce the permission or overflowing errors, which slightly affect the traffic fluctuation of the cloud communication network. Thus, the performance of the detection will decrease.

In the method based on CT, we utilize a most commonly used cusp catastrophe model to implement the anomaly detection. However, other catastrophe models, such as the fold catastrophe model, and butterfly catastrophe model [30], are possibly used to describe the sudden change process of the network traffic in cloud communications. When a given catastrophe model is chosen, the suitable number of the state and control variables is adjustable. For instance, the butterfly catastrophe model is corresponding to one state variable and three control variables. For the future work, we can construct the corresponding catastrophe models of the each type of attacks. Thus, we can further detect the type of anomalies to improve the accuracy and decrease the false alarm.

In this paper, our research object is the network traffic in cloud communications, but it's not limited. Our approaches can be easily transplanted to other system to identify anomalies only if the data of the system are available in the form of a time series, such as the exchanging data of the stock market. By using our approaches, the exchanging anomalies also can be detected.

6. Conclusions

Cloud computing represents a new paradigm where computing resources are being offered as services in the world via communication Internet. However, there are several open issues that still need to be addressed such as the loss of control over sensitive data, definition of new management and integration models and lack of standards. Furthermore, many new types of attacks are arising at a high frequency, which makes the cloud computing services are exposed to an increasing amount of security threats.

In our study, two dynamic approaches based on SNN and CT are presented to detect the network traffic anomaly in cloud communications. Unlike most reported methods of the network traffic anomaly detection, these two approaches are constructed based on the dynamic characteristics of the network traffic in cloud communications. In the method using SNN, the competition mechanism of the multiple network factors is implemented by the evolution of the synergetic dynamic equation, which leads to the classifying process is greatly determined by the winner of the dynamic competition called as the primary factors. Thus, it is more effective than the traditional methods which regard that the multiple network characteristics are with the dominant factors of the evolving process of the network traffic in cloud communications. In the other method based on the catastrophe model, the catastrophe characteristic which is an essential dynamic characteristic of the network traffic is described by the corresponding catastrophe model. The anomaly detection based on this model is more reasonable and practical than the reported anomaly detection methods which extracted the statistical physics characteristics of the network traffic based on its stationary hypothesis. The experimental results show that the proposed approaches can achieve the high detection rates and the low false alarm rates. In a word, the anomaly detection constructed on the dynamic characteristics of the network traffic in cloud communications is considered to be and effective and potential research work.

Appendix A

A.1. The solving of adjoint vector

For some known vector v_k, the adjoint vector v_h^+ composes of the transposition and superposition vector v_k, $1 \leqslant k \leqslant M$, $1 \leqslant k \leqslant M$. M denotes the number of the vector v_k. That is

$$v_h^+ = \sum_{n=1}^{M} a_{nh} \overline{v_n} \tag{I}$$

In formula (I), $\overline{v_n}$ is the transposition of v_n, the coefficient a_{nh} is to make the orthogonal condition $\left(v_h^+, \overline{v_n}\right) = \delta_{nh}$ established well. Then use v_k multiplied by the formula (I), the following formula can be gotten:

$$\delta_{kh} = \sum_{n=1}^{M} a_{nh} (\overline{v_n}, v_k) \tag{II}$$

where δ_{nh}, δ_{kh} are unit vectors.

Making

$$A = (a_{nh}), \quad W = [(\overline{v_n}, v_k)]$$

There exist $I = AW$, I is an unit vector. Moreover

$$A = W^{-1} \tag{III}$$

In the formula $W = [(\overline{v_n}, v_k)]$, $\overline{v_n}$, v_k are well defined and have known value. So the matrix A composed of the coefficient a_{nh} can be determined. According the coefficient a_{nh}, it can be gotten

$$v_h^+ = \sum_{n=1}^{M} a_{nh}\overline{v_n}$$

Then the adjoint vector v_h^+ is determined.

References

[1] L. Amaral, J. Ottino, Complex networks, The European Physical Journal B – Condensed Matter and Complex Systems 38 (2004) 147–162.

[2] T. Benson, A. Anand, A. Akella, M. Zhang, Understanding data center traffic characteristics, ACM SIGCOMM Computer Communication Review 40 92–99.

[3] K. Chandra, C. You, G. Olowoyeye, C. Thompson, Non-linear time-series models of ethernet traffic, in: Submitted to INFOCOM '99, 1998.

[4] P. Cloetens, R. Barrett, J. Baruchel, J.P. Guigay, M. Schlenker, Phase objects in synchrotron radiation hard X-ray imaging, Journal of Physics D: Applied Physics 29 (1996) 133–146.

[5] Z. Fadlullah, T. Taleb, A. Vasilakos, M. Guizani, N. Kato, DTRAB: combating against attacks on encrypted protocols through traffic-feature analysis, IEEE/ACM Transactions on Networking 18 (2010) 1234–1247.

[6] V. Frost, B. Melamed, Traffic modeling for telecommunications networks, IEEE Communications Magazine 32 (1994) 70–81.

[7] H. Haken, Synergetic Computers and Cognition: A Top-Down Approach to Neural Nets, Springer, 2004.

[8] S. Horng, P. Fan, Y. Chou, Y. Chang, Y. Pan, A feasible intrusion detector for recognizing IIS attacks based on neural networks, Computers & Security 27 (2008) 84–100.

[9] Y. Hu, D.M. Chiu, J. Lui, Profiling and identification of P2P traffic, Computer Networks 53 (2009) 849–863.

[10] M. Iliofotou, B. Gallagher, T. Eliassi-Rad, G. Xie, M. Faloutsos, Profiling-by-association: a resilient traffic profiling solution for the internet backbone, in: ACM CoNEXT 2010, Philadelphia, USA, 2010, pp. 1–5.

[11] H. Jiang, Z. Ge, S. Jin, J. Wang, Network prefix-level traffic profiling: Characterizing, modeling, and evaluation, Computer Networks 54 3327–3340.

[12] T. Karagiannis, K. Papagiannaki, M. Faloutsos, BLINC: multilevel traffic classification in the dark, in: SIGCOMM'05, Philadelphia, Pennsylvania, USA, 2005, pp. 229–240.

[13] S. Kim, A. Reddy, Statistical techniques for detecting traffic anomalies through packet header data, IEEE/ACM Transactions on Networking 16 (2008) 562–575.

[14] A. Lakhina, M. Crovella, C. Diot, Mining anomalies using traffic feature distributions, in: SIGCOMM '05, 2005, pp. 217–228.

[15] S. Lee, D. Heinbuch, Training a neural-network based intrusion detector to recognize novel attacks, IEEE Transactions on Systems Man and Cybernetics Part A 31 (2001) 294–299.

[16] R. Lippmann, D. Fried, I. Graf, J. Haines, K. Kendall, D. McClung, D. Weber, S. Webster, D. Wyschogrod, R. Cunningham, Evaluating intrusion detection systems: the 1998 DARPA off-line intrusion detection evaluation, in: DISCEX, 2000, pp. 12–26.

[17] R. Lippmann, J. Haines, D. Fried, J. Korba, K. Das, The 1999 DARPA off-line intrusion detection evaluation, Computer Networks 34 (2000) 579–595.

[18] J. Mai, A. Sridharan, C. Chuah, H. Zang, T. Ye, Impact of packet sampling on portscan detection, IEEE Journal on Selected Areas in Communications 24 (2006) 2285–2298.

[19] I. Paschalidis, G. Smaragdakis, Spatio-temporal network anomaly detection by assessing deviations of empirical measures, IEEE/ACM Transactions on Networking 17 (2009) 685–697.

[20] V. Paxson, Bro: a system for detecting network intruders in real-time, Computer Networks 31 (1999) 2435–2463.

[21] Y. Qiao, X. Xin, Y. Bin, S. Ge, Anomaly intrusion detection method based on HMM, Electronics Letters 38 (2002) 663–664.

[22] M. Roesch, Snort-lightweight intrusion detection for networks, in: Proc. of LISA '99: 13th Systems Administration Conference, 1999, pp. 229–238.

[23] T. Ryutov, C. Neuman, K. Dongho, Z. Li, Integrated access control and intrusion detection for web servers, IEEE Transactions on Parallel and Distributed Systems 14 (2003) 841–850.

[24] N. Santos, K.P. Gummadi, R. Rodrigues, Towards trusted cloud computing, in: USENIX Workshop On Hot Topics in Cloud Computing (HotCloud), 2009, pp. 1–5.

[25] S. Shanbhag, T. Wolf, Accurate anomaly detection through parallelism, IEEE Network 23 (2009) 22–28.

[26] S. Shin, S. Lee, H. Kim, S. Kim, Advanced probabilistic approach for network intrusion forecasting and detection, Expert Systems with Applications 40 (2013) 315–322.

[27] T. Shon, J. Moon, A hybrid machine learning approach to network anomaly detection, Information Sciences 177 (2007) 3799–3821.

[28] F. Simmross-Wattenberg, A.-P.J. I, and P. Casaseca-de-la-Higuera, Anomaly detection in network traffic based on statistical inference and alpha-Stable modeling, IEEE Transactions on Dependable and Secure Computing 8 (2011) 494–509.

[29] G. Thatte, U. Mitra, J. Heidemann, Parametric methods for anomaly detection in aggregate traffic, IEEE/ACM Transactions on Networking 19 (2011) 512–525.

[30] R. Thom, Structural stability, catastrophe theory, and applied mathematics, SIAM Review 19 (1977) 189–201.

[31] X. WANG, B. FANG, An exploratory development on the Hurst parameter variety of network traffic abnormity signal [J], Journal of Harbin Institute of Technology 37 (2005) 1046–1049.

[32] X. Wei, H. Han-ping, Y. Yue, Anomaly detection of network traffic based on autocorrelation principle, Journal of Communication and Computer 4 (2007) 15–19.

[33] X. Wu, Z. Wang, Estimating parameters of chaotic systems under noise-induced synchronization, Chaos, Solitons & Fractals 39 (2009) 689–696.

[34] Y. Xie, S. Yu, A large-scale hidden semi-Markov model for anomaly detection on user browsing behaviors, IEEE/ACM Transactions on Networking 17 (2009) 54–65.

[35] K. Xu, F. Wang, L. Gu, Profiling-as-a-service in multi-tenant cloud computing environments, in: Singapore Palliative Care Conference Biopolis Singapore, 2012.

[36] K. Xu, Z.L. Zhang, S. Bhattacharyya, Internet traffic behavior profiling for network security monitoring, IEEE/ACM Transactions on Networking 16 (2008) 1241–1252.

[37] A. Ziviani, A. Gomes, M. Monsores, P. Rodrigues, Network anomaly detection using nonextensive entropy, IEEE Communications Letters 11 (2007) 1034–1036.